Developing Heuristic Evaluation Methods for Large Screen Information Exhibits Based on Critical Parameters

Jacob Somervell

Dissertation submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the degree requirements for

Doctor of Philosophy in Computer Science and Applications
at
Virginia Polytechnic Institute and State University

June 22, 2004

Dr. D. Scott McCrickard, Chair
Dr. Doug A. Bowman
Dr. John M. Carroll
Dr. David Hicks
Dr. Christopher L. North

Keywords: heuristics, evaluation, notification systems, critical parameters

Developing Heuristic Evaluation Methods for Large Screen Information Exhibits Based on Critical Parameters

Jacob Somervell

ABSTRACT

Evaluation is the key to effective interface design. It becomes even more important when the interfaces are for cutting edge technology, in application areas that are new and with little prior design knowledge. Knowing how to evaluate new interfaces can decrease development effort and increase the returns on resources spent on formative evaluation. The problem is that there are few, if any, readily available evaluation tools for these new interfaces.

This work focuses on the creation and testing of a new set of heuristics that are tailored to the large screen information exhibit (LSIE) system class. This new set is created through a structured process that relies upon critical parameters associated with the notification systems design space. By inspecting example systems, performing claims analysis, categorizing claims, extracting design knowledge, and finally synthesizing heuristics; we have created a usable set of heuristics that is better equipped for supporting formative evaluation.

Contributions of this work include: a structured heuristic creation process based on critical parameters, a new set of heuristics tailored to the LSIE system class, reusable design knowledge in the form of claims and high level design issues, and a new usability evaluation method comparison test. These contributions result from the creation of the heuristics and two studies that illustrate the usability and utility of the new heuristics.

Acknowledgements

This thesis is the result of three years of continued thought and effort. This work would not have been possible without the support of a system of friends and family.

First I would like to thank D. Scott McCrickard, my advisor. He has been a source of energy, a sound board for ideas, a golfing buddy, and certainly not least, an excellent advisor. He was always ready to talk about anything, and he was always available to me, either electronically or in person. I never felt that I could not ask him about anything, from the academic to the insane. His dedication to his students still amazes me.

I would also like to thank Christa Chewar. She has been my colleague since the fall of 2001, and I am honored to have met and worked with her. Her dedication and work ethic are nothing short of super-human. It is her constant drive and desire to produce valuable scientific findings that has motivated me on those occasions when it seemed hopeless.

I also want to thank Ali Ndiwalana for his continued support. He is always in good spirits and is easy to talk to about anything. We spent quite a bit of time together in the lab, and he has become a good friend. He has taught me to appreciate Linux, and I learned quite a bit about computer hardware from him.

I have to thank my family, which is quite large. I have four brothers and one sister, all of whom encourage me and share feelings of pride that I have accomplished so much. My parents were also supportive and provided a loving retreat when things got overwhelming.

My younger brother Logan was always supportive of me and we often talked about the good ol' days: playing video games, fishing, swimming – even working with our father. John, my next older brother, has probably been the closest to me during my time at Virginia Tech. He always seemed to have some project going on in which I was needed. These little jobs often provided a nice alternative to setting in front of a computer all day. Jason, Jim, and Lillian are the next eldest. I really didn't get to see them very much but they were always supportive and always welcomed me to their homes when I needed a vacation.

My mother and father have a direct impact on the fact that I was able to complete this thesis. They raised me to strive for my personal best and to never give up. These fundamental attitudes towards life can help anyone accomplish the goals he/she sets out to tackle. They are the best parents anyone could ever ask for, and they have always given me what I need.

Of course, I have to thank my wife, Cynthia, for being understanding throughout this process. She has sacrificed so much for me; I don't know how to express my gratitude. I only hope that I can make her as happy as she makes me.

There are numerous other people who have influenced me and the completion of this work. My committee members – John M. Carroll, Doug Bowman, Chris North, and David Hicks – each gave thoughtful and useful feedback on my work. There are too many graduate students in the Computer Science department at Virginia Tech to mention by name, but they are always friendly and approachable.

Jacob Somervell
June 22, 2004

To Cynthia

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Problem Description

Suppose you were asked to design and evaluate a new electronic poster that is to display important news, information, and upcoming activities within your local workplace. How would you go about starting your design, and furthermore, how would you assess how well it performs? Traditional approaches involve talking with potential end users to determine requirements, developing initial prototypes, performing formative evaluation of the prototypes, redesigning and developing a full system, performing summative evaluation of the system, and finally, deploying the product.

This process usually works well with multiple iterations in the earlier phases. However, given the constraints of this particular system, evaluation is a difficult problem. Setting up a prototype and having users work with the system is difficult due to the nature of the display. It is designed to allow the users to determine when they want to see the information, while simultaneously providing important information and updates. One would need to set up an environment that modeled the user's typical work style, present the user with the system, then try to assess how well he/she completes both the primary tasks as well as the tasks associated with the new system. Set-up and execution of this type of test is often too costly. Other methods of evaluation could rely upon expert analysis of the system, but existing evaluation tools do not readily apply to this type of system. Extensive effort would be necessary to modify pre-existing tools before they could be applied to this new situation. What we need is an evaluation tool that is applicable to the type of system that we are creating.

But, how would be go about creating an evaluation tool that applies to this type of system? Would we want to create a tool dedicated to this single system or would a more generic, system-class level tool be a better investment of our time? Evidence from preliminary work suggests that system-class level evaluation tools hold the most promise for long-term performance benchmarking and system comparison, over more generic tools or even tools tailored for an individual system [85, 56, 5]. A system class level tool is situated more towards the specific side of the generality/specificity scale; yet, it is still generic enough to apply to many different systems within a class. So, again, how would we go about creating a new tool for this type of system? The key to successful evaluation tool creation is focusing on the user goals associated with the target system class. This requires an understanding of the system class, in terms of these critical user goals. Basically, if given a set of attributes that accurately capture the user goals associated with a system class, one

could more readily create new evaluation tools for that class of system, based on those attributes. This work investigates the creation and testing of new analytic evaluation tools based on the notion of critical parameters.

1.2 Terminology

To understand the purpose and need for this work, we must understand some terminology. This work is focused on *usability evaluation methods* (UEMs), and more specifically, *heuristic evaluation* methods. UEMs are tools or techniques used by usability engineers to discover problems in the design of software systems, typically measuring performance against some usability metric (ease of use, learnability, etc). Heuristic evaluation is a specific type of UEM in which expert usability professionals inspect a system according to a set of guidelines. This method is analytic in nature because the experts review a system (through prototypes or screen-shots) and try to discover usability issues from inspection and reflection upon the guidelines. Other UEMs can be empirical in nature, i. e. they rely upon involving real users in situated testing for feedback on usability of a system. However, testing systems like our example require significant modification to existing UEMs. **We need a specific tool, like heuristics, that can support formative evaluation of these displays.**

Heuristics have been used throughout the HCI community for quick, efficient usability evaluation [66, 70, 69, 48, 40, 32, 56, 21]. They involve the use of guidelines, and target systems are evaluated by experts in a walk-through type process. Actual system use is not required, only simulation or mock ups of the target system are needed to assess the usability of the system. Although they are high-level and generic, when tailored to support the design model associated with a notification system, heuristics could provide more detailed insight into the information and interaction design aspects of these systems. However, generic heuristics are ill-suited for new applications, creating a gap in evaluation tool support. **A method for creating heuristics that focus on the key user goals of a system class would provide developers much needed evaluation support, and allow us to create a set of heuristics tailored to a system class.**

Usability metrics allow designers and evaluators to create a system that meets the needs of the users by focusing evaluation effort on important system functionality. These metrics are typically described at evaluation time and exist mainly as means for measuring system performance. In this work, we specifically build on the notion of *critical parameters*, a set of values or attributes that defines whether a system serves its purpose [68]. Besides being useful for guiding standard system evaluation, like typical usability metrics, critical parameters also provide higher level utility in system classification and categorization [62] that transcends single systems to focus more on the underlying principles that define a system class. By focusing on the most important issues in a system from the onset, these parameters can guide development and evaluation throughout the design process. In our work, we use critical parameters to guide the creation of a new evaluation tool designed to facilitate formative evaluation of specific system classes. **It is this more robust use of critical parameters that allows us to create new heuristics targeted to the key user goals associated with a particular system.**

The example system described earlier can be classified as a large screen information exhibit. Software applications that run on the large screens to provide interesting or useful information, during times when the large screen is not being used for presentations or meetings, are examples

of large screen information exhibits. It is this usage of these systems that is interesting to this research. We are interested in large screen interfaces because institutions are purchasing displays for the purpose of providing easily updatable information, yet software systems to support this need are lacking. **Providing evaluation support can help developers create effective systems.** These systems will be referred to as *large screen information exhibits* or *LSIEs* for the remainder of this work (see Section 2.3 for more on LSIEs).

Large screen information exhibits fall into a category of interfaces known as *notification systems*. Notification systems are interfaces which provide interesting or important information to users who are busy with other tasks [61, 62]. Familiar examples of notification systems include stock tickers, email biffs, system load monitors, and clocks. Other examples, perhaps not labeled by their creators as notification systems, include vehicle dashboard instruments, handhelds or cell phones with alerting mechanisms, and sometimes even fans and lamps [67]. **Notification systems are seeing rapid acceptance and as more and more people are willing to sacrifice attention for secondary information display, evaluation methods that ensure good design become increasingly important.** More on notification systems can be found in Section 2.1.

The defining characteristic of notification systems is that they are used in so called *dual task* situations. Dual task situations are those in which a user is engaged in two or more tasks simultaneously. Consider the college professor who watches for new email while writing a proposal as an example. A small icon appears in the system tray whenever she receives a new email. She can then decide whether or not to open her email client or continue working on the proposal. The main work this professor is trying to accomplish is the writing of a proposal. We call this a *primary task*. The *secondary task* in this example is watching for new email. The dual-task paradigm is important when discussing notification systems because without a dual-task situation, we no longer consider notification system goals. **It is this dual-task nature of notification systems that necessitates development of new testing techniques.**

Use of notification systems is increasing, and as people are more willing to attempt to multitask and use these systems, effective and efficient design will become increasingly important. But, tools to support the design and evaluation are lacking. Determining the most effective and efficient evaluation methods for specific interfaces has been pursued for years. Now, as information becomes available through newly developed interfaces for devices off the desktop, the need for design and evaluation becomes paramount in ensuring adoption and use. Missing UEMs for specific system classes puts extra burden on developers and usability professionals because they are forced to adapt and/or create methods tailored to their needs.

This research seeks to address this need by studying a specific type of notification system and the most effective usability evaluation methods for analyzing systems in this class. By focusing on LSIEs, we start filling the holes in the evaluation aspect of the emerging field of notification systems, while simultaneously supporting the development of cutting-edge software systems [79]. Leveraging critical parameters in the creation of new evaluation tools should provide necessary structure and focus to development effort.

1.3 Research Goals

This research deals with evaluating information design and interface usability for LSIE systems. LSIEs show great promise when users decide to pause their current work to look at the display.

By focusing on a single type of notification system (i.e. LSIEs), exemplification of the techniques utilized in this work is clearer. The following statement motivates and summarizes the nature of this work.

> Although new applications are being introduced as large screen display information exhibits, there is a lack of clear methods for recognizing when a system supports its intended goals. Critical parameters allow us to create tailored heuristics to facilitate earlier system testing, ensure quality designs, and improve design knowledge capture and reuse.

To deal with the lack of dedicated UEM materials for notification systems, this work describes a structured, repeatable heuristic creation method that is based on the critical parameters associated with LSIE systems. Critical parameters provide a classification scheme for different systems from a certain class. This classification allows one to systematically analyze multiple interfaces and extract the the underlying design tradeoffs. The following section details three phases of research geared towards the creation of heuristics based on critical parameters.

Research Plan

To develop and test new heuristics that are tailored to LSIE systems, based on critical parameters, three separate efforts were required. We will briefly discuss these phases here; detailed descriptions of the work come in later chapters.

Phase 1 – Creation

This phase involves the development of heuristics for large screen information exhibits. This depends on examination of five large screen systems based on the critical parameters for notification systems (Section 2.1.2). The general process involves methods from scenario based design [77, 13] and claims analysis [15]. Using scenarios for each system, claims are extracted and classified with respect to the critical parameters. With claims from each of the five systems classified in the notification system framework, heuristics for supporting the user goals can be developed based on the claims analysis. A detailed description of the processes utilized in the creation of large screen information exhibit heuristics is provided in Chapter 4.

Phase 2 – Comparison

This phase serves two purposes. The first purpose is to provide support for the heuristic set as a viable evaluation method. The second purpose is to show that the newly created heuristic set is at least as good as other methods for evaluating large screen information exhibits. This is necessary for showing that the creation method produces comparably good heuristics. To do this, we performed an experiment that pitted the heuristics against each other in an evaluation of three example large screen information exhibits. The set of heuristics developed in phase 1 (found in [88] and Chapter 4), along with Nielsen's heuristics [70], and a set for general notification systems [9] are the heuristics that we tested. These methods were compared using a subset of a UEM comparison technique recently introduced by Hartson, Andre, and Williges [40]. This comparison technique

involves calculation of each UEMs' thoroughness, validity, effectiveness, and reliability. The data necessary for each calculation was obtained during the evaluations. Full descriptions of the setup and execution of this experiment are provided in Chapter 5.

Phase 3 – Application

This phase involves concentrated effort to show the utility of the newly created heuristics and pilot test their use in real-world evaluation of large screen information exhibits. This consists of two experiments involving the use of the heuristics in guiding evaluation, as well as expert feedback from the international Human-Computer Interaction community. This work is necessary to show that the creation method actually produces usable and useful heuristics. The descriptions of these efforts are provided in Chapter 6.

1.4 Summary

This research seeks to develop a set of heuristics tailored to the LSIE system class, to support early evaluation and ensure quality in designs. In developing this new UEM, we leverage the critical parameters of the notification system design space, as well as SBD and claims analysis. The result is a structured heuristic creation method that can be repeated for other system classes. In addition, an experiment to investigate three LSIE systems with each of the three heuristic sets, comparing them with a recent comparison technique provides evidence of the utility of the newly created heuristics.

Contributions of this work include:

- **Critical parameter based creation of system class heuristics** We develop and use a new heuristic creation process that leverages critical parameters from the target system class. Researchers can now focus UEM development effort on a structured process that yields usable heuristics.

- **Heuristics tailored to the LSIE system class** LSIE researchers and developers now have a new tool in their arsenal of evaluation methods. These heuristics focus on the unique user goals associated with the LSIE system class.

- **LSIE system design guidance** In addition to the heuristics, we produced significant numbers of design tradeoffs from system inspection. These claims are useful to other system developers because the claims can be reused in disparate projects.

- **UEM comparison tool** Through our efforts to compare the new heuristics to other existing alternatives, we developed a new comparison technique that relies upon expert inspection to provide a simplified method for calculating UEM comparison metrics.

- **Deeper understanding of the generality vs. specificity tradeoff** Finally, we also provide more insight into the question of the level of specificity a UEM should have for a given system. We also find support for system-class specific UEMs, as other work has indicated.

Up to this point, a general description of the problem area, notification systems, large screen information exhibits, and the research approach to the problem has been introduced. This introduction serves as an overview of the proposed work, to both situate the work and provide motivation. More detailed descriptions follow in subsequent sections of this document.

The remainder of this document is organized as follows:

- Chapter 2 discusses appropriate literature and related work, situating our critical parameter based approach and providing motivation;
- Chapter 3 provides details on early studies that illustrate the need for an effective UEM creation method, it also illustrates the utility of claims analysis for uncovering problem sets;
- Chapter 4 describes the UEM creation process, including descriptions of the five LSIE systems (phase 1);
- Chapter 5 describes the comparison experiment, including discussion (phase 2);
- Chapter 6 describes three efforts to show the heuristic set produced in Chapter 4 is indeed useful and usable (phase 3);
- Chapter 7 provides a discussion of the implications of this work;
- and Chapter 8 provides detailed descriptions of the contributions and information on future work directions.

Chapter 2

Literature Review

We are interested in developing new heuristics for the LSIE system class, based on critical parameters. This chapter outlines and reviews prior work which investigates evaluation techniques, notification systems, critical parameters, and large screen displays and associated technologies. These areas are important to the research goals, in terms of reviewing what has already been done and what needs to be addressed, thereby situating this work and illustrating the logical place it will hold in the road to betterment. Each above mentioned area has its own subsection that discusses relevant work in that field.

2.1 Notification Systems

Before we discuss prior work that pertains to the creation of a new UEM for the LSIE system class, information on notification systems is necessary to ensure understanding of the types of systems with which we are concerned. The following paragraphs provide information on notification systems and the associated critical parameters that define the different types of notification system classes. Understanding the nature of notification systems and the underlying critical parameters provides motivation for the creation process.

Notification systems are information presentation systems which seek to provide important or useful information, without being overly distracting to other primary tasks [62, 61]. The types of systems existing in this classification were previously labeled “peripheral” or “secondary” displays [55, 87, 60]. This new moniker is used to stress the user goals and functionality associated with these systems. They really exist to provide “notifications” of changes to some information source. Users tend to run these applications to achieve a greater understanding and awareness of various information sources, while busy performing other tasks. Sometimes these notification systems are used to support current work activity, and other times they are used for completely separate tasks. In all cases, notification systems are part of some dual- or multi-task situation.

Familiar examples of notification systems include instant message buddy lists, email biffs, and system load monitors. They are used to keep track of friends, family, and coworkers-workers; or to monitor information sources (machine load, network traffic, status of large downloads). Other less familiar examples include displays and monitors for nuclear power plant safety inspectors and air traffic controllers. Various windows and audible sounds could inform these users of changes in critical information. These windows and sounds are examples of notification systems.

2.1.1 Examples of Notification Systems in Literature

Here we provide some discussion of examples of early notification systems. Most early notification systems were designed to reside on computer desktops. They existed as windows or icons residing around the periphery of the computer screen. Several of these desktop systems attempted to integrate multiple information sources into a single application.

Desktop

Irwin [57], Information Resource Watching In a Nutshell, as the name implies, was developed to provide a common location for monitoring information sources such as local news, sports, stocks, weather, and email. The application was intended to be used while one was busy with other tasks (writing, reading, surfing the web). Sideshow [12] was a similar system that resided on the right margin of the screen, somewhat like the toolbar at the bottom of Windows operating systems. Like Irwin, it provided an integrated approach to information monitoring. Users could decide what information they wanted on the side bar; anything from email to local traffic reports. It leveraged freely available information from web sources.

A more recent desktop notification system is the Scope [92]. This system sought to integrate different types of notifications (from your inbox, calendar, to-do lists) into a single area, where the status of various notifications could be assessed through quick glances. It resided in the lower right corner of the desktop and provided cues about status and additions to important items.

These desktop systems are only a small portion of the applications that have been developed for supporting various types of information sharing and awareness. Other systems involve the use of video or video 'snippets' [27]. In fact, several media spaces rely on video channels to support distance communication [83]. In addition to video, other systems leverage different types of information to enable awareness. The Peepholes system [33] leveraged *ruser* information (from Unix servers) to provide lightweight awareness of colleagues. It was implemented as a desktop system that ran in a small portion of the desktop.

These examples illustrate the dual-task nature of notification systems and provide some insight into the challenges that arise during evaluation. Modeling these situations for empirical evaluation is difficult, so we turn to analytic techniques. But, there is still a lack of support for analytic evaluation. **Again we see the need for dedicated evaluation tools but lack the requisite support for effective UEM creation.** This problem is exacerbated when considering notification systems that are not on a typical desktop.

Off-Desktop

Other notification systems can appear in off-desktop applications. Weiser's dangling string representation of network traffic [93], in-vehicle information systems [74, 39, 51], ambient media [47, 56], and multi-monitor displays are examples [37, 42]. These types of off-desktop notification systems leverage the physical space in which people work and exist to provide information while people are busy with other tasks.

An interesting example of a truly off-desktop notification system is "Phidget Eyes" [34]. This system leverages physical objects in the environment to reflect specific information states. A pair of fabricated eyes can open, close, and 'look around' to indicate various information states. An

example usage could be to monitor when colleagues are available in a distributed office; the eyes could open when a colleague came into his office.

Others are looking into real world interfaces (RWI) as notification systems [67, 63]. These notification systems are everyday lights, fans, and other electrical equipment that is attached to a computer control. Information can be represented with these devices, serving as notification systems, without taking up precious desktop real estate. Consider as an example a light that reflects when a meeting is scheduled. As the meeting time draws near, the light turns on and gets brighter. When the meeting time is reached, the light could flash on and off to let the user know it is time for the meeting.

Along with this myriad of platforms for hosting notification systems, large screen displays can be used to show information to users. They provide rich display capabilities and leverage the space in which they are located. Information shown on these displays would be visible from multiple locations within the space. See Section 2.3 for more discussion on large screens and how they tie in with notification systems.

These examples provide an idea of the variability in notification system design and implementation. It should be clear that notification systems can take on many forms and appear on many types of platforms. This variability can lead to disjoint evaluation efforts from researchers, and results may not be readily usable by others. **It is clear that a structured, repeatable UEM creation technique is necessary to provide the analytic methods for supporting formative evaluation.** The next section presents some background on a framework created to support evaluation of notification systems, to promote comparison and reuse. This framework uses critical parameters to allow for definition of various types of interfaces (or system classes) within the notification system design space. In fact, these critical parameters define the notification system design space [62].

2.1.2 Framework for Understanding Notification Systems

Discussing notification systems in a cohesive framework, defined by critical parameters, allows for effective evaluation and comparison. This ability stems from the fact that critical parameters capture the overarching goals of a system class, not just those for a single system. Instead of focusing evaluation on metrics derived from developer expectation, critical parameters provide grounded, reusable, and comparable metrics where evaluation is focused on determining if new systems provide advancements. Indeed, critical parameters provide the criteria for establishing long term performance measures so that we can assess whether new systems are "better" or "just different" [68]. We now describe a framework for describing notification systems based on the notion of critical parameters. The thrust of this work supports our goal of producing a structured, repeatable heuristic creation process by providing established parameters with which we can assess specific systems within a class.

Critical Parameters

William Newman put forth the idea of critical parameters for guiding design and strengthening evaluation in [68] as a solution to the growing disparity between interactive system design and separate evaluation. For example, consider airport terminals, where the critical parameter would be flight capacity per hour per day [68]. All airport terminals can be assessed in terms of this capacity, and improving that capacity would invariably mean we have a better 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" [68].

In addition, Newman argues that critical parameters can actually provide support for developing design methodologies, based on the most important aspects of a design space. This ability separates critical parameters from traditional usability metrics. Most usability metrics, like "learnability" or "ease of use" only probe the interaction of the user with some interface, focusing not on the intended purpose of the system but on what the user can do with the system. Critical parameters focus on supporting the underlying system functions that allow one to determine whether the system performs its intended tasks. Indeed, the connection between critical parameters and traditional usability metrics can be desribed as input and output of a "usability" function. Critical parameters are used to derive the appropriate usability metrics for a given system, and these metrics are related to the underlying system goals through the critical parameters. Thus, as we test and evaluate systems, we can determine if we are making progress in system design.

Critical Paramters for Notification Systems

In [62], we embraced Newman's view of critical parameters and established three parameters that define the notification systems design space. Interruption, reaction, and comprehension are three attributes of all notification systems that allow one to assess whether the system serves its intended use. Furthermore, these parameters allow us to assess the user models and system designs associated with notification systems in terms of how well a system supports these three parameters. *High* and *low* values of each parameter capture the intent of the system, and allow one to measure whether the system supports these intents.

Representing the high and low values for each parameter as 1's and 0's provides unique descriptions of eight classes of notification systems. As shown in [62], these eight classes cover all combinations of the levels of the three parameters. Furthermore, each class is unique, implying fundamental differences in the nature of each of the system classes. Figure 2.1 provides a graphical depiction of these classes, with labels capturing the nature of each. The framework described above [62] will be adopted in this research. It will be referred to as the "IRC". These three critical parameters are used to categorize all notification systems and correspond to varying levels in user goals for the notification (secondary task): interruption, reaction, and comprehension. These classes represent ideal instantiations of systems for each blend of the critical parameters. There can be many types of systems that hold varying levels for each of the critical parameters that still fall within a system class. The binary representation of 0 or 1 is only a simplification of a continuous spectrum from which many systems can be classified.

Interruption

Interruption occurs when attention is allocated from a primary task to the notification [62]. It is most easily seen when a user switches their current task to address the notification. This parameter deals with whether or not a user has the goal of being interrupted to receive information from the notification system. In the case where interruption is desired, we would have a high level (1), otherwise we would have a low level (0). In some instances, being interrupted from current work tasks could have serious negative consequences (like driving a car, or performing brain surgery).

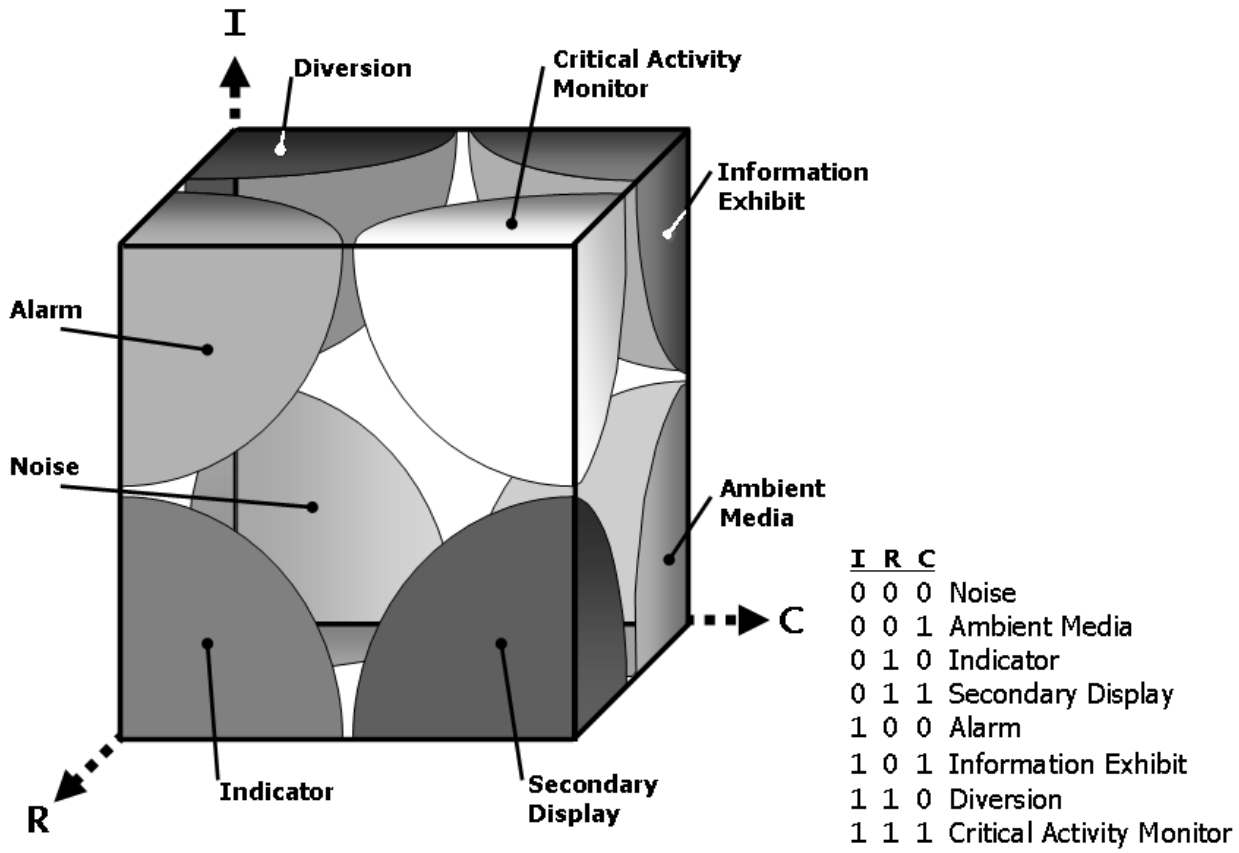

Figure 2.1: Notification system classes according to design objectives for each of interruption (I), reaction (R), and comprehension (C), simplified as high (1) or low (0).

But, in other cases, interruption could be desired or even necessary (think about a nuclear reactor about to blow). In fact, recent work suggests that interruptions become important for managerial tasks [43].

Examples in Literature Researchers have been interested in the effects interruption may have on ongoing tasks for years. Recently, focus has been on the negative aspects of interruption and methods for avoiding or reducing these impacts. Bailey et al. [4] looked at how annoying interruptions were as well as how it contributed to anxiety levels. Not surprisingly, unwanted interruptions were considered annoying and increased anxiety in users.

Other researchers have investigated negative aspects of interruption as well. Cutrell et al. looked at the effects interruptions have on memory and performance [23]. Participants were asked to find book titles in a listing, using a scrolling display. The titles were located down in the list and required some scrolling to find. Interruptions were initiated by the investigators at specific times and measures of how long it took the participant to find the title after being interrupted were used to analyze the effects of the interruptions. Interruptions in this context carried a negative impact on remembering the current task (specific book title), and on how long it took to find the title.

Similar work looked at the relationship of the interruption to the primary task [25]. Related interruptions (those that dealt with something similar to the ongoing task) were found to be more disruptive in terms of resuming the interrupted task than unrelated interruptions. Others have investigated how interruptions impact different task types. Czerwinski et al. investigated instant message interruptions on editing tasks, playing Tetris, and search tasks [24]. They found that the impacts of interruption indeed differed for the different task types. It was suggested that the more cognitively demanding tasks suffered higher levels of disruption.

Some researches looked to how to eliminate or reduce the negative aspects of interruption. McFarlane came up with a taxonomy for classifying types and styles of interruptions [65, 64]. He studied four types of interruptions to determine relative effects on ongoing tasks. He found that negotiated interruptions were better for reducing the disruptive effects associated with the interruptions. A similar finding by Trafton et al. suggests that having time to rehearse before task switching facilitates task resumption [91].

Self-defined Interruption These findings lead one to think about interruption as a necessary part of life but that we can reduce or alleviate some of the negative aspects if we can design systems to leverage our abilities to rehearse and negotiate our time. The ideas posed by McFarlane and Trafton led me to think about a particularly interesting type of interruption; that which is defined by the user. This *self-defined* interruption occurs without real thought and effort.

For example, consider the secretary busy writing a memo for his boss. The secretary is roughly half way through the memo and decides to stop and check the news headlines on his news ticker. The ticker has been visible on the screen the whole time while he was typing in his word processor but he explicitly decided to look at the ticker to get a sense of the current news items. The ticker perhaps only mildly distracted him, shifting some attention away from the typing task; but, the secretary defined his own interruption and looked at the ticker. Nothing really caused or prompted the interruption, but subtle cues in the moving ticker helped the secretary notice changes, prompting the secretary to look at that particular time. Since the interruption was self-defined, the secretary could easily rehearse the position of the current task to facilitate task resumption. This idea helps define the notion of large screen information exhibits, as this is one of the fundamental characteristics of typical LSIE use.

However, we need to expand the IRC framework to include this idea of self- defined interruption. The original framework only addressed distinct levels of each parameter, designated as high and low (or 1 and 0) [62] in an effort to simplify the presentation of the framework. But, the levels of each parameter can move along a range from 0 to 1. This implies other levels between high and low. For example, consider the idea of self-defined interruption, which is neither a high interruption goal nor a low interruption goal but something in between, or something that can cover a range from high to low. We do not desire high interruption because we need to stay focused on current work tasks, but we are more tolerable to distractions. We also do not want low interruption because we need to shift our attention to the notification system in order to assess the display. Hence, we need to include various levels and ranges for interruption, and similarly for each parameter, as potential user goals. Including a *medium* level for each parameter (represented as .5) can give us more flexibility when dealing with the idea of self-defined interruption, while simultaneously leveraging the utilility of the critical parameters for system design. Including continuous ranges (from medium to high, or low to medium, or even low to high) can also provide some flexibility

in system classification. We claim that self-defined interruption can be thought of as requiring a medium or medium-to-high level of interruption.

Attribute Leveraging A different look at interruption deals with specific design elements in a system and how they cause or reduce disruptive effects of interruption. Different uses of color, shape, and motion as information encoding mechanisms bring different levels of interruption. Healey and Enns looked at choosing effective colors in information design, to facilitate quick comprehension of information to avoid disruption [41]. Shape has been investigated in other studies as well. Chewar et al. actually compared color, shape, and position as encoding mechanisms to assess which would be better for supporting interruption [18]. They found that position was best overall but interestingly, as more interruption to the primary task was allowed, color and shape switched in which was better. Bartram also compared shape to motion with respect to grabbing attention [8, 7, 6]. Her findings suggest that motion is best for grabbing attention, especially as the target gets farther and farther from the center of focus. Other work has investigated how increasing the numbers of notifications can increase interruptiveness. Somervell et al. found that increasing the number of secondary tasks caused performance degradation in the primary task [87]. Interestingly, they found that when extra secondary tasks were added, users seemed to ignore them in favor of completing the primary tasks.

Some studies of interruption also illustrate methods for measuring interruption in relation to a primary task, illustrating the difference between a critical parameter and a test metric. A common strategy is to measure performance degradation (a test metric) on a primary task to assess the disruptive effects (critical parameter of interruption) of a notification system. McCrickard [60] and Somervell [86, 87] illustrate an effective testing methodology in which a dual-task experimental setup is used to assess various aspects of notification systems. These studies provide measures of interruption, as well as reaction and comprehension. The point being that the critical parameter suggests which metrics to use in the evaluation phases of system design.

Reaction

Reaction is the rapid and accurate response to important information provided by the notification [62]. An example of a reaction to a notification would be sending an instant message in response to a notification of a friend becoming available online. Often, the ability to quickly perform some action is the most important goal for a notification. This parameter refers to the goal of performing a specific action in response to a notification.

Sometimes users may not have the goal of responding to information in a notification. For example, a person using a stock ticker may simply want to know what the market is doing; buying or selling a particular stock may not be part of his/her intended usage. However, a different user could use the same notification system for the explicit goal of being able to know when to buy/sell stocks. These two different user goals illustrate the differences associated with notification systems and how they support intended user goals.

Examples Most of the systems that were mentioned earlier also deal with reaction to some extent. Scope, Irwin, and Elvin all support reaction to changes in the information. Appropriate reactions in Scope might include clicking on an urgent item to get more details, or leaving to go

to a meeting in response to a reminder [92]. Similarly, Irwin supported reaction by allowing a user to respond to emails or news events in a timely fashion [58]. The Elvin notification server sends notifications when specific events occur. Correct reactions to these notices include checking emails, opening web browsers, or placing a phone call [30].

There are some empirical evaluations of notification systems that deal with reaction to some extent [60, 84]. These studies used a dual-task setting to separate measures on primary and secondary task performance. Measures on reaction included timings for indicating certain states in the information had occurred. This method provides an effective technique for measuring the reaction support for a given notification system. Again we see how the evaluation metrics are directly related to the critical parameter.

Others also measured reaction in their studies of other aspects of notification systems. Czerwinski measured reaction times for responding to instant messages [25]. McFarlane investigated timings as well as correct responses in his investigations of interruptions [65, 64]. Empirical measures of timings seem to be an effective method of measuring the reaction levels associated with a notification.

For the LSIE system class we define the notion of *appropriate reaction* to capture the range of possible goals with respect to reaction. Sometimes a user may need to immediately perform some action as a result of the information in the LISE (high reaction). At other times, users may not need to do anything with the information (low reaction). For LSIEs then, the appropriate reaction depends on the use context and can vary from low to high. So, like interruption, the reaction parameter requires some flexibility.

Comprehension

Comprehension refers to the goal that the information in the notification be remembered and retained in long term memory [62]. Being able to recall and use information over extended periods of time are associated with the goal of high comprehension. For example, a user may want to know what the headlines are for a particular day. This knowledge could be used later to start a conversation. But, as with interruption and reaction, users may not always want to gain high comprehension from the notification. An example is with a fire alarm. People probably don't want to know what the cause of the fire is, or even the location, they only want to know that there is a fire and they need to evacuate the premises. So user goals vary with respect to how much comprehension of the information they want to attain.

Examples Some of the systems introduced in the earlier sections also deal with comprehension. Sideshow [12] provides information from multiple sources and this access to information helps with comprehension. Likewise, Irwin [58] provided information which aided comprehension.

There are few empirical works that investigate how to measure comprehension when dealing with notification systems. McCrickard et al. used correctness scores in [59]. Questions asked about general and specific information shown in notifications were used to measure comprehension. Similar techniques were used in [86], [84], and [87]. These measures provided useful insight into various information encoding techniques for comprehension support.

Other types of notification systems leverage comprehension as their main purpose or objective. Examples include work by Ishii [47] that deals with ambient media. Information displays such as water ripples projected on a ceiling and moving lights convey information about certain sources,

but the goal is to obtain some amount of comprehension of the information, not to be able to react to it, and definitely not to be interrupted by it. Another example of a system that provides high comprehension is Informative Art [76], with its depiction of weather as abstract artwork. High comprehension of the weather forecast is the main objective of this display.

For LSIEs, comprehension goals suggest a *high* rating is most accurate. Understanding and making sense of the information provided by the LSIE system is important for the users of the system. While this requirement can be less stringent in some instances, the typical comprehension requirements of LSIE systems is high.

It should be clear that notification systems are complex systems and need to be evaluated with respect to interruption, reaction, and comprehension. Researchers have touched on these ideas in existing studies but they have only recently been considered together [62] as critical parameters that define the notification system design space.

2.1.3 Applicability To This Work

Now that we have the IRC framework, we can begin systematic development of UEMs tailored to the user goals for specific system classes. This framework provides a common discussion and classification scheme for notification systems. Using this framework, we can identify systems that on the surface seem completely different, but with respect to user goals are actually quite similar. Identifying the design models (user goals) associated with these systems allows researchers to focus evaluation and probe issues that are important to the users of those systems [17]. Since our work seeks to understand evaluation methods for large screen information exhibits, this framework will provide a starting point for identifying heuristics for these systems, by classifying the target systems in terms of the critical parameters (see Chapter 4). To clarify, LSIE systems typically require high comprehension, self-defined interruption, and varying levels of reaction (depending on usage context), which would fit into the design space as a range across the right face of the cube (as opposed to just a corner). Figure 2.2 provides this depiction.

It is prudent at this point to describe how a system class can range across a face on the Notification Systems design space. The key to establishing a system class within the IRC framework is by restricting the parameters. For example, if a system tries to support high comprehension while simultaneously providing rapid reaction, but not eliciting user attention, that system would be classified as a "secondary display". In this instance we restrict comprehension, reaction, and interruption. For the LSIE system class we restrict both comprehension and interruption. At least two of the three parameters need to be restricted to establish a system class, otherwise the resulting design space would be addressing four or more combinations of user goals simultaneously, which would suggest creating a system that would be too complicated to function well for any of the tasks.

This also brings to light the notion of further categorizing the notification system classes into smaller chunks. Indeed, one wonders if there are different kinds of LSIE systems. As discussed in Chapter 6, there is an indication that the level of coupling between the primary and secondary tasks may differentiate some underlying difference in LSIE systems, but this notion is not further explored here. Future work could consider whether there may be refined critical parameters for each of the system classes in the notification systems design space (see Chapter 8).

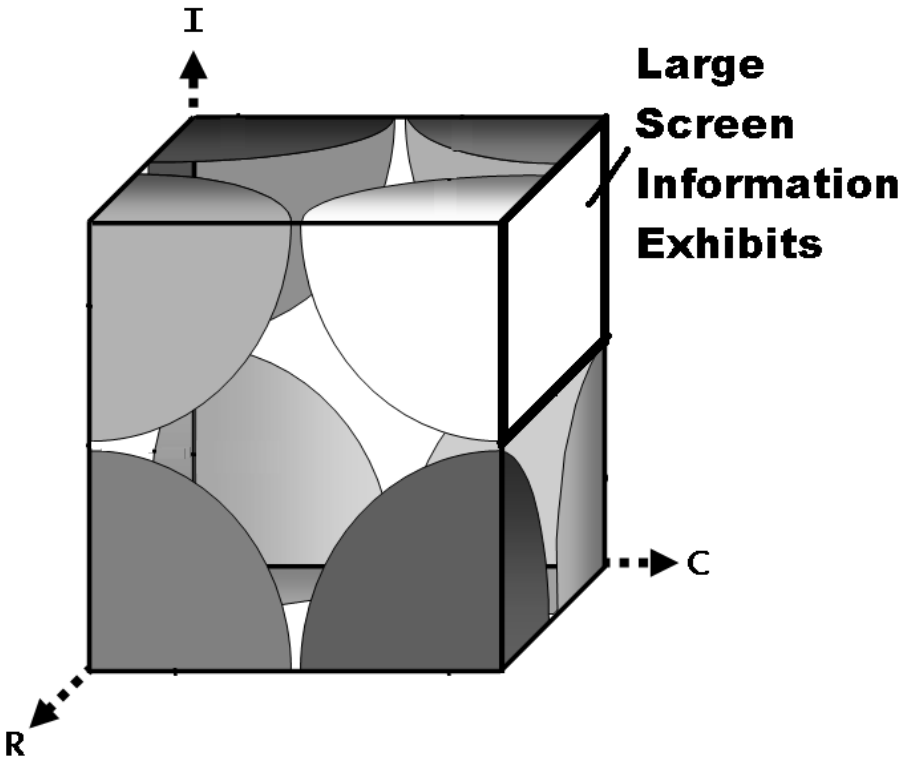

Figure 2.2: The LSIE system class within the notification systems design space.

2.2 Evaluation of Large Screen Information Exhibits

There has been little evidence of evaluations of large screen notification systems. Granted, some investigators have done limited user observations (as in [36] and [75]), but real empirical evaluations of whether the displays support their intended use are lacking. This is often due to difficulties in modeling the dual-task situation. Furthermore, these disjoint evaluations are difficult to leverage in design of other systems. The methodology and results are tailored explicitly for the individual systems tested, minimizing the chances for generalizing results for sharing and reuse.

One way to achieve generalizability is by effective evaluation of these systems based on desired user goals, or critical parameters. Developing generic evaluation methods based on critical parameters could promote reuse and generalizability of results [17]. A question then arises about which methods could be generalized for notification systems, specifically large screen information exhibits, based on the critical parameters for that system class. There are many evaluation methods that could be used, both analytical and empirical. We focus on analytic techniques in our work.

2.2.1 Analytical Methods

Analytical methods show great promise for ensuring formative evaluation is completed, and not just acknowledged in the software life cycle. These methods provide efficient and effective usability results [70]. The alternative usually involves costly user studies, which are difficult to perform, and increase the design phases for most interface development projects. It is for these reasons that we focus on analytical methods, specifically heuristics.

Heuristic methods are chosen in this research for two reasons. One, these methods are considered "discount" methods because they require minimal resources for the usability problems they uncover [70]. Two, these methods only require system mock-ups or screen shots for evaluation, which makes them desirable for formative evaluation. These are strong arguments for developing this method for application in multiple areas.

2.2.2 Heuristic Evaluation

A popular evaluation method, both in academia and industry is heuristic evaluation. Heuristics are simple, fast approaches to assessing usability [70]. Expert evaluators visually inspect an interface to determine problems related to a set of guidelines (heuristics). These experts identify problems based on whether or not the interface fails to adhere to a given heuristic. When there is a failure, there is typically a usability problem. Studies of heuristics have shown them to be effective (in terms of numbers of problems found) and efficient (in terms of cost to perform) [48, 50].

Some researchers have illustrated difficulties with heuristic evaluation. Cockton & Woolrych suggest that heuristics should be used less in evaluation, in favor of empirical evaluations involving users [21]. Their arguments revolve around discrepancies among different evaluators and the low number of major problems that are found through the technique. Gray & Salzman also point out this weakness in [32].

Despite these objections, heuristic evaluation methods, particularly Nielsen's, are still popular for their "discount" [70] approach to usability evaluation. Several recent works deal with adapting heuristic approaches to specified areas. Baker et al. report on adapting heuristic evaluation to groupware systems [5]. They show that applying heuristic evaluation methods to groupware systems is effective and efficient for formative usability evaluation. Mankoff et al. actually compare an adapted set of heuristics to Nielsen's original set [56]. They studied ambient displays (which are similar to the systems that would be classified as ambient displays in the IRC framework) with both sets of heuristics and determined that their adapted set is better suited to ambient displays.

This renewed interest in heuristic approaches is part of the motivation for this work, beyond the general need for evaluation methods for large screen information exhibits. As such, the heuristic usability evaluation method will be investigated in this research, but with different forms of heuristics, some adapted specifically to large screen information exhibits, others geared towards more general interface types (like generic notification systems or simply interfaces). **The focus of our work is to create a new set of heuristics by reliance on critical parameters. One that is tailored to the LSIE system class.**

2.2.3 Comparing UEMs

There is obvious interest in determining which, when, and how usability evaluation methods should be applied to certain types of systems [48, 50, 52, 11]. These types of evaluations of UEMs have sparked further discussion and debate [32]. Issues with current UEM comparison studies mainly revolve around lack of validity in the comparison [32]. Others have made counter arguments, suggesting that some comparison is better than none at all. Regardless of the back and forth arguments for various studies of UEMs, the HCI community recognizes the need for comparison and evaluation of UEMs in all areas. The lesson to take away from this discussion is to be careful and methodical in future UEM comparisons, striving for the highest validity in comparison studies.

Recent examples of work that strives to compare heuristic approaches to other UEMs (like lab-based user testing) include work shown at the 46th Annual Meeting of the Human Factors and Ergonomics Society. Chattratichart and Brodie report on a comparison study of heuristic methods [16]. They extended heuristic evaluation (based on Nielsen's) with a small set of content areas. These content areas served to focus the evaluation, thus producing more reliable results. It should also be noted that subjective opinions about the new method favored the original approach over the new approach. The added complexity of grouping problems into the content areas is the speculated cause of this finding [16].

Tan and Bishu compared heuristic evaluation to user testing [90]. They focused their work on web page evaluation and found that heuristic evaluation found more problems, but that the two techniques found different classes of problems. This means that these two methods are difficult to compare since the resulting problem lists are so different (like comparing apples to oranges). This difficulty in comparing analytical to empirical methods has been debated (see *Human Computer Interaction* 13(4) for a great summary of this debate) before and this particular work brings it to light in a more current example.

2.2.4 Comparing Heuristics

The approach in this work is to compare different types of heuristics, to illustrate the utility of a new set of heuristics targeted towards large screen information exhibits as compared to more general alternatives. By focusing on heuristics, any comparisons will be on similar output from the methods.

There has been some work on the best ways to compare UEMs. These studies are often limited to a specific area within HCI. For example, Lavery et al. compared heuristics and task analysis in the domain of software visualization [52]. Their work resulted in development of problem reports that facilitate comparison of problems found with different methods. Their comparisons relied on effectiveness, efficiency, and validity measures for each method.

Others have also pointed out that effectiveness, efficiency, and validity are desirable measures for comparing UEMs (beyond simple numbers of usability problems obtained through the method) [40, 21]. Hartson et al. further put forth thoroughness, validity, reliability, and downstream utility as measures for comparing usability evaluation methods [40].

These criteria (suggested by Hartson et al.) for comparison will serve useful in this research and will be adopted for several reasons. One, this work is recent, done in the last few years, meaning it is at or near the top of the list of current, accepted methods. Two, the comparison technique relies on multiple measures for each UEM, providing a more robust indication of the overall "goodness" of the UEM. And finally, the technique stresses the utilization of a usability problem set as the basis for comparing the methods. This makes the comparison rest on the real problems inherent in the systems used in the comparison study, thereby increasing the validity of the comparison [40]. **We developed a new UEM comparison approach that simplifies the calculation of these metrics.** Chapter 5 provides more detail on the comparison experiment done in this work.

The next section talks in some detail about large screen displays, some uses and evaluations of these displays, and how they may be used as notification systems. This discussion provides some insight into LSIE systems and motivates why we focus on them in this research.

2.3 Large Screen Displays

Large screen displays are much larger than typical computer desktops. They are most easily recognized as large flat panels attached to a computer, usually in a self-contained box unit. This discussion doesn't limit itself to only this type of large screen; in fact, any large surface that could be used as a display would fall into this category for discussion. There are some displays that are extremely large and would not be directly available for study (most sporting events arenas use very large screens to show interesting highlights). As such, the research proposed in this document deals exclusively with large screens designed for use inside buildings, but the results should generalize to these larger, outdoor displays.

Part of the motivation for focusing on large screen technologies comes from the fact that more and more institutions are purchasing large screens for use in workplaces. However, more often than not, institutions that own or possess large screen displays only use them sporadically, if at all. When they are used, it is typically for presentations or informal meeting support. These usage periods represent a small fraction of the time these displays could be used for notification tasks. Developing design recommendations and guidelines for using large screens as notification systems can help designers create systems that could utilize these displays during periods of typical non-use.

2.3.1 Early Forms

Early large screens came from the Xerox corporation in the early nineties. The Liveboard provided a large display surface along with some interesting software that allowed one to draw and annotate the programs being used on the display [29]. This new interaction technique would spark interest in the product and ultimately lead to its inclusion in numerous research institutions and businesses. These early large screens could be thought of as a desktop computer with a very large monitor.

Other early instances of large screen usage are reported in [44]. The Clearboard system was an instance of a large screen display that allowed users to share the space and work together while also maintaining eye contact. It used a mirroring technique to show the image of the other person correctly. Basically the display surface had a "see-through" layer that allowed the people working on the display to see each other and their gestures. Evaluations of Clearboard indicate that this technology supported awareness of other peoples' actions and activities with the display [45, 46].

Currently, SMART is a leader in large screen technology for large monitors. With integrated touch screen control these displays provide effective interaction without the use of pens or other devices. As such, interaction capabilities are increased and more and more universities and institutions are purchasing these boards [22]. However, with decreasing prices and development of newer technologies, other companies are beginning to develop their own systems.

Other Technologies

Other large screen display technologies include wrap around displays and projected surfaces. Wrap around displays could involve multiple monitor setups to provide extra display surface. Mary Czerwinski at Large Display User Experience group at Microsoft is investigating these types of display surfaces [26]. Projectors can also be used to project information on large surfaces such as walls or curtains. Bowden et al. illustrate this technique in their projection of "Jeremiah", a large

human-like face that reflects the status of its surroundings by facial expressions [10]. These types of displays are similar to the ones that will be focused on in this research, the large, box units described above, in that they provide extra display space away from the desktop.

Development of new and exciting display technologies has implications for large screen information exhibit use and development. Plasma screens are highly desirable for extended life and clarity in picture. Of course, cost makes these displays almost unattainable. Cheaper alternatives include LCD and OLED display technologies. LCD (liquid crystal display) screens are significantly cheaper to produce but they suffer from degraded picture quality and shorter life span. OLED (organic, light emitting diode) technology is still in its infancy and has not been successfully marketed as a viable large screen platform. However, businesses, institutions, and even individuals are intensely interested in purchasing larger and larger screens for use in communication and entertainment tasks.

Regardless of the possibilities for large screen platforms, this research is focused on the large screen software systems that run on these platforms. These software programs often allow information sharing and communication and do not necessarily rely on the underlying technology. As such, we are not concerned with the actual presentation medium in this research. Nonetheless, it is desirable to include information on this technology when discussing these types of systems. Limiting this research to these screens should not limit the results to this type of display. Any suggestions for evaluating information exhibits on "standard" large screens (i.e. those found running on dedicated large screen technologies) should readily apply to these other technologies.

2.3.2 Early Uses of Large Screen Displays

Predominantly these large screen displays have been used for meeting support. These displays provide rich capabilities for showing vivid images and color without the use of projecting equipment. They provide unique drawing surfaces for use during presentations and meetings. They are also used for informal meetings as extra desktop space. The Tivoli system [75] was one example of this type of system. It provided necessary interaction support for informal collaborative work meetings. The i-Land system also supported team work and collaboration [89]. It provided large screens for information sharing surfaces, as well as multiple input support for creativity and innovation.

Another example is the BlueBoard [78, 79]. This large screen system was mainly used for collaboration support for distributed colleagues. It was stationary and "knew" its place, so that it could provide relevant information to the occupants of a particular space. Interestingly, when the BlueBoard was not being actively used, it cyclically displayed web pages of information relevant to the location. This use actually makes the BlueBoard a notification system. The next section further investigates how large screen displays have been used as notification systems.

Not surprisingly, large screen displays are making their way into classrooms as well as meeting rooms. One early instance of a large screen in a classroom comes from Dufrense's implementation and usage of the Classtalk system [28]. A large screen was used to present the results of the student responses for the entire class to view and discuss. Another classroom system incorporated a large screen. The e-Class system had a large screen with a software program to allow the instructor to scribble notes on a scroll-able whiteboard [2, 3]. The large screen served as a note-taking space and specific notes on student comments were explicitly added during the lecture to provide comprehensive coverage of the course material.

Another example of large screens in the classroom is with the ClassroomBridge system [31].

This system provided both teachers and students with information regarding progress towards semester-long research projects. An LSIE display showed icons representing different types of work, as well as upcoming project deadlines. The idea was that the teachers and students, while busy completing projects and working with one another, could look at the display to assess and compare progress.

2.3.3 Shift Toward Notifications

In addition to typical dedicated uses, these large screen display systems are being used for tasks that are becoming more and more related to notification systems. Goals are shifting from presenting information as a primary task to a secondary task. One example is the WebAware system developed by Skog and Holmquist [82]. This system presented web page hits in a galaxy-like visualization on a wall in a common space. People could easily see what web sites were receiving the most traffic. This display existed only to provide the information to those who wanted it. The display was not the primary focus of any specific task, instead it existed to provide useful or interesting information in a secondary nature to other ongoing tasks.

Another example of a large screen display being used for information presentation is the Notification Collage by Saul Greenberg and Michael Rounding [36]. This system allowed users in a common area to post tidbits of information to the large screen. Others could then come by and see the postings and make new postings in response. Here again the display was not used as a primary activity but existed as a message information center, used only for secondary communication tasks.

Informative Art is yet another example of a large screen being used to present interesting information in a secondary fashion [76]. Large wall mounted displays provided information about weather in an aesthetically pleasing form. Users could look at the display during times of reflection and thought. Systems like WebAware, Notification Collage, and Informative Art illustrate the possibilities of using large screens as notification systems.

Each of the above systems can be classified using the IRC framework and the three critical parameters. If we examine the intended use of these systems, we see that they each strive to provide information to users. Users decide when to look at the displays (self-defined interruption) in hopes of gaining some useful understanding of the information (high comprehension), to perhaps perform some response (reaction).

2.4 Summary

Thus far background information on relevant related work has been presented. Discussion of evaluation methods and UEM comparison has illustrated the necessity for system-class level specific heuristics. This requires development of new heuristics tailored to the user goals associated with a system class. **Furthermore, critical parameters and the notification system design space help focus our work and provide underlying structure to our heuristic creation method** (as shown in Chapter 4).

In addition, we must take care in the comparison of our new heuristics with existing alternatives. Focusing on specific metrics, as proposed by Hartson et al. can provide much needed validity to our UEM comparison. We also are motivated to avoid the previous problems UEM researchers have encountered when conducting UEM comparisons. **To reduce ambiguity and**

increase validity, we have devised a new comparison technique that puts each of the target UEMs on equal ground, thus ensuring a fair comparison (see Chapter 5).

General discussion of notification systems and critical parameters, as well as information on LSIE systems, provides motivation for this work. Notification systems are rapidly gaining attention in all aspects of the HCI field. Large screen display technologies are rapidly approaching ubiquity in universities, industries, schools, offices, and in the home. **Providing developers and researchers with much needed evaluation tools can support the creation of effective, useful systems.**

The following chapter describes some early work that further motivates the need for a system-class level UEM for the LSIE system class. In addition, this background work illustrates the utility of Scenario Based Design [77] and claims analysis [15], which are used in the heuristic creation process.

Chapter 3

Background and Motivation

This chapter contains information on preliminary and background work done to identify requirements and needs for heuristics for large screen information exhibits, and to illustrate the utility of claims analysis in system inspection.[1] We will provide evidence of the potential that tailored heuristics provide as a usability evaluation method for large screen information exhibits, illustrate the utility of claims analysis for extracting design tradeoffs, and motivate the idea of structured heuristic creation.

3.1 Introduction

Recognizing the need for efficient evaluation of large screen information exhibits, a concentrated effort has been made to understand the best approach to address this need. The following sections provide information on a preliminary assessment of the utility in creating evaluation methods which are specific to a single system class, such as large screen information exhibits, yet are also generic enough to be applied across different systems in the same class. This background work provides a look at some LSIE systems, presents discussion of alternative heuristic creation approaches, and illustrates the need for a structured heuristic creation process.

As information presentation shifts from the desktop to ubiquitous displays (like a large screen), usability evaluation methods need to be tailored or newly developed to address pivotal user concerns and ensure quality software development. Ubiquitous systems, like LSIEs, bring new challenges to usability [1], mostly due to the nature of their multi-tasking use, in which attention is shared between ongoing tasks. Hence, keeping those challenges in mind can further evaluation method development efforts for large screen information exhibits.

However, there are many different types of usability evaluation methods one could employ to test design, and it is unclear which ones would serve as the best for this system class (large screen information exhibits). One important variation in methods is whether to use an interface-specific tool or a generic tool that applies to a broad class of systems. This preliminary study investigates tradeoffs of these two approaches (generic or specific) for evaluating LSIEs, by applying two types of evaluation to example LSIE systems. This work provides the motivation and direction for the creation, testing, and use of a new set of heuristics tailored to the LSIE system class.

[1] Parts of this chapter are published in [85].

3.2 Assessing Evaluation Methods

Specific evaluation tools are developed for a single application, and apply solely to the system being tested (we refer to this as a per-study basis). Many researchers use this approach, creating evaluation metrics, heuristics, or questionnaires tailored to the system in question (for example see [5, 56]). These tools seem advantageous because they provide fine grained insight into the target system, yielding detailed redesign solutions. However, filling immediate needs is costly—for each system to be tested a new evaluation method needs to be designed (by designers or evaluators), implemented, and used in the evaluation phase of software development.

In contrast, system-class evaluation tools are not tailored to a specific system and tend to focus on higher level, critical problem areas that might occur in systems within a common class. These methods are created once (by usability experts) and used many times in separate evaluations. They are desirable for allowing ready application, promoting comparison between different systems, benchmarking system performance measures, and recognizing long-term, multi-project development progress. However, using a system-class tool often means evaluators sacrifice focus on important interface details, since not all of the system aspects may be addressed by a generic tool. The appeal of system- class methods is apparent over a long-term period, namely through low cost and high benefit.

We conducted an experiment to determine the benefits of each approach in supporting a claims analysis, a key process within the scenario-based design approach [15, 77]. In a *claims analysis*, an evaluator makes claims about how important interface features will impact users. Claims can be expressed as tradeoffs, conveying upsides or downsides of interface aspects like supported or unsupported activities, use of metaphors, information design choices (use of color, audio, icons, etc.), or interaction design techniques (affordances, feedback, configuration options, etc.). These claims capture the psychological impacts specific design decisions may have on users.

3.3 Motivation from Prior Work

UEM research efforts have developed high level, generic evaluation procedures, a notable example being Nielsen's heuristics [70]. Heuristic evaluation has been embraced by practitioners because of its discount approach to assessing usability. With this approach (which involves identification of usability problems that fall into nine general and "most common problem areas"), 3-5 expert evaluators can uncover 70% of an interface's usability problems.

However, the drawbacks to this approach (and most generic approaches) are evident in the need to develop more specific versions of heuristics for particular classes of systems. For example, Mankoff et al. created a modified set of heuristics for ambient displays [56]. These displays differ from regular interfaces in that they often reside off the desktop, incorporating parts of the physical space in their design, hence necessitating a more specific approach to evaluation. They came up with the new set of heuristics by eliminating some from Nielsen's original set, modifying the remaining heuristics to reflect ambient wording, and then added five new heuristics [56]. **However, they do not report the criteria used in eliminating the original heuristics, the reasons for using the new wordings, or how they came up with the five new heuristics.** They proceeded to compare this new set of heuristics to Nielsen's original set and found the more specific heuristics provided better usability results.

Similar UEM work dealt with creating modified heuristics for groupware systems [5]. In this work, Baker et al. modified Nielsen's original set to more closely match the user goals and needs associated with groupware systems. They based their modification on prior groupware system models to provide guidance in modifying Nielsen's heuristics. The Locales Framework [35] and the mechanics of collaboration [38] helped Baker et al. in formulating their new heuristics. **However, they do not describe how these models helped them in their creation, nor how they were used.** From the comparison, they found the more application class-specific set of heuristics produced better results compared to the general set (Nielsen's).

Both of these studies suggest that system-class specific heuristics are more desirable for formative evaluation. However, the creation processes used in both are not adequately described. It seems that to obtain the new set of heuristics, all the researchers did was modify Nielsen's heuristics. Unfortunately, it is not clear how this modification occurred. Did the researchers base the changes on important user goals for the system, as determined through critical parameters for the system class? Or was the modification based on guesswork or simple "this seems important for this type of system" style logic? Based on what is provided in [56] we can assume that the latter was the case, as no mention of detailed inspection or analysis was provided. Baker et al. do provide some justification of their method. They modified pre-existing guidelines to form heuristics tailored for groupware applications. Unfortunately, specifics on how this transformation was done are lacking [5].

Based on these efforts, it is clear that a structured, repeatable heuristic creation method is necessary for development of system-class specific heuristics. However, there are specific processes required to ensure that the method can be repeated. To illustrate the utility of these processes, we performed an experiment that highlights both the analytic techniques for system inspection, as well as the need for a system-class level UEM tailored to the LSIE system class.

3.4 Experiment Description

These successes in creating evaluation tools that are specific to an application class represent new hope for human-computer interaction research — perhaps we can have the long-term comparison and benchmarking advantages with valuable, immediate feedback about interface usability problems. Therefore, as the field pursues UEM adaptation for large screen information exhibits, it is necessary to clarify the techniques that lead to effective UEM creation.

Our early work focused on evaluating LSIE systems through questionnaires [85], and compared single-system questionnaires to system-class questionnaires. Findings suggested that system-class questionnaires were the more desirable evaluation methods for the LSIE class. However, an important impact from this work involved the use of claims analysis [15] for assessing usability concerns. The following sections provide descriptions of the methods used in our earlier work, and support our decision to use claims analysis in our heuristic creation method (Chapter 4).

3.4.1 System Descriptions

We selected two interfaces within the large screen information exhibit application class for comparison in our earlier study [85]. Large screen information exhibits are software interfaces created for use on large display surfaces, providing interesting or useful everyday information to groups

or individuals in multi-use areas, such as meeting rooms, break rooms, and labs. These "off the desktop" interfaces provide context-aware access to deeper information about ongoing activities (high comprehension) in a format that allows users to decide when they want to look at the display (self-defined interruption) and supports necessary response to the information (reaction).

GAWK

The GAWK (Group Awareness, Work Knowledge) display was designed as part of the Virtual School [31] software suite to show student group work progress as icons within a timeline metaphor. As project groups complete work on documents and charts, icons appear in group rows. The systems cycles through newer icons, highlighting each and displaying a summary in the banner. This representation provides a history and current summary of the work done in each group, allowing teachers (and students) to better understand how they should help.

Photo News Board

The Photo News Board shows photos of recent news stories arranged by news type, allowing people who use common areas such as break rooms, labs, and meeting rooms with large screen displays to gain awareness of the day's news events [54]. Highlighted stories (photos) correspond with the text descriptions at the bottom. The system polls and retrieves photos and news clips from Internet sources, introducing newer stories in the center and constantly shifting older stories toward the edge. Highlighting patterns reflect the news category the occupants of the room are most interested in.

3.4.2 Methodology

We conducted an analysis of usability evaluation results on both systems to evaluate how well system-class or single-system surveys could support claims associated with these systems, lead to redesign conclusions, and impact long-term design processes. The overall methodology of this analysis consists of three phases: conducting the usability evaluations, assessing the claims analysis according to each result set from the usability evaluations, and recognizing potential long-term benefits.

Usability evaluations

We built several assumptions into our analytical approach that we believe to be typical of a usability study in the formative stages of system development. For instance, since participant time is quite costly, our evaluation sessions were designed to be completed within one-half hour. This made a controlled, lab-based test appealing, since we also wanted the feedback to be based on actual experience with the system rather than impressions from screenshots or storyboards. Therefore, we used scripted, rapid prototypes displayed on a 52" screen to illustrate how each system would support a real situation.

To conduct our testing, we used a 2 (system) x 2 (survey type) between-subjects experimental design. Twenty computer science undergraduate students participated in this experiment voluntarily. Participants were tested individually and asked to take on the role of a typical user for the

system they were evaluating. To do this, they performed other tasks (such as reading a newspaper or recording quiz grades) that would be part of the usage context (a classroom for the GAWK system and a break room for the Photo News Board). While the participant was engaged in these tasks, the interfaces presented scripted scenarios to familiarize the participants with the information presentation as it would actually be used in the intended situation. After experiencing each of several scenarios, the participant was asked simple, free-response questions about the information displayed by the interface, reinforcing their awareness of system features. However, the only recorded feedback was answers to a nine-question survey provided to the participant once all scenarios were completed.

The between-subjects design allowed both displays to be evaluated using two separate evaluation tools—a specific survey derived for each system that focused on important system features and a system-class survey based on the typical user goals for applications within the large screen information exhibit system class. System-class survey questions were loosely based on a framework for understanding user goals of notification systems [61] . The same system-class survey was used for both systems. See Appendix A for the survey questions.

To maintain consistency and usability study brevity, all three survey versions were developed within our research group and had nine questions. The surveys used Likert-style rating scales for various aspects of the systems. Participants read a statement and indicated their level of agreement with the statement, ranging from strongly disagree to strongly agree.

After aggregating responses for each survey, questions with ratings that clearly showed agreement or disagreement (average responses within one-standard error of the "neutral" response) were then applied to the claims analysis to determine the impact of participant responses on our claims.

Claims analysis assessment

To determine the impact of survey responses to understanding usability problems, we had to perform a claims analysis [15, 77] on each interface. Within the scenarios of use developed for each system, claims were made about the various design choices. These claims indicate how the design choices were thought to positively or negatively impact users. Claims analyses produced 58 design tradeoffs for GAWK and 56 for Photo News Board — each addressing system-specific claims based on activity design (e.g. supported or unsupported activities), information design (e.g. font/icon usage), and interaction design considerations. Examples of two categories of claims for each system are shown in Figure 3.1. Numbers of upside and downside tradeoffs by category can be seen in Table 3.1's left-most column for each system.

Next, survey questions from both the system-class and single-system surveys were mapped to each system's claims, although some claims were not addressed by questions on a given survey. This mapping was then used to determine whether or not claims were supported or refuted according to participant opinion. After capturing these numbers for the two types of evaluation tools, we compared how thoroughly the surveys addressed the claims analysis, gauging the impact of both survey tools on targeting immediate, per-study usability concerns and suggesting redesign conclusions.

Claims

Category	GAWK	PhotoNewsBoard
Supported Activities	(+) showing deadlines helps teachers focus students on tasks ***G9***, ***B4***	(+) seeing photos triggers curiosity about the story ***G9***, *A9*
Font/Icon Usage	(–) size constrains message length to ~80 characters, lack of detail causes distraction *G3, B8*	(–) smallest pictures on outer edges may not be recognizable *G6,* ***A3***

Survey Questions

G9: Appropriate reactions were obvious and intuitive.
A3: I could easily tell which news stories were recent and which were older.
B4: If I were busy with something, changes in the display would NOT distract me.

Figure 3.1: Example claims and survey questions, with upside (+) and downside (-) tradeoffs that correspond to sample questions from the system-class (G9) and single-system (A3, B4) surveys.

Recognizing long-term benefits

We compared system-class survey responses for both systems. We started by identifying questions that exhibit low response variance, since these could be candidate questions for benchmark establishment. Then, we looked for cases where the two systems demonstrated similar results (average response value and amount of response variance) on questions that map to similar design tradeoffs, allowing recognition of potential general guidelines that would be useful in designing new systems. We also looked for questions that had wide response variation, since the associated claims might allow detection of design artifacts that are responsible for the usability concern. Finally, we thought about how the two systems compared to each other. This allowed appraisal of the system-class survey's impact on long-term design processes — by suggesting guidelines, benchmarking response values, and allowing overall system comparison [85].

3.5 Discussion

This experiment investigated the tradeoffs associated with using single-system and system-class evaluation tools for large screen information exhibit systems — in terms of immediate, per-study contributions to the usability engineering process and impact to long-term design processes. These

	GAWK Claims		Supported or Refuted						PNB Claims		Supported or Refuted					
			Specific			Generic					Specific			Generic		
	+	-	of	S	R	of	S	R	+	-	of	S	R	of	S	R
Supported activities	5	1	6	5	0	5	5	0	6	2	8	3	0	7	3	0
Metaphors	3	3	6	1	0	4	2	1	3	2	5	2	0	4	2	1
Layout	5	4	8	2	3	8	3	2	6	3	8	2	1	7	1	1
Colors	6	2	7	5	0	8	3	2	4	0	3	3	0	0	0	0
Fonts/Icons	3	2	5	3	2	5	2	0	2	2	4	0	0	4	1	1
Audio	1	1	2	1	0	2	0	1	1	1	1	0	0	2	1	0
Animation	4	3	7	3	2	7	3	2	3	3	6	2	0	6	2	1
Affordance	1	2	3	1	1	2	1	0	1	1	1	0	0	0	0	0
Transition of states	4	2	6	2	2	6	3	1	5	3	5	0	0	5	1	0
Feedback	2	0	2	1	0	2	2	0	2	2	2	0	2	2	0	2
Config.	2	2	4	2	1	3	1	1	2	2	0	0	0	0	0	0
Subtotals	**36**	**22**	**56**	**26**	**11**	**52**	**25**	**10**	**35**	**21**	**43**	**12**	**3**	**37**	**11**	**6**
Totals	*58*		*37 (64%)*			*35 (60%)*			*56*		*15 (27%)*			*17 (30%)*		

a

b

Table 3.1: Survey result impact on claims analysis: numbers of claims are shown for claim analysis categories. Single-system surveys addressed slightly more claims (a), but the system-class survey supported/refuted similar percentages of claims (b).

tradeoffs highlight the need for system-class level UEMs. Furthermore, this experiment has illustrated two important considerations for the creation of heuristics based on critical parameters.

3.5.1 Drawbacks to Surveys

The comparison of redesign conclusions made available through each survey type did not show any advantage for either system-class or single-system evaluation tools, largely because the strong mapping between questions and claims provides a rich basis for analyzing design artifact usability performance. These findings provide no clear support for either type, suggesting no difference between the two tools for per-study usability evaluations. This means that the apparent advantages of the single- system method – addressing finer details of a design, as a result of tighter coupling with a claims analysis, to reveal better redesign options – did not manifest in this study. Reasons could be as simple as insufficient experimental conditions or could be as complex as individual interpretation of question wordings. **The important point is that we must provide support for system-class level UEMs.**

3.5.2 Strength of Claims Analysis

We note that the claim analysis process showed to be an extremely useful approach for supporting depth and breadth in usability problem identification, despite the relatively small amount of data, few users, rapid prototype systems, and brief session durations. This approach to usability evaluation provides direct feedback on design artifacts. By associating user responses to specific claims through the question-to-claims mappings, we were able to determine directed redesign conclusions from both surveys. It is this mapping that provides the redesign capability and insight into the usability of an interface, broadening the analytical scope afforded by each question. Using the claims analysis approach and assessing the coverage a UEM provides to a set of claims seems to complement newer UEM comparison methods (e.g. [40]).

From this study, we see that a system-class approach to large screen information exhibit usability evaluation seems like a logical choice. Hence, the long term benefits of these system-class methods (as opposed to a method created explicitly for a single system) suggest taking the initial cost to produce them, so that they may be reused in subsequent evaluations of new versions or other systems within the application class. As refinement of usability evaluation material for these types of systems proceeds, there is an impetus for carefully considering system-class tools that can be created by experts and leveraged by development teams for low-cost reuse and design knowledge collection. However, as pointed out in Section 3.3, we need a structured UEM creation process that produces usable and useful evaluation guidance.

3.6 Summary

The findings of our early study, which compared tailored, application-specific usability surveys to system-class surveys, can be summarized as follows:

- There is insufficient evidence that system-specific evaluation tools have an advantage over system-class tools in facilitating better identification of usability concerns or redesign strategies.
- We observed the potential long-term benefits of guideline and benchmark development, as well as system comparison in system-class evaluation tools.
- Claims analysis proved to be an extremely useful approach for producing problem sets in a consistent manner, which is necessary for validly evaluating UEMs [40].
- System-class evaluation tools for large screen information exhibits interfaces should be researched and developed by experts to provide development teams the benefits of low-cost reuse and design knowledge collection.

These findings suggest many directions for future work: improving upon the actual evaluation tools, extending our UEM comparison process with complementary, metric-centered techniques, investigating other evaluation methods, and drawing out the long-term benefits that are embedded in system-class specific approaches. Certainly, our evaluation tools can be improved upon. Our initial work can be extended with Hartson's equations [40], comparatively assessing UEM thoroughness, reliability, and downstream utility. In addition, this analytical process (claims analysis)

can be applied to other evaluation methods, such as heuristics, cognitive walkthroughs, and critical incident reports. As other systems are evaluated with system-class specific tools, it will be especially important to collect results in a cohesive manner that empowers formulation of benchmarks, guidelines, and other reusable design knowledge.

This background work suggests that system class level methods are most promising because they are at a desired level of specificity and generality, without being too much of either. The implications of this finding suggest that we create an evaluation method tailored to the large screen information exhibit system class, then compare it with other, accepted alternatives to illustrate method effectiveness. As noted in Section 2.2, heuristics are an excellent candidate for development as a formative evaluation tool for LSIE systems. The next chapter describes the creation method used to develop a set of heuristics tailored to the LSIE system class.

Chapter 4

Heuristic Creation

This chapter describes the creation process used for developing a set of usability heuristics for large screen information exhibits. The basic approach involves:

- inspecting example systems from the target system class, performing claims analysis from scenarios of use;
- classifying and categorizing these claims into manageable groups based on similarities;
- inspecting the wordings of the claims to extract design issues;
- and finally synthesizing heuristics from the issues

After providing some motivation and review, the following sections fully describe the processes used to arrive at the set of heuristics[1], and provide details on the final set.

4.1 Introduction

Ensuring usability is an ongoing challenge for software developers. Myriad testing techniques exist, leading to a trade-off between implementation cost and results effectiveness. Some methods are easier to administer, others perhaps are less costly. Finding and using the right method for a specific application is part of the usability process, but determining the most effective methods for a given application class is not clear.

Usability testing techniques are broken down into analytical and empirical types. Analytical methods involve inspection of the system, typically experts in the application field, who identify problems in a walkthrough process. Empirical methods leverage people who could be real users of the application in controlled tests of specific aspects of the system, often to determine efficiency in performing tasks with the system. Using either type has advantages and disadvantages, but practitioners typically have limited budgets for usability testing. Thus, they need to use techniques that give useful results while not requiring significant funds. Analytic methods fit this requirement more readily for formative evaluation stages. With the advent of new technologies and non-traditional interfaces, analytic techniques like heuristics hold the key to early and effective interface evaluation.

[1] A terse description of the process and the full listing of heuristics has been published in [88]

There are problems with using analytical methods (like heuristics) that can decrease the validity of results [21]. These problems come from applying a small set of guidelines to a wide range of systems, necessitating interpretation of evaluation results. This illustrates how generic guidelines are not readily applicable to all systems [40], and more specific heuristics are necessary. As we realize the potential in analytical evaluation techniques (namely cost effectiveness and early adoption and use), we have developed a set of heuristics tailored for evaluating large screen information exhibits. Our goal was to create a more specific set, tailored to this system class, yet still have a set that can be generic enough to apply to all systems in this class. This idea follows from what we learned in previous work on system-class evaluation methods (see Chapter 3).

Large screen information exhibits (LSIEs) are information presentation applications built to run on large screen displays. These displays can range from projections on walls to large electronic LED displays (like at sporting arenas), but are perhaps most easily recognized on situated large screens like the SMART board or Liveboard. These applications are part of a larger class of systems known as notification systems [62, 61]. Typically used to support secondary tasks, these notification systems are characterized through some common user goals revolving around dual- and multi-task situations.

LSIEs focus on very specific user goals based on the critical parameters of interruption, reaction, and comprehension. Differing levels of each parameter (high, medium, or low) define different system classes [62]. We focus on LSIEs which require medium interruption, low to high reaction, and high comprehension.

First, users want to gain a better understanding of the information presented on the display. This *high-level comprehension* involves making sense of the information and storing it in long term memory. All LSIE systems support the understanding of some information source through combinations of design artifacts like colors, layout, and groupings. The mapping of information meaning to design artifact provides the ability for increased understanding of the information source. By designating a *high* level for this parameter, we are requiring LSIE systems to support increased comprehension of the secondary information source, through storage in long term memory or relation to existing knowledge.

A second goal associated with LSIEs deals with minimizing the distraction caused by the display, while simultaneously allowing the user to decide when he/she wants to look at the information. This *self-defined interruption*, along with being shown on large screen displays, is what clearly separates these applications from other typical information interfaces. The self-defined aspect is important because users often need to stay focused on the primary task; only checking the LSIE when it is convenient. Self-defined interruption maps to a *medium* to *medium-high* level for the interruption parameter because a significant shift in attention is required. However, the user defines when this shift occurs. In other words, we do not want a *low* level of interruption, because we must shift our attention to the display. Also, we do not want a *high* level of interruption because we need to focus on our primary tasks, but we can more readily tolerate required shifts in attention. This suggests a medium to medium-high range.

A third goal, although somewhat more flexible than the other two, is to be able to react to the information. This *appropriate reaction* depends on usage context and personal goals, as some users may need to be able to make important decisions based on the information shown on the display in a quick and efficient manner (high reaction), while others may not need to do anything (low reaction). Users are busy with other tasks, such as editing documents, or searching through databases, and rely on these displays to facilitate awareness and understanding of the secondary information. As

such, the appropriate level for reaction can be different for various types of displays, ranging from *low* to *medium* to *high*.

Creating effective, useful applications for large screen displays is an important goal for developers. Effective evaluation methods, which can be readily implemented, are needed to ensure user goals are met early in the design life-cycle. Heuristics are a logical choice as an evaluation method for this system class because they can provide early design feedback with lower cost than empirical methods, but no heuristics specific to this class are available. This work seeks to create a set of heuristics, specifically targeting LSIEs, with the eventual goal of being able to allow efficient, accurate testing of formative designs. In so doing, we expected to learn how critical parameters can support heuristic creation, how to effectively compare different sets of heuristics, and how design knowledge can be reused in future LSIE development efforts.

4.2 Motivation

Tremendous effort has been devoted to the study of usability evaluation, specifically in comparing analytic to empirical methods. Nielsen's heuristics are probably the most notable set of analytical techniques, developed to facilitate formative usability testing [71, 70]. They have come under fire for their claims that heuristic evaluations are comparable to user testing, yet require fewer test subjects. Comparisons of user testing to heuristic evaluation are numerous [48, 50, 90], yet none seem to address the apparent lack of creation description. In other words, researchers seem focused on using, testing, and comparing heuristics, but few seem interested in how they are developed.

Some have worked to develop targeted heuristics for specific application types. Baker et al. report on adapting heuristic evaluation to groupware systems [5]. They show that applying heuristic evaluation methods to groupware systems is effective and efficient for formative usability evaluation. Mankoff et al. compare an adapted set of heuristics to Nielsen's original set [56]. They studied ambient displays with both sets of heuristics and determined that their adapted set is better suited to ambient displays.

However, as discussed in Chapter 3, these studies fail to provide readily applicable heuristic creation processes. Neither of these studies adequately describes the process used to arrive at the new heuristic sets. In the case of [5], they relied upon some previous theoretical underpinnings, but do not detail how to move from theory to heuristics. In the case of [56], they simply rely on expert experience to suggest plausible new heuristics. In both cases the new sets are based heavily on Nielsen's original set [71] and the processes are not clear to someone who may try to use them in their own efforts.

These previous works illustrate the interest and need for effective heuristics; furthermore, it illustrates the desire to create evaluation methods that are effective for specific types of interfaces. However, these works do not specify exactly how one can create heuristics for an application class (as discussed in Section 3.3). The following section describes how we approached this problem.

4.3 Processes Involved

How does one create a set of heuristics anyway? We could follow the steps of previous researchers and just use pre-existing heuristics, then reason about the target system class, hopefully coming up

with a list of new heuristics that prove useful. There is little structure to this approach, and it is highly dependent upon the individuals involved in the analysis. In fact, in the original published work that describes the heuristics, Nielsen and Molich explicitly state that the heuristics come from years of experience and reflection. Not surprising as the heuristics emerged some 30 years after graphical interfaces became mainstream. In the case of Nielsen and Mack, they at least validated their method through using it in the analysis of several systems, after they had created their set. But this approach is not feasible for most system classes, mainly because the systems are new and have received little evaluation attention, and the time required to amass necessary experience would be unacceptable for current and near term development efforts. Because these systems are so new, targeted evaluation methods become even more desirable and necessary to ensure early formative feedback.

Other types of heuristics have not seen this level of use and validation but still show promise for usability. However, one issue with these other sets is the approach behind their creation [56, 5]. In particular, these two studies fail to provide repeatable, structured creation methods that can be readily applied in other areas. Indeed, the two mentioned studies relied upon vague descriptions of theoretical underpinnings [5] or simple tweaking of existing heuristics [56]. Researchers struggle to come up with methods to obtain usable heuristics in their particular domains, and the approaches described in [5] and [56] do not provide a clear, structured approach to heuristic creation.

Our approach to this lack of structure in creating heuristics is to take a logical look at how one might uncover or discover heuristics for a particular type of system. Basically, to gain insight about a certain type of system, one could analyze several example applications in that system class based on the critical parameters for that system class, and then use the results of that analysis to categorize and group the issues discovered into re-usable design guidelines or heuristics.

This sounds simple but in reality takes a concentrated effort in several stages. These stages involve:

- **selection of target systems.**
- **inspection of these systems.** An approach like claims analysis [15] provides necessary structure to knowledge extraction and provides a consistent representation.
- **classifying design implications.** Leveraging the underlying critical parameters can help organize the claims found in terms of impacts to those parameters.
- **categorizing design implications.** Scenario Based Design [77] provides a mechanism for categorizing design knowledge into manageable parts.
- **extracting high level design guidance.** Based on the groupings developed in the previous step, high level design guidelines can be formulated in terms of design issues.
- **synthesizing potential heuristics.** By matching and relating similar issues, heuristics can be synthesized.

The following sections describe in detail the process used in this work to create a set of heuristics tailored to the LSIE notification system subclass.

4.4 Selecting Systems

The first step in the creation process requires careful selection of example systems to inspect and analyze for uncovering existing problems in the systems. The idea is to uncover typical issues inherent in that specific type of system. These issues are the keys to design guidance and information re-use in that knowing about them and mitigating them can help future designers create better systems.

Our goal was to use a representative set of systems from the LSIE class. Because these types of systems are relatively new, our selection space is limited. We wanted systems that had been in use for a while, with reports on usage or studies on usability to help validate the analysis we would perform on the systems. Given these constraints, we chose the following five LSIE systems, including some from our own work and some from other well-documented design efforts, to further investigate in the creation process.

- GAWK [31] This system provides teachers and students an overview and history of current project work by group and time, on a public display in the classroom.
- Photo News Board [85] This system provides photos of news stories in four categories, shown on a large display in a break room or lab.
- Notification Collage [36] This system provides users with communication information and various data from others in the shared space on a large screen.
- What's Happening? [94, 95] This system shows relevant information (news, traffic, weather) to members of a local group on a large, wall display.
- BlueBoard [78] This system allows members in a local setting to view information pages about what is occurring in their location (research projects, meetings, events).

These five systems were chosen as a representative set of large screen information exhibits. The GAWK and Photo News Board were created in local labs and thus we have access to the developers and potential user classes. The other three are some of the more famous and familiar ones found in recent literature. These five systems also clearly illustrate the unique user goals associated with the LSIE system class. The following sections provide some more details on these systems and discussion of the user goals associated with them.

4.4.1 Are these LSIEs?

Each of the systems we chose to include in our heuristic creation process are considered LSIEs. The following sections provide some details on the systems and why they are classified as LSIE systems. To summarize the following descriptions of the target systems, we have provided Table 4.1, listing the systems and the user goals associated with them according to the LSIE system class. Recall that we have defined LSIEs to be notification systems that support the user goals of self-defined interruption (medium to medium-high interruption), high comprehension, and appropriate reaction (low to medium to high reaction). Determining the goals of each system is accomplished through assessment of typical usage and system intent, as stated by the system developers. In describing these systems, we provide justification for classification in the LSIE system class.

System	*Interruption*	*Comprehension*	*Reaction*
GAWK	medium-high	high	high
Photo News Board	medium	high	low
Notification Collage	medium	high	low to high
What's Happening?	medium-high	high	high
Blue Board	medium	high	low to high

Table 4.1: Target systems and user goals. Multiple entries come from different scenario parameter values.

4.4.2 Systems

GAWK

The GAWK system was designed as part of the ClassroomBridge [31] software suit to provide middle school science teachers with extra awareness information about the student groups in their classes. Specifically, the GAWK is the name of the large screen awareness application used in the classrooms. Students in separate classrooms work together to complete long term projects in which they must collaborate on and share the work. Teachers need to stay on top of how the groups are performing, and the GAWK display provides this information in real time, in the classroom. Icons representing various documents are displayed on the large screen, arranged by group and week. This provides the teachers (and students) with an overall view of on what the groups worked and when.

As project groups complete work on documents and charts, icons appear in group rows, representing their work activity. These rows are further broken down into 6th and 8th grade work by showing specific icons on the top or bottom of the row. The system cycles through recent work, highlighting the icon and previous versions of the work, and displaying a summary in the banner. This representation provides a history and current summary of the work done in each group, allowing teachers (and students) to better understand how groups have worked over time.

Facilities for collaborative editing, real time chat, and document sharing are included among the many tools in the desktop software. The GAWK was designed to support awareness of each groups' work in the science projects the students complete as part of class work. Work artifacts (documents, charts, pictures) are represented as icons, distributed over time. This allows both teacher and students to assess how well groups compare in progress towards specific project goals.

User Goals The intended use of this system was to support activity awareness [14] by providing a work history on the large screen in the classroom. Users (both students and teachers) would be able to look over at the display and assess how well groups are making progress towards goals. Understanding the current state of the project, as well as the history of the work is part of this awareness. Presence and absence of work icons reflects when work was completed. This information supports the *high comprehension* goal associated with this display.

The display was designed to support gaining high comprehension without requiring significant user attention (medium interruption). To this end, users should be able to decide when to look at the display, as opposed to the display aggressively grabbing their attention (not high interruption), in order to maximize efficiency with their primary tasks and still maintain an understanding of group work status. This falls in line with the notion of *self-defined interruption* mentioned earlier.

This particular display supports various types of reactions to the information; from a teacher deciding to intervene with a group, to students deciding to increase work effort to accomplish goals (perhaps they recognized they may be behind other groups). These are some of the *appropriate reactions* for this display in this particular context. Scenarios describing typical usage can be found in Appendix B.1.

Photo News Board

The Photo News Board is a news dissemination system shown on a large screen display in break rooms or labs. Community members associated with these places can quickly and easily view the news stories from the day, and engage in conversations with other community members about the content on the display. News items from World News, Top Stories, Sports, and Entertainment are shown in a radial grid pattern. These items are represented with cropped photos, arranged by time. Recent additions to the display are added at the center, and older photos are shifted towards the outer edges. This time metaphor is reinforced by size as well — larger photos are closer to the center of the display, and older photos are smaller, nearer the edges.

A highlighting technique is used to provide textual information about a photo in a banner at the bottom of the screen. When a story is highlighted, the corresponding text blurb appears in the banner. The highlighting reflects the news category the occupants of the room are most interested in. It moves from photo to photo within a category or across categories if two or more categories receive similar interest rankings. This highlighting can provide meta-information about the users of the system, in addition to the current news happenings.

User Goals Typical users of the Photo News Board seek comprehension of the recent news events in world news, sports, entertainment, and the top stories. This directly translates to a *high comprehension* need. Since these users are busy with other tasks (editing documents, etc), this display is only glanced at when the user wants to see any new stories or if they happen to look up from their work and an image catches their attention (medium interruption). This corresponds with a *self-defined interruption* requirement. Usually, the users will not need to react to seeing a specific news story, hence there is a *low reaction* requirement, but they may sometimes strike up conversations with others based on the content. Scenarios describing typical usage can be found in Appendix B.2.

Notification Collage

The Notification Collage is an information sharing system to allow lab members to post various interesting content to a large public display. Here others can view the information, post new items, or comment about the current content. This particular system has an associated desktop component, or private view. Data can be made public or private in a conscious choice by the user.

Users can share almost anything, from simple text to screenshots. The data files are sent to the Notification Collage and are randomly placed on the screen. Newer items can partially or completely cover other items. This random, haphazard layout is intentionally designed to stress the collage metaphor. Users can post comments in Post-It style notes to give some feedback on specific items.

The whole purpose of the Notification Collage is to increase the awareness of the people in a local lab setting. The idea is based on the theory that if people know what is going on with others in a small setting, those people would be more likely to talk about their activities with each other. This display attempts to achieve this goal through static, omnipresent work artifacts shown on a large public display.

User Goals As mentioned, typical users seek information about their colleagues and current work activities in the lab or organization. The display provides this information without aggressively seeking attention (not high interruption). Users can survey the screen, assessing what is going on in their environment, without their attention being drawn explicitly to the display, however they must shift their attention to the display to gain understanding (medium interruption). This falls in line with the idea of *self-defined interruption*. The display also provides the users with the information they are seeking, hence it helps them gain *higher comprehension* of the current work efforts in the group. Finally, the display shows the information to the users so that they can decide what actions are necessary, whether it is to immediately perform an action (high reaction) or do nothing (low reaction), thus supporting the goal of *appropriate reaction*. Scenarios describing typical usage of the Notification Collage can be found in Appendix B.3.

What's Happening?

What's Happening? provides interesting, useful information to users by showing relevant pictures and text on a large display in a busy hallway or common area. Information about traffic, weather, news, and local events is displayed for users to quickly and easily see, and hopefully assimilate the important aspects. The display automatically cycles through the information pages and users can look at the display when they feel the need or desire to do so.

User Goals This display provides its users with useful, desired information on the screen in large images (medium interruption) without being overly intrusive (not high interruption), thus satisfying the need for *self-defined interruption*. Also, the display supports *high comprehension* by providing users with the information they need and want. Finally, the display allows users to perform any required actions at opportune times (high reaction), supporting the *appropriate reaction* requirement. Typical usage scenarios can be found in Appendix B.4.

Blue Board

Blue Board is similar to What's Happening? in that it provides information to users in a common area, but the information is more specifically tailored to the location in which the board is situated. Specifically, users can view information pages related to the area in which the board is located. Typical information includes special events, pertinent announcements, local traffic reports, local weather, and other information. See Figure 4.1 for a screenshot. In addition to its large screen

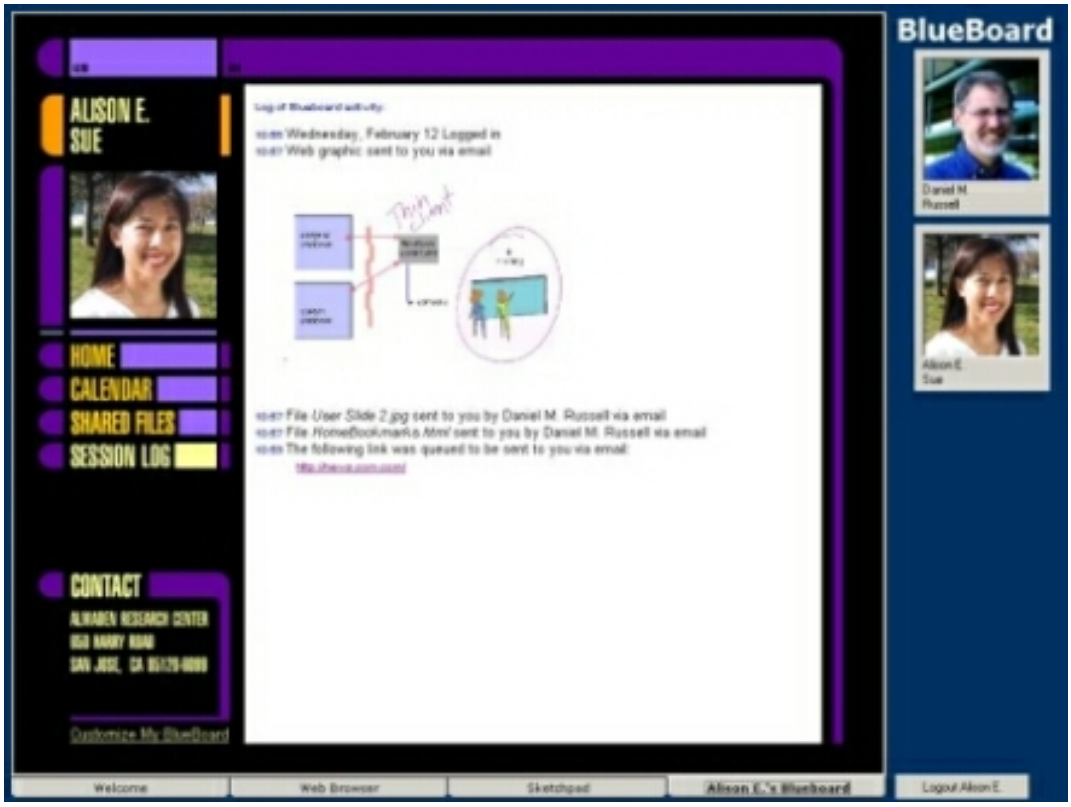

Figure 4.1: Blue Board. Attract loop shows users information about the current environment. ©2003 - IBM. Printed here with permission.

information exhibit characteristics, it can also present information about individual users within the organization. This information is accessed by users and allows them to exchange and share information through the large screen display. However, this particular usage would not be classified as an information exhibit, it would be more like a typical application in that case.

User Goals The Blue Board provides users in a local setting interesting and useful information about the local environment. Traffic reports, local happenings, weather information, and other information types are shown in a cyclic manner (medium interruption) on the large screen in an open area of the organization, providing *high comprehension*. Passers-by and others can gain an understanding of these events by glancing at the display. Additionally, users do not need to spend significant time (not high interruption) viewing the display as they are busy writing, attending meetings, and doing other tasks, supporting the notion of *self-defined interruption*. Some content may elicit specific responses (high reaction) but in general users do not need to respond (low reaction) to the items they see on the Blue Board, thus it supports *appropriate reaction*. Scenarios for the Blue Board can be found in Appendix B.5.

4.5 Analyzing Systems

Now that we have selected our target systems, we must now determine the typical usability issues and problems inherent in these systems. Performing usability analysis or testing of these systems finds the issues and problems each system holds. To find usability problems we can do analytic or empirical investigations, recording the issues we find.

We chose to use an analytic evaluation approach to the five aforementioned LSIEs, based on arguments from Section 3.3. We wanted to uncover as many usability concerns as possible, so we chose claims analysis [15, 77] as the analytic vehicle with which we investigated our systems. This depends on multiple scenarios describing each system. This method proved useful in early investigations into the generality vs. specificity question (see Chapter 3), so we are using it here as well.

4.5.1 Claims Analysis

Claims analysis is a method for determining the impacts design decisions have on user goals for a specific piece of software [15, 77]. *Claims* are statements about a design element reflecting a positive or negative effect resulting from using the design element in a system [15]. For example, if an interface used a form of blinking text, a possible claim could be:

```
Using blinking text can:
  + direct users' attention to important information
  BUT might also distract users from other tasks
```

Claims analysis involves inspection and reflection on the *wordings* of specific claims to determine the psychological *impacts* a design artifact may have on a user [15]. The wordings are the actual words used to describe positive and negative effects of the claims. The impacts are the overall psychological effect on the user. From the wordings and impacts, one can determine how a particular claim might effect the user goals associated with large screen information exhibits. For example, if we inspect the example claim from above, we can see that there is an impact on interruption, reaction, and comprehension. Why? By directing a users attention to important information, we are improving the user's ability to react to the information, possibly increasing comprehension as well. However, we also see that it could cause interruption to other tasks, which may or may not be desired.

Using claims, we can analyze design choices in terms of user goal impact through critical parameters; revealing groups or classes of problems that share similar characteristics. These problem classes form the basis for formulating higher level heuristics, which encompass several detail driven problems. By synthesizing many problems into fewer, high-level heuristics, practitioners and researchers are better able to apply the knowledge learned in our inspection of these five systems in their own design projects.

This analysis technique can uncover underlying problems with an entire system class, like information exhibits. This technique also allows for knowledge building and re-use based on looking at several example systems and performing claims analysis on them. Identifying poor design elements as well as good design elements can further the development cycle for large screen information exhibits.

System	# Claims
GAWK	58
Photo News Board	56
Notification Collage	48
What's Happening?	41
Blue Board	50
Total	**253**
Average	**51**

Table 4.2: Numbers of claims found through inspection of five systems. Claim numbers are listed for each system

4.5.2 System Claims

Claims were made for each of the five systems that were inspected. These claims focused on design artifacts and overall goals of the systems. These claims are based on typical usage, as exemplified by the scenarios shown for each system. On average, there were over 50 claims made per system. Table 4.2 shows the breakdown of the numbers of claims found for each system. Each claim dealt with some design element in the interface, showing upsides or downsides resulting from a particular design choice. These claims can be thought of as *problem indicators*, unveiling potential problems with the system being able to support the user goals. These problem indicators include positive aspects of design choices as well. By including the good with the bad, we gain fuller understanding of the underlying design issues. Claim wordings indicate immediate classification into whether the issue holds a positive or negative impact on the user goal. See Appendix C for complete listing of all claims found for each system. Understanding these problem indicators and assessing their impact on interruption, reaction, and comprehension is a key to developing heuristics for large screen information exhibits. By leveraging real issues from real world systems, we can determine the immediate problems that surround current implementations of large screen information exhibits.

4.5.3 Validating Claims

How do we know that the claims we found through our analysis represent the "real" design challenges in the systems? This is a fair question and one that must be addressed. We need to verify that the claims we are using to extract design guidance for LSIE systems are actually representative of real user problems encountered during use of those systems. We tackled this problem through several different techniques. For the GAWK and Photo News Board, we relied upon existing empirical studies [85] to validate the claims we found for those systems. The earlier studies actually contributed to the claims analysis and served as validation of those claims for these two systems, so we feel confident in the claims used from those systems.

For the Notification Collage we relied upon discussion and feedback from the system developers. We sent the list of claims and scenarios to Saul Greenberg and Michael Rounding and asked them to verify that the claims we made for the Notification Collage were typical of what they observed users actually doing with the system. Michael Rounding provided a thorough response that indicated most of the claims were indeed correct and experienced by real users of the system.

There was one claim that he said was not observed in real users:

> - lack of organization frustrates users when trying to look for an artifact [response] I don't know if I've ever observed this. More often than not, people will ask a question like "did you see x website that y posted to the NC? You should check it out!" This happens almost daily. [personal communication, 8/27/2003]

However, he did not specify that the claim was not correct, so we kept it in our analysis.

A similar effort was attempted with both the What's Happening? and Blue Board systems. The developers of these systems were contacted but no specific feedback was provided on our claims. However, John Stasko, co-developer of the What's Happening? system, provided interview feedback on the system and provided a nice publication [95] that served as validation material for the claims. This report provides details on user experiments done with the What's Happening? system. Using this report, we were able to verify that most of the claims we made for the system were experienced in those experiments. We decided to keep the claims that were not refuted in that report in our analysis, since the extra information would not reduce potential design guidance.

Unfortunately, none of the developers of the Blue Board system responded to our request. We were able to use existing literature on the system to verify some of the claims but the reports on user behavior in [78] did not provide enough material to validate all of the claims we found for that system.

Thus, we have empirical evidence coupled with developer feedback to validate the claims found in our system inspection. This validation is important because we want to extract design guidance from the "real" issues with LSIE systems. While these efforts did not validate every single claim in our analysis, we have support for the overwhelming majority of them, and those without validation could contain important design knowledge. Thus we elected to keep all of the claims that were not refuted through our validation efforts in completing the remainder of the creation effort.

4.6 Categorizing Claims

Now that we have analyzed several systems in the LSIE class, and we have over 250 claims about design decisions for those systems, how do we make sense of it all and glean reusable design guidance in the form of heuristics? To make sense of the claims we have, we need to group and categorize similar claims. This will allow us to more fully understand the underlying issues that appear across the five systems we have been studying. This requires a framework to ensure consistent classification and facilitate final heuristic synthesis from the classification. This is where the idea of critical parameters plays an important role, and how focusing on scenarios of use can support categorizing the claims.

4.6.1 Classifying Claims Using the IRC Framework

Recall that notification systems can be classified by their level of impact on interruption, reaction, and comprehension [62]. This classification scheme can be simplified to reflect a *high*, *medium*, or *low* impact to each of interruption, reaction, and comprehension. Furthermore, this classification can be applied to the claims we have from our earlier inspections of these systems.

Claim	**IRC classification**
+ fading banner *minimizes distraction*	low interruption
+ showing movement of pictures when new items arrive *facilitates recognition* of new items	high reaction
- flashing with highlighting may be *confusing*	low comprehension

Table 4.3: Example classification of claims with keywords in italics. The resulting classification is provided in the right column. The italicized keywords suggest the correct classification (high, medium, or low).

In other words, we can take a single claim and classify it according to the impact it would have on the user goals associated with the system. For example, we have a claim about the collage metaphor from the Notification Collage system that suggests that the lack of organization can hinder efforts to find information. This claim would be classified as "high" interruption because it increases the time required to find a piece of information. It could also be classified as "low" comprehension because it reduces a person's ability to understand the information quickly and accurately. It is perfectly acceptable to have the claim fit into both classifications. Note how this claim fits in with the definition of the LSIE system goals. The following describes the mechanism used to perform this classification.

4.6.2 Assessing Goal Impact

Determining the impact a claim has on the user goals was done through inspection and reflection techniques. Each claim was read and approached from the scenarios for the system, trying to identify if the claim had an impact on the user goals. A claim impacted a user goal if it was determined through the wording of the claim that one of interruption, reaction, or comprehension was modified by the design element. Since each claim has the potential to impact interruption, reaction, and comprehension, care must be taken in determining what those impacts are. The inspection process used here does this through discussion and reference back to the scenarios for the target system when there is disagreement among evaluators.

The wordings of the claims often contain catch phrases or key words that indicate the appropriate user goal that is impacted by the claim. Example keywords include: distraction, understanding, decision, notice, know, and awareness. It is not difficult to determine to which of the three user goals these keywords pertain (distraction maps to interruption, understanding maps to comprehension, notice maps to reaction). Example catch phrases include: "focuses attention", "increases understanding", "recognize photos", etc. Again, matching the phrases to user goals is not difficult (focuses attention maps to high interruption, increases understanding maps to high comprehension). Table 4.3 provides example claims and the keywords contained in them.

To assign user goal impacts to the claims, a team of experts should assess each claim. These

experts should have extensive knowledge of the system class, and the critical parameters that define that class. Knowledge of claims analysis techniques and/or usability evaluation are highly recommended. We used a two- person team of experts. These experts have extensive knowledge of the IRC framework [62] and of claims analysis techniques. We had each of the two experts provided his/her classification for each claim separately. Differences occurred when these classifications were not compatible. Agreement was measured as the number of claims with the same classification divided by the total number of claims. We found that initial agreement on the claims was near 94% and after discussion was 100% for all claims. This calculation comes from the fact that out of 253 individual claims, 237 were classified by the inspectors as impacting the user goals in the same way, i.e. all of the experts agreed on the same classification. Differences in the remaining 16 claims consisted mainly of an inspector having chosen one factor out of multiple factors as the dominant classification. For example, the claim "banner suggests late breaking topics and adds excitement" was rated as both "high" interruption and "high" reaction. In cases like this, discussion about reasons for choosing one classification over another led to total agreement among the inspectors on the final classification. It is important that all evaluators agree on the final classifications for all claims, so that in later stages, these earlier disagreements do not cause problems.

Table 4.3 provides some example claims and their resulting classification. In reality, the classification differences were even smaller because it was deemed acceptable for a claim to have multiple goal impacts. In cases where there was disagreement, discussion was necessary to ensure that the inspectors all understood the reasoning behind the classification. The full classification of all claims can be found in Appendix F.1.

4.6.3 Categorization Through Scenario Based Design

Categorization is needed to separate the claims into manageable groups. By focusing on related claims, similar design tradeoffs can be considered together. An interface design methodology is useful because these approaches often provide a built-in structure that facilitates claims categorization. Possible design methodologies include Scenario Based Design [77], User Centered Design [73], and Norman's Stages of Action [72].

Scenario based design (SBD)[77] is an interface design methodology that relies on scenarios about typical usage of a target system. This system can be a conceptual design or, as in this case, a pre-existing system. The scenarios capture typical usage context and illustrate user goals. This is useful in analyzing and understanding the typical user interactions with a system, and how specific factors can impact the utility of the system for the user. For system analysis, this is important for determining possible functionality needs, as well as identifying specific usage settings and restrictions resulting from those settings.

We chose SBD to use in this work for several reasons. First, it is a simple framework consisting of activity, information, and interaction design. This framework is a simplification of Norman's seven stages of action [72]. So instead of focusing on seven different categories, we can focus on three. Secondly, the framework provides nice sub-categories for each of activity, information, and interaction design; further supporting the structured creation effort. Finally, we chose SBD because it is tightly coupled to claims analysis [15], which we found in earlier work ([85] and Chapter 3) to be excellent for assessing system usability.

Scenarios describing users and their interaction with a system are at the heart of SBD [77]. By focusing on believable stories, insight into the target system is achieved. While intended to guide

design for new systems, SBD captures typical interface usage within a simple framework that can be applied to analysis of existing systems. For example, to better understand how large screen information exhibits are used, scenarios were created for them, to illustrate a typical user and their interaction with the system. Literature on the five systems and discussions with developers helped in the creation of the scenarios. These scenarios were then analyzed to identify claims relating to activity, information, and interaction design choices. Appendix B contains three scenarios for each of the systems used in this creation process.

We used the scenarios of the systems to feed our analysis in uncovering claims about the design decisions for the five LSIEs. Identifying the effects on goals is an important part of how we used SBD for guiding the creation of heuristics for large screen information exhibits. To more fully understand how this was accomplished, one must understand the framework suggested by SBD. This framework includes activity, information, and interaction design.

Activity Design

Activity design involves what users can and cannot accomplish with the system, at a high level [77]. These are the tasks that the interface supports, ones that the users would otherwise not be able to accomplish. Scenarios are excellent methods for identifying activities for a given interface because they illustrate what users can do with a system. Understanding what activities are possible with an existing system can help identify problems with how the system was designed.

Activity design encompasses metaphors and supported/unsupported activities [77]. Understanding and identifying the presence and strength of metaphors is one half of activity design. Metaphors can help users comprehend an interface and identify the ways in which it may or may not be used. Realizing and understanding exactly what tasks can be completed with an interface is the other half of activity design. This also involves identifying those tasks that are not supported by the interface.

Metaphors and supported/unsupported activities can directly impact user goals related to large screen information exhibits. A suitable metaphor can increase comprehension and reduce the amount of unwanted interruption, thereby creating a more effective system. Whereas a poorly chosen metaphor can increase the time it takes to learn the interface, decreasing comprehension, and increasing unwanted interruption.

Information Design

Information design deals with how information is shown and how the interface looks [77]. Design decisions for information presentation directly impact comprehension, as well as interruption. Identifying the impacts of information design decisions on user goals can lead to effective design guidelines. Furthermore, effective information design can allow users to react appropriately when necessary.

Information design is an enormous area with many different facets. This category must be broken down into smaller, more identifiable parts. We chose to use the following sub-categories for refining the information design category: use of screen space, foreground and background colors, use of fonts, use of audio, use of animation, and layout. These sub-categories were chosen because they cover almost all of the design issues relevant to information design [77].

Interaction Design

Interaction design focuses on how a user would interact with a system (clicking, typing, etc) [77]. This includes recognizing affordances, understanding the behavior of interface controls, knowing the expected transitions of states in the interface, support for error recovery and undo operations, feedback about task goals, and configurability options for different user classes [77]. Interestingly, interaction with large screen information exhibits is minimal, thus the impacts to user goals would be minimal for most aspects of interaction design. This is discussed in more detail in a Section 4.7.1.

Categorization

Armed with the above categories, we are now able to group individual claims into an organized structure, thereby facilitating further analysis and reuse. So how do we know in which area a particular claim should go? This again is done through group analysis and discussion regarding the wording of the claim. The claim wordings typically indicate which category of SBD applies, and any disagreements can be handled through discussion and mitigation.

Similar to the classification effort, this categorization process relied upon the claim wordings for correct placement within the SBD categories. The sub- categories for each of activity, information, and interaction provide 14 areas in which claims may be placed. Typical keywords that placed a claim within the activity design category revolve around descriptions of metaphors and user tasks. Other claim wordings suggested other placements, within either activity, information, or interaction design categories. Table 4.4 provides some example claims and their resulting categorization. Keywords in the claim suggest the categorization choice. As an example, consider the claim about the banner adding excitement, we can see that it would fall in the activity design category, particularly within the "metaphors" sub-category. Why? Because the banner is an instance of a type of information sharing mechanism that people are familiar with from other areas (television, billboards, etc.) Here the designers are trying to leverage that metaphor (of a banner) to help users understand the interface, and thus which activities are supported by the interface.

It took two inspectors six hours over a 6 week period to completely categorize all 333[2] claims categorizations found in the claims analysis phase. This time calculation only includes joint effort required to justify a particular claim categorization, as well as instances of disagreement and resulting mitigation. Additional time was required by each expert to individually categorize the 333 claims instances. Accurate time records were not kept by each expert, but estimates suggest 3-4 hours per week over the six week period for individual claims categorization, before meeting at the designated time for discussion and mitigation. The full categorization of all claims can be found in Appendix F.2.

Unclassified Claims It is necessary to discuss what we are calling *unclassified* claims. Some of the claims were deemed to be unclassified, since the claim did not impact interruption, reaction, or comprehension. While it is possible to situate these claims within the SBD categories, if the claim does not impact one of the three user goals, it was said to be unclassified. These unclassified

[2]There are 333 total claims classifications (see Section 4.6.1) because some of the individual claims have multiple classifications according to their impact on interruption, reaction, and comprehension. Hence, we have more than the original 253 claims to work with.

Upsides and downsides	**SBD categorization**
+ using pictures as a *single form of information delivery* reduces clutter	activity: supported activities
+ small amount of *white space separates* individual photos	information: screen space
- heavy use of *red color* draws focus away from history and current screen areas	information: color
- use of fancier *font* decreases clarity	information: fonts
- *transition* of the slideshow can distract users	information: animation

Table 4.4: Example categorization of claims tradeoffs. Particular key words (in italics) suggest the correct classification area within the categories (category: sub-category).

claims were revisited twice to make sure the classification assessment was correct. Typical claims from this classification are exemplified by referring to aspects that do not fit within the notification system realm, i.e. they involved aspects of primary task work instead of the secondary task.

Just because a claim did not have an impact on any of the user goals for the system, does not mean that the claim can not be categorized into the framework according to activity, information, or interaction design. It is still possible to discern where a claim fits in the framework and all of the claims, regardless of classification, were categorized. Table 4.5 shows the breakdown of the unclassified claims, according to where they belong in the framework.

Looking at these unclassified claims gives us more evidence about the nature of the large screen information exhibit as a notification system. Recall that notification systems are dual-task systems and provide information to users while they are busy with other tasks. This table reinforces that concept because it shows that functionality concerning interaction with the interface has little or no impact on the user goals associated with the system. This is directly a result of the notification system aspect of these systems. In other words, once a user starts interacting with the system (clicking buttons, looking for information, etc.) then the system is no longer functioning as a notification system; and hence, the original notification system user goals are no longer being pursued.

Overall, the claim categorization process had more instances of disagreement among the inspectors than the classification of the claims using the IRC framework. Often an inspector wanted a particular claim to be in an interaction design category and another would want it in an information design category. These instances were resolved through discussion, with each inspector defending his/her categorization. After discussion, both inspectors were in agreement on the final categorization.

Branch	*Sub-Branch*	*# Unclassified*	*Total*	*%*
Activity	Presence/strength of metaphors	1	24	4.2
	Supported/Unsupported activities	10	54	18.5
Information	Screen space	3	24	12.5
	Object/Background colors	3	28	10.7
	Fonts	2	24	8.3
	Audio	2	11	18.2
	Animation	0	25	0
	Grouping/Layout	2	32	6.25
Interaction	Recognizing Affordances	19	34	55.9
	Behavior of interface controls	8	8	100
	Transition of states	4	23	17.4
	Error recovery/undo	3	3	100
	Feedback on task progress	4	13	30.8
	Configurability	5	30	16.7

Table 4.5: Breakdown of unclassified claims and where they were found. Most of these claims came from the interaction design branch of the framework.

4.7 Synthesis Into Heuristics

After classifying the problems within the framework, we then needed to extract usable design recommendations from those problems. This required re-inspection of the claim groupings to determine the underlying causes to these issues. Since the problems come from different systems, we get a broad look at potential design flaws. Identifying and recognizing these flaws in these representative systems can help other designers avoid making those same mistakes in their work. To facilitate this process, we created a visualization of the claims, to allow easy identification of similar claims, and thus more readily extract underlying design issues.

4.7.1 Visualizing the Problem Tree

To better understand how claims impacted the user goals of each of the systems, a problem tree was created to aid in the visualization of the dispersion of the claims within different areas of the SBD categories. A *problem tree* is a collection of claims for a system class, organized by categories, sub-categories, and critical parameter. It serves as a representation of the design knowledge that is collected from the claims analysis process. A *node* in the problem tree refers to a collection of claims that fits within a single category (from SBD) with a single classification (from the critical parameters). A *leaf* in the tree refers to a single claim, and is attached to some node in the tree.

Recall that SBD encompasses activity, information, and interaction design phases. Within each of these phases, there are more specific areas in which to classify design work. These areas were mentioned earlier in the SBD introduction and serve as sub-categories for our claims analysis. A tree structure was created based on these categories, and it was modified to include the user goals from the notification systems critical parameters [62], specifically high and low levels of

interruption, reaction, and comprehension, as based on the classification effort (Section 4.6.1). The problems identified through the claims analysis were then placed in this tree structure based on the impact to the user goals of the system and the SBD classification.

This problem tree was created specifically for aiding in the creation of heuristics. The natural categorization provided by SBD, augmented with the IRC framework allowed us to effectively determine where specific claims from each system should go in the problem tree. Classifying the problems in this way allows us to determine which areas have the most impact, and also facilitates creating higher level heuristics for other designers and evaluators to use.

Figure 4.2 shows the problem tree that was created for the five target systems. The three main branches correspond to the activity, information, and interaction design phases from SBD [77]. The sub-branches for each of these are taken from generic topics that fall into these categories. For example, when dealing with activity design, designers usually focus on metaphors or which activities to support with the software system. Likewise, information design often deals with color, layout, font types and styles, animation, audio, and grouping. Interaction design focuses on affordances, expected state transitions, feedback, and error control and handling functionality. These sub-categories provide ample coverage of the design phases (activity, information, interaction) while also providing a manageable set.

It is interesting, however not surprising, that the interaction branch has fewer claims associated with it. As mentioned earlier, this results from the fact that interaction with an LSIE typically means that the interface has become the primary task of the user, and thus, is no longer functioning as a notification system. Hence any problems that arise from these areas would have little to no impact on the user goals associated with the large screen information exhibit. To clarify, once a user has started interacting with the display, he/she is no longer interested in the dual-task support that the system primarily provides.

The major strength of using the problem tree comes into play when trying to synthesize the problems into reusable chunks (like heuristics). By using the problem tree, the classification scheme is available in one physical place and you can see which areas have the most impact (more dense nodes) by looking at the number of claims attached to each node. It also provides a summary of the classification and categorization work done through this creation process, both with respect to the SBD categories and the IRC framework. Without this problem tree, one is forced to look at an electronic version through web pages and a complete picture is not possible without extensive effort. The full electronic problem tree, with all of the classifications and categorizations is provided in Appendix D.

4.7.2 Identifying Issues

To glean reusable design guidance from the individual claims, team discussion was used. A team of experts who are familiar with the claims analysis process and the problem tree considers each node in the tree with the aim of identifying one or more *issues* that capture the claims within said node. Issues are design statements, more general than individual claims, that capture underlying design ideas inherent in multiple, related claims. In this case, our experts were the same ones who performed the classification/categorization work. The research team looked at each leaf node in the problem tree and through discussion, formulated one or more underlying issues that seemed to explain the claims in that node. This effort produced 22 issues that covered the 333 claims. It should be remembered that in the classification effort, some of the claims were deemed to be

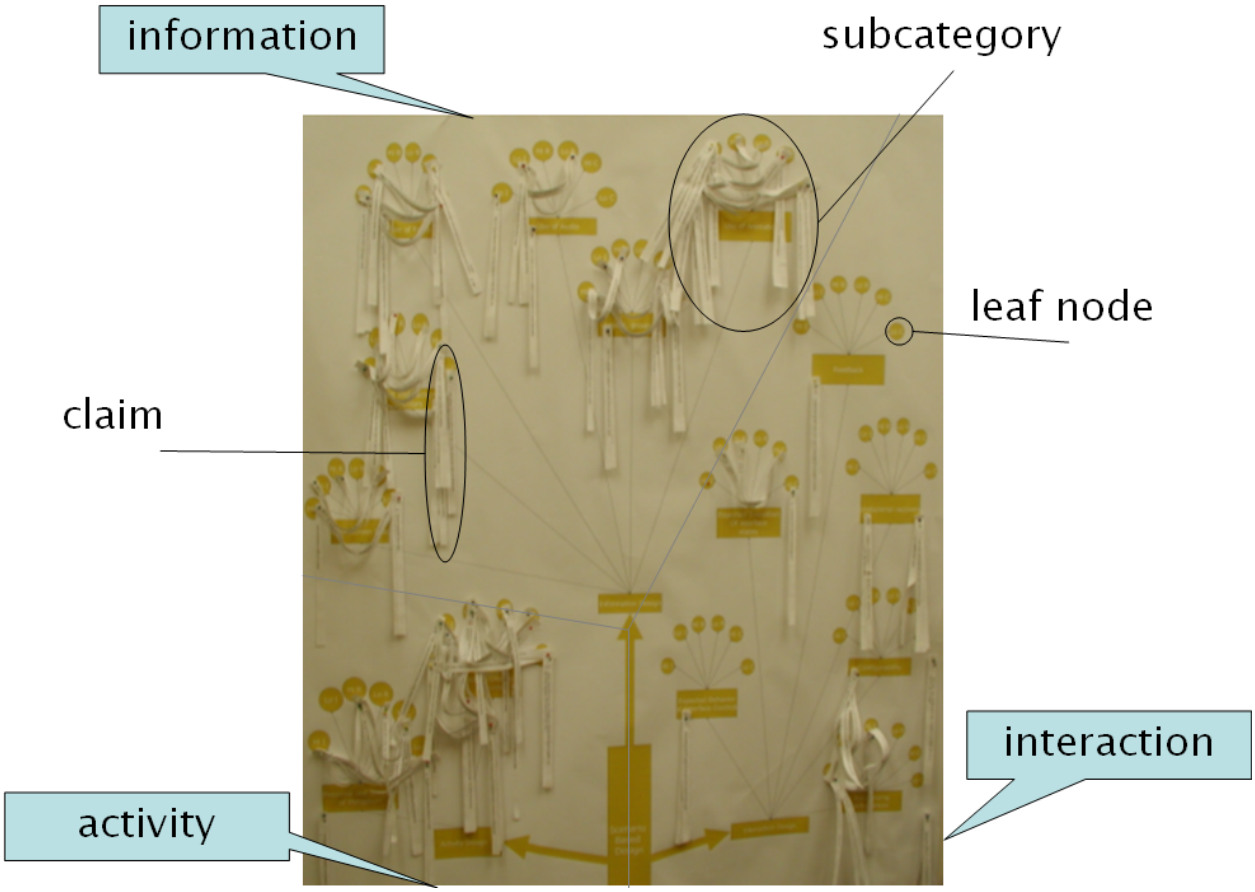

Figure 4.2: Problem tree based on claims from the five large screen information exhibits. Labeled are the activity, information, and interaction branches, a sub-category within the information branch, a leaf node within a sub-category, and a claim attached to a leaf node. Note the sparsity of the interaction branch due to the small number of claims.

unclassified, due to the fact that they mostly referred to interaction with the displays, and thus those claims did not contribute to the issues.

This process relies on the wording of the claims in conjunction with the specific claims in the categorization. Here the wording refers to the actual words used to describe the claim. Each leaf node has a unique type of claim associated with it from the categorization, and because each claim in this node has a similar impact on the user goals, we can determine underlying causes for these impacts. In other words, the problem tree we have, when taken as a whole, allows one to systematically extract design guidance from system analysis by visiting one leaf node at a time. Re-inspection involves taking the leaf nodes of the problem tree and determining what, if any, underlying causes produced the problems in that specific node. For example, under the activity design branch, in the metaphors sub-branch, we find five problems that increase the level of interruption a user would experience. Analyzing these problems reveals that inappropriate metaphors can increase the time it takes a user to understand a display, thereby increasing the amount of interruption he/she would experience with the display. This finding is recorded in a list of high level issues that serve as candidate heuristics.

Claim tradeoff	*Issue*
+*Banner* suggests late-breaking changes and adds excitement +*pictorial representation* of story will draw interest to the story -*collage metaphor* may give *disorderly haphazard appearance* -*lack of organization* because of *collage metaphor* can *hinder efforts* to find an artifact +*collage metaphor* allowed the system to place pictures in an unorganized fashion to *use more screen space*	Employ highly recognizable metaphors that use/stress organizational layout

Table 4.6: Example of transforming specific claims tradeoffs into high level issues. Here we have five tradeoffs from the "metaphor" sub-branch within the "activity" branch. The issues serve as potential heuristics and capture high level design guidelines for LSIE systems. The italicized words indicate the metaphor used or the consequence of the metaphor. These keywords suggest possible underlying issues and lead to the creation of the wording of the issue.

So how was this extraction performed? Since the claims are classified according to the user goals of the LSIE system class and they are categorized within the correct area of SBD, we have a general idea of what the underlying causes could be. Further inspection of the claim wordings suggest specific design issues pertaining to the claims. The overall goal is to extract the commonalities among the claims within a leaf node. By focusing on one or two similarities within multiple claims in a node, we can identify potential design guidelines that capture the common elements. This process relies upon the problem tree, which is basically a representation of the classification and categorization efforts described earlier.

Table 4.6 provides an example of the five aforementioned claims tradeoffs and the resulting issue after inspection. Again the wordings within the claims help us to identify common attributes of the claims, and thus to formulate phrases that capture specific design issues. From the example claims in Table 4.6, we have several references to metaphors and most deal with some form of layout or organization; hence we claim that using familiar metaphors with good organization would be better for supporting the particular user goals associated with the LSIE class. Appendix F.3 provides a complete list of the claims and the resulting issues.

Each of the branches and sub-branches was analyzed in this way, resulting in a list of issues. In all, 22 issues were identified through the analysis of the problem tree. Our two experts went through the entire problem tree, identifying potential claims and marking those claims that seemed strange. Unclassified claims were not considered. Then a high level issue was created

Issues	**Resulting Heuristic**
Use *cool colors* (blues, greens) for *borders and backgrounds* Use *warm colors* (reds, yellows) for important information items and highlighting *Avoid heavy use* of *bright colors*.	Appropriate color schemes can be used for supporting information understanding.

Table 4.7: Example of how to extract heuristics from the design issues. Here we have several design issues on the left and the resulting heuristic on the right. Italics show the keywords that led to the formulation of the heuristic.

that attempted to capture the majority of the claims within a leaf node of the problem tree. These wordings are somewhat arbitrary but they do provide useful design guidance. These 22 issues (found in Appendix E) represent high level design guidance extracted from the claims in the problem tree. They do not quite serve as heuristics because some are related by other, even higher level causes. These issues were then categorized and synthesized into general heuristics through a similar discussion and mitigation technique, relying upon commonalities within the issues. The final synthesis resulted in a list of eight potential heuristics, based on the type and frequency of the problem occurring in the five systems.

4.7.3 Issues to Heuristics

Armed with the 22 high level issues, we now needed to extract a subset of high level design heuristics from these issues. Twenty-two is unmanageable for formative heuristic evaluation [66] and in many cases the issues were similar or related, suggesting opportunities for concatenation and grouping. This similarity allowed us to create higher level, more generic heuristics to capture the issues. This process involved inspecting the issues for underlying similarities and then creating a new wording that captured the issues. This new wording serves as the heuristic in the final set. The wordings of some of the original issues are also provided with the heuristics as details describing the new heuristics, providing better understanding of the applicability and scope of an individual heuristic. These issues also provide some clarification for how the final heuristic could be applied in an evaluation. We created eight final heuristics, capturing the 22 issues discovered in the earlier process. Table 4.7 provides an example of how we moved from the issues to the heuristics. In most instances, two or three issues could be combined into a single heuristic. However some of the issues were already at a high level and were taken directly into the heuristic list. The technique in this synthesis process relies upon team discussion to come up with the individual wordings that captured the issues. Appendix F.4 provides a complete listing of the issues and resulting heuristics from the synthesis process. The following section provides the final heuristics with explanatory text taken directly from the issue list (Appendix E).

4.7.4 Heuristics

Here is the list of heuristics that can be used to guide evaluation of large screen information exhibits. Explanatory text follows each heuristic, to clarify and illustrate how the heuristics could impact evaluation. Each is general enough to be applied to many systems in this application class, yet they all address the unique user goals of large screen information exhibits.

- **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.

4.8 Discussion

Initially one may have questions about the applicability of this method for other system classes. Would different inspectors come up with the same heuristics, if they followed the method as described? Perhaps, perhaps not. Individual differences can manifest in all stages of this method, from system selection, to claims analysis, to classification, and so on. Of course different people would uncover different heuristics through this process. However, the point of the method is to provide a structured process to producing such heuristics, not ensure that the set of heuristics produced is the best set for the system class. Proving that requires testing and comparing the new set of heuristics with other alternatives, to assess system problem coverage and applicability.

To better illustrate this point, consider claims analysis by itself [15]. Given a scenario for some system, two different inspectors will come up with different sets of claims for the scenario. Does this suggest that the claims analysis method is weak or faulty? No, it simply illustrates the complexity of design, and how individuals insert their knowledge upon the process. In fact, it is this complexity and reliance upon individual knowledge that strengthens the claims analysis technique. More people can identify more claims for a given scenario, thus broadening the understanding of the system. An analogous argument can be made for the method used in this work. The set of heuristics found in this particular effort may not capture every last detail for every LSIE, but it does not have to do that. Furthermore, if other evaluators went through our creation method and came up with a different set of heuristics, it is likely that the set would provide similar design guidance as the one produced in this work because the would be based on the same underlying design problems in the target systems, as identified through impacts on the critical parameters and how they fall within the SBD categorization.

The strength of this method is evident in the clearly defined steps for producing heuristics from system inspection. Instead of blindly guessing about correct design guidelines for a system class, one can follow this process to systematically derive heuristics from example systems. This structure is necessary for gaining design guidance in new areas of system development, like notification systems, or ubiquitous computing, or real world interfaces. These areas are relatively new, and they do not have established usability techniques specifically tailored to the unique user goals associated with them.

Process Analysis When reflecting upon this creation process, it is important to contrast it with other heuristic development approaches. Consider the creation of Nielsen's 10 heuristics. These heuristics were originally described in a 1990 Communications of the ACM article [66] and were based on observations of system use by those authors and several years of experience. Of course Nielsen was/is an experienced consultant, so his and Molich's experience is indeed valuable. Later works by Nielsen do not describe the genesis of these heuristics [71, 69, 70], and it seems that the creation of that original set is still a mystery. Perhaps more important, they do not describe any structure for creating one's own heuristics. They seemed more concerned with generating and perfecting generic heuristics. In contrast, our creation process is based on six distinct steps, with a clear structure that can be followed from start to finish.

As hinted earlier, neither Mankoff et al. nor Baker et al. provide a detailed description of their respective heuristic creation processes. It appears that Mankoff et al. relied upon Nielsen's original set of heuristics as a basis, then went through some modification process to derive a set tailored to ambient displays [56]. This modification is not entirely clear from their description

but they did show that tailored heuristics are more desirable over more generic sets. Baker et al. performed similar heuristic creation effort, producing heuristics tailored to groupware systems [5]. While their creation method was not entirely replicable, at least their process is grounded in theoretical underpinnings surrounding their target system class. Like Mankoff, Baker also showed that specific heuristics are better for evaluation over more generic heuristics [5]. 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. This provides some inspiration for our effort, because we embrace the notion of critical parameters and it is encouraging to see others attempting similar efforts.

It seems that the most important steps in our creation process are the scenario extraction, claims analysis, classification, and categorization. These steps provide the background and support for the synthesis of heuristics. It is in these steps that the critical parameters support the creation process by focusing creation effort on identifying claims that impact the parameters. This leads to heuristics that allows evaluators to describe problems that are related to the important user goals for a system. This further illustrates the utility of critical parameters and how, when identified for a system class, they can guide both design and evaluation cycles.

We feel good about this creation process, and our successful creation of heuristic tailored to the LSIE system class provides some indication of how the technique can be applied. However, there are some drawbacks related to the specific steps in the process. Specifically this process is highly dependent upon the individuals involved in the process due to the analytic requirements in extracting design issues and synthesizing heuristics. If there were mechanisms to guide the analysis required in these phases, we could reduce variability among different creators using the method, reducing overall creation time.

Finally, consider the time it took for Molich and Nielsen to publish their heuristics. The accumulated knowledge reported in their heuristics was aggregated roughly 30 years after computing became mainstream. However, new research areas (like notification systems) need evaluation tools in the short term as developers and designers can not wait 30 years to test their systems. Hence, the method described here would allow usability professionals to develop heuristics in a systematic and structured way, with turn around time on the order of a few weeks as opposed to years. Granted, there is significant effort involved, but the process at least produces some form of usable heuristic guidance (see Chapters 6 and 5 for validation of this claim).

4.9 Summary

We have described the process of creating usability heuristics for LSIEs. By using scenario based design, which focuses on user goals and tasks, we have inspected five different systems from the information exhibit class, and identified several high level heuristics. Several important steps make up this creation process. Claims analysis allows us to extract potential design tradeoffs from the systems. Classifying these claims according to impact on user goals provides initial indicators of similar claims, but there are too many claims within a single classification (like high interruption). Further categorization is required to more fully separate the different types of claims from one another. The SBD categories provide a simple and manageable breakdown of claim types. By using a physical model of the resulting problem tree, one is able to consider a small group of claims at a time to process and extract higher level design issues. After the design issues are extracted from

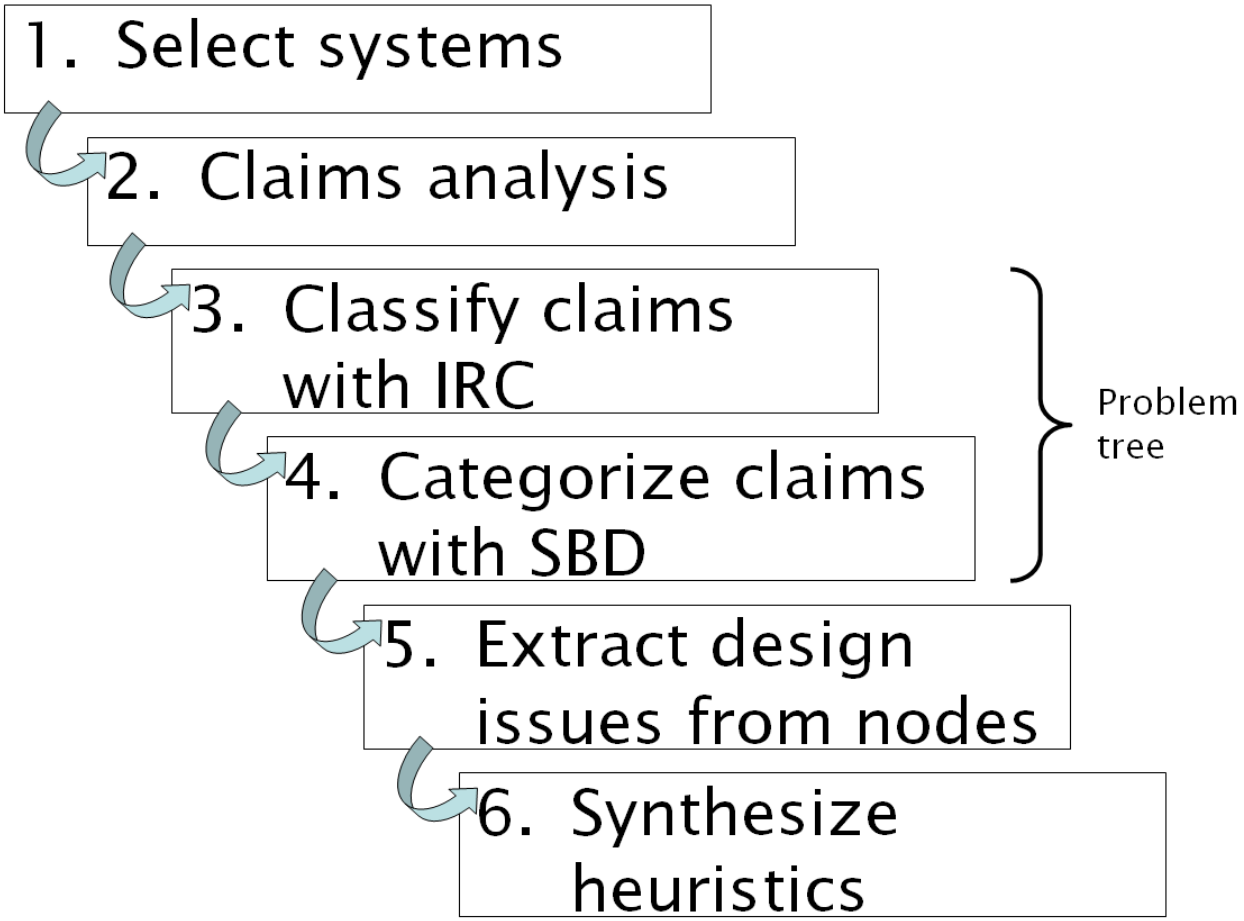

Figure 4.3: Creation process used to extract heuristics from system inspection.

the problem tree, one is able to synthesize heuristics from similar issues, resulting in a smaller, more manageable set. Figure 4.3 provides a graphical depiction of the creation process.

By grounding these heuristics in real systems that have been developed and used, we have established a set that is based on real system problems. Other researchers do not adequately describe how their heuristics were developed [5, 56], which allows critics to question their origins and doubt their validity; whereas we have shown the explicit steps taken in the creation process, which provides the background and foundation for this heuristic set.

We envision these heuristics as guiding and grounding analytical evaluation of LSIEs. However, we do not expect practitioners to simply pick these up and use them without knowing whether or not they work, especially when they have alternatives that have been extensively studied (Nielsen's for example). Therefore, the next step in this creation process is to perform empirical tests on our heuristics. We will compare them to the more established heuristics (like Nielsen's), using accepted UEM comparison metrics [40], thereby determining whether or not our heuristics will be useful in formative usability evaluation for the LSIE class. The next chapter reports an experiment in which this comparison study was executed. Chapter 6 describes an effort to show the utility of the heuristics described here through actual use in evaluation. It includes two application examples, and some feedback from international experts.

Chapter 5

Heuristic Comparison Experiment

5.1 Introduction

This chapter reports on an experiment in which the heuristics that were developed in Chapter 4 are tested to see if they indeed provide a useful and sound evaluation approach to the usability analysis of large screen information exhibits. If these new heuristics perform as well as or better than existing alternatives (other heuristic sets), then that is evidence that the heuristic creation method produces usable evaluation tools, further strengthening the notion of critical parameters as a sound UEM creation approach.

Now that there is a set of heuristics tailored for the large screen information exhibit system class, a comparison of this set to more established types of heuristics can be done. The purpose of this comparison would be to show the utility of this new heuristic set. This comparison needs to be fair, so that determining the effectiveness of the new method will be accurate.

To assess whether the new set of heuristics provides better usability results over existing alternative sets, we conducted a comparison experiment in which each of three sets of heuristics were used to evaluate three separate large screen information exhibits. We then compared the results of each set through several metrics to determine the better evaluation methods for large screen information exhibits.

This comparison is necessary for several reasons. Because we have a new UEM, we need to show that it performs as well as or better than alternative methods. If this method does not produce similar results, then an alternative UEM creation approach is necessary. In addition, this experiment serves as initial validation of the heuristic set by showing it can be used to discover usability issues with large screen information exhibits.

5.2 Approach

The following sections provide descriptions of the heuristics used, the comparison method, and the systems used in this experiment.

Visibility of system status
Match between system and real world
User control and freedom
Consistency and standards
Error prevention
Recognition rather than recall
Flexibility and efficiency of use
Aesthetic and minimalist design
Help users recognize, diagnose, and
recover from errors
Help and documentation

Figure 5.1: Nielsen's heuristics. General heuristics that apply to most interfaces. Found in [70].

5.2.1 Heuristic Sets

We used three different sets of usability heuristics, each at a different level of specificity for application to large screen information exhibits, ranging from a set completely designed for this particular system class, to a generic set applicable to a wide range of interactive systems. The following sections provide more information on each set, as well as a listing of the heuristics.

Nielsen

The least specific set of heuristics was taken from Nielsen and Mack [70]. This set is intended for use on any interactive system, mostly targeted towards desktop applications. Furthermore, this set has been in use since around 1990. It has been tested and criticized for years, but still remains popular with usability practitioners. Again, this set is not tailored for large screen information exhibits in any way and has no relation to the critical parameters for notification systems. See Figure 5.1 for a listing of this set of heuristics.

Content Comparison It is worthwhile to consider at a conceptual level what is contained in the actual heuristics that are provided to the evaluators. As seen in Chapter 4, the new heuristics consist of a general statement and several sentences describing in more detail the idea contained in the general statement. These sentences are the *issues* that were identified through the creation process (Section 4.7.2). This bundle of information (statement plus issues) provides considerable design knowledge to the evaluator as he/she performs the heuristic evaluation.

In contrast, Nielsen's heuristics are more generic in the information provided, requiring the evaluator to perform a mental interpretation of the heuristic meaning when applied in an evaluation. Figure 5.1 illustrates the vague nature of the general statements provided in [70]. Nielsen also provides some sentences to clarify the general statements, but the information contained in these statements is also less specific than the content of the issues provided with the new set of heuristics.

The implication of having more design-related content in the heuristic descriptions is that evaluators may be able to word problems that are found in terms of offending artifacts in the design, possibly suggesting immediate fixes or at least suggesting which areas to search for ideas in similar artifacts, like in a claims library [17]. Without this close tie to the underlying goals of the system,

Notifications should be timely
Notifications should be reliable
Notification displays should be consistent
(within priority levels)
Information should be clearly understandable
by the user
Allow for shortcuts to more information
Indicate status of notification system
Flexibility and efficiency of use
Provide context of notifications
Allow adjustment of notification parameters
to fit user goals

Figure 5.2: Berry's heuristics. Tailored more towards Notification Systems in general. Found in [9].

evaluators are forced to write out problems in generic terms, which must then be interpreted by the designer during re-design phases, potentially increasing the overall development time and costs.

Berry

The second heuristic set used in this comparison test was created for general notification systems by Berry [9]. This set is based on the critical parameters associated with notification systems [62], but only at cursory levels. This set is more closely tied to large screen information exhibits than Nielsen's method in that large screen information exhibits are a subset of notification systems, but this set is still generic in nature with regards to the specifics surrounding the LSIE system class. See Figure 5.2 for a listing of this heuristic set.

Content Comparison Although Berry's heuristics are at least attuned to the critical parameters associated with Notification Systems, they are still generic in relation to the LSIE system class. In assessing the content of these heuristics, they attempt to illustrate the overarching goals of general notification systems by focusing evaluator attention on conceptual issues surrounding the application and use of the system. This is similar to Nielsen's heuristics, but these are more focused on a specific type of system (NS) as opposed to any generic interface.

Like Nielsen's, these heuristics also have some explanatory text associated with the heuristic with which evaluators can more accurately assess an interface. But, an evaluator will still need to perform some form of mental interpretation when applying these heuristics to LSIE systems. In Berry's case, the explanatory text describes examples of possible notification systems and why the heuristic would be important to consider in an evaluation. This information is more specific than Nielsen's, and helps evaluators understand how the heuristic is intended to be used.

In comparison to Somervell's heuristics, the content in Berry's heuristics is much more general in nature. While they do address notification system concerns, they do not address the more specific levels of each parameter that define the LSIE system class (namely self-defined interruption, high comprehension, and appropriate reaction). This difference could impact design guidance during the redesign portion of system development cycles. Because evaluators must be more creative in

their application of Berry's heuristics, he/she may not be able to describe the problem in terms that suggest immediate re-design fixes.

Somervell

The final heuristic set is the one created in this work, as reported in Chapter 4. This set is tailored specifically to large screen information exhibits, and thus would be the most specific method of the three when targeting this type of system. It is based on specific levels of the critical parameters associated with the LSIE system class.

In contrast to the other two sets, these heuristics are tuned to the unique challenges surrounding the development and use of LSIE systems. Application is straightforward, with less mental interpretation from the evaluator. Describing problems should also be facilitated by focusing the wordings on the artifacts in the design that cause or contribute to the problem. Evidence from example application of these heuristics is provided in Chapter 6

Thus, we have a small spectrum of specificity with these three heuristic sets, ranging from the generic to the specific. In line with previous research [56, 5], we hypothesize that the more specific methods will produce "better" evaluation results. The following section describes the comparison technique used and what we will use as a definition for "better".

5.2.2 Comparison Technique

To determine which of the three sets is better suited for formative evaluation of large screen information exhibits, we use a current set of comparison metrics that rely upon several measures of a method's ability to uncover usability problems through an evaluation. This technique is described fully in [40]. A terse description of this method follows.

The comparison method we are using typically relies on five separate measures to assess the utility of a given UEM for one's particular needs, but we will only use a subset in this particular comparison study. Hartson et al. report that thoroughness, validity, effectiveness, reliability, and downstream utility are appropriate measures for comparing evaluation methods [40]. A subset of these measures was used for this work due to data collection methods and relative worth of the metric. Specifically, our comparison method capitalizes on thoroughness, validity, effectiveness, and reliability, abandoning the downstream utility measure. This choice is used here because long-term studies are required to illustrate downstream utility. Besides, application examples in Chapter 6 further illustrate the utility of the new method for providing re-design guidance and act as a replacement.

Thoroughness

The first measure we will use is *thoroughness*. This measure gives an indication of a method's ability to uncover a significant percentage of the problems in a given system. Thoroughness consists of a simple calculation of the number of problems uncovered by a single UEM divided by the total number of problems found by all three methods.

$$thoroughness = \frac{\#\ of\ problems\ found\ by\ target\ UEM}{\#\ of\ problems\ found\ by\ all\ methods}$$

Validity

Another measure, which relies on the same data, is *validity*. Validity refers to the ability of a method to uncover the types of problems that real users would experience in day to day use of the system, as opposed to simple or minor problems. Validity is measured as the number of real problems found divided by the total number of real problems identified in the system.

$$validity = \frac{\#\ of\ problems\ found\ by\ target\ UEM}{\#\ of\ problems\ in\ the\ system}$$

The number of real problems in the system refers to the problem set identified through some standard method that is separate from the method being tested. Section 5.2.5 provides a description of the problem sets used in this test.

Effectiveness

Effectiveness combines the previous two metrics into a single assessment of the method. This measure is calculated by multiplying the thoroughness score by the validity score.

$$effectiveness = thoroughness * validity$$

Reliability

Reliability is a measure of the consistency of the results of several evaluators using the method. This is also sometimes referred to as inter-rater reliability. This measure is taken more as agreement between the usability problem sets produced by different people using a given method. This measure is calculated from the *differences* among all of the evaluators for a specific system as well as by the total number of *agreements* among the evaluators, thus two measures are used to provide a more robust measurement of the reliability of the heuristic sets:

$$reliability_d = difference\ among\ evaluators\ for\ a\ specific\ method$$

$$reliability_a = average\ agreement\ among\ evaluators\ for\ a\ specific\ method$$

For calculating reliability, Hartson et al. recommend using a method from Sears [81] that depends on the ratio of the standard deviation of the numbers of problems found by the average number found [40]. This measure of reliability is overly complicated for current needs, thus a more traditional measure that relies upon actual rater differences is used instead.

These measures are only part of what Hartson et al. provide in [40]. These and not other measures from their work (e.g. downstream utility) are used because these measures are the most prevalent among other UEM comparison studies [40, 21].

5.2.3 Systems

Three systems were used in the comparison study providing a range of applications for which each heuristic would be used in an analytic evaluation. The intent was to provide enough variability in the test systems to tease out differences in the methods. The following sections describe the three systems used in this study.

Source Viewer

One of these large screen systems is a local television station's master control room. This room provides quality control to the broadcasts before they leave the station and go to the transmission antennas. The controller sits at a work station, surrounded by controls, video screens, and oscilloscopes. Mounted on the wall directly in front of this work station is a large screen display, upon which there are 13 different video feeds, plus a clock. This large screen display, which we will call the Source Viewer, helps the controller keep track of the programming for the day. The controller's main focus is upon the myriad video monitors and control devices at his work station, and he relies upon the large screen for source switching.

The Source Viewer exists to provide surveillance-like information to the controller as he/she makes sure the broadcast is correct and on schedule with the programming guidelines. To accomplish this task effectively, the controller must be able to quickly and easily see the information from several different sources, and press buttons on the control panel to initiate changes on what is transmitted through the broadcast antennas. See Figure 5.3 for a screenshot of the Source Viewer and its layout.

The following scenario illustrates the setup and typical usage of the Source Viewer:

> John, the control manager, must ensure that the appropriate breaks occur in the broadcast at specific times. This job is one of the most important in the station because he controls what is broadcast over the antennas, and ultimately, what the end consumer sees on his/her television set. The main aspect of this work involves switching among sources to broadcast the required signal at the right times. Timed switches are triggered mostly through audio cues, with some visual cues to add redundancy. However it is paramount to understand exactly what will be broadcast from a particular source, especially if that source is the one to be broadcast on the button press.
>
> Hence, there is a large screen display to help John keep track of what is being broadcast and what options he has for switching the signal. This screen shows the current signal and up to 13 other sources that can be queued for switching at the touch of a button. In addition, there is a digital clock to assist with timings. Each source has a label box, positioned at bottom center. This label is opaque and thus obscures what is behind it. The label can be removed but not moved within the source box.
>
> This simple display provides invaluable information to John as he performs his duties. He does not look at the display the whole time, but rather relies on audio and certain visual cues to perform the source switching required in completing his tasks. He looks at the display briefly before switching sources to ensure that the correct video feed is broadcast at the correct times.

Why Source Viewer? The Source Viewer was chosen as a target system for this study because we wanted an example of a real system that has been in regular use for an extended period. We immediately thought of command and control situations. Potential candidates included local television stations, local air traffic control towers, electrical power companies, and telephone exchange stations. We finally settled on local television command and control after limited responses from the other candidates.

Figure 5.3: Layout of the Source Viewer large screen display at WDBJ 7 in Roanoke, VA.

We finally chose WDBJ 7 in Roanoke, VA as the target location because they were interested in new technology and had a large screen system installed at their Roanoke broadcast station. Furthermore, the president of the station was interested in public relations and wanted to be of assistance to this research effort.

By using a local system, we are giving back to the community through improved usability in a local television broadcast station, thereby increasing the quality in the broadcasts to the public. We have the opportunity to show real impact in this work through improved system design for a real world command and control situation.

Plasma Poster

The Plasma Poster [20] is a large screen program that runs on a situated plasma screen in different locations within an organization. Typical installation locations include kitchen areas, hallways, alcoves, atriums, and similar areas with high traffic. Information is posted to the Plasma Poster by members of the organization. This information can include any type of content but is typically announcements of upcoming events or web page snippets of information that people find interesting or amusing.

The intent of the display is to encourage information sharing and thus casual, unplanned interactions among the organization members [20]. These types of interactions would typically occur during break times but could occur at anytime. Users are often busy reading, editing, or working and rely on the Plasma Poster for local updates.

Content is sent to the display in two ways; either through a web interface or through email. The web interface is a standard submission style interface where a user types in relevant information

and hits a submit button. The email interface is unusual in that the Plasma Poster has an email address and you can send messages to it like it is a real person. Content is not posted from the actual Plasma Poster itself, so a person cannot walk up to the display and add content.

People who post content have limited information (name, work location, etc.) about them also posted at the bottom of the display. In addition, there is a cyclic list of upcoming posts. These upcoming posts are the ones that will be displayed after the current wait cycle has expired. Thus, the display automatically changes its content after a short time (about 30 seconds). At the very bottom of the screen are controls for browsing through content, gaining details, and sending a post to yourself via email.

The following scenarios describe the Plasma Poster and how it is typically used:

1. Elizabeth goes to the kitchen to get some coffee. She glances at the Plasma Poster and sees a new announcement for an upcoming presentation by her friend on his recent research effort. She goes over to the display and reads the date, time, and location of the presentation and makes a mental note to write it down in her schedule.

2. Alex is walking down the hall to his office when he sees Kathy looking at the Plasma Poster. He stops by and sees that she is viewing a posting from a mutual friend in the building about an informal get together later in the week. He stops, asks her if she is going, and they make plans to car-pool. He then remembers they have a meeting and suggests they go over some information beforehand.

Why Plasma Poster? We wanted to include the Plasma Poster because it is one of very few LSIE systems that has seen some success in terms of long term usage and acceptance. It has seen over a year of deployment in a large research laboratory, with reports on usage and user feedback reported in [20].

This lengthy deployment and data collection period provides ample evidence for typical usability problems. We can use the published reports as support for our problem sets. Coupled with developer feedback, we can effectively validate the problem set for this system.

Notification Collage

The Notification Collage [36] (NC) shows a variety of information on a large screen that looks like a large wall. A full description can be found in Section 4.4.2.

Why Notification Collage? We chose the Notification Collage as the third system for several reasons. First, we wanted to increase the validity of any results we find. By using more systems, we get a better picture of the "goodness" of the heuristic sets, especially if we get consistent results across all three systems. Secondly, we wanted to explicitly show that the heuristic set we created in this work actually uncovered the issues that went into that creation process. In other words, since the Notification Collage was one of the systems that led to this heuristic set, using that set on the Notification Collage should uncover most of the issues with that system. Finally, we wanted to use the Notification Collage out of the original five because we had the most developer feedback on that system, and like the Plasma Poster, it has seen reasonable deployment and use.

5.2.4 Hypotheses

At this point we need to specify exactly what we are looking to find in this experiment. We needed to know about the systems we were testing, and the measurements we were using before we could explicitly state any of our hypotheses. We have three main hypotheses to test in this experiment:

1. **Somervell's set of heuristics has a higher validity score for the Notification Collage.** We believed this was true because the Notification Collage was used in the creation of Somervell's heuristics, thus those heuristics should identify most or all of the issues in the Notification Collage.

2. **More specific heuristics have higher thoroughness, validity, and reliability measures.** We felt this was true because more specific methods are more closely related to the systems in this study. Indeed, from Chapter 3, we discussed how previous work suggests system-class level heuristics would be best. This experiment illustrates this case for heuristic evaluation of large screen information exhibits.

3. **Generic methods require more time for evaluators to complete the study.** This seems logical because a more generic heuristic set would require more interpretation and thought, hence we felt that those evaluators who use Nielsen's set would take longer to complete the system evaluations, providing further impetus for developing system-class UEMs.

5.2.5 Identifying Problem Sets

One problem identified in other UEM comparison studies involves the calculation of specific metrics that rely upon something referred to as the "real" problem set (see [40]). In most cases, this problem set is the union of the problems found by each of the methods in the comparison study. In other words, each UEM is applied in a standard usability evaluation of a system, and the "real" problem set is simply the union of the problems found by each of the methods. There are issues with this approach because there is no guarantee that the problems found by the UEMs are the problems that would be experienced by real users in normal day to day activity with the system in question.

This comparison study also faced the same challenge. Instead of relying on evaluators to produce sets of problems from each method, then using the union of those problem sets as the "real" problem set, analysis and testing was performed on the target systems *beforehand* and the problem reports from those efforts were used to come up with a standard set of problems for each system. Coupled with a new testing approach, this eliminated much of the variability inherent in most UEM comparison studies that arises from having to read through problem reports and deduce (perhaps erroneously) the intention of the evaluator. The following sections describe how the problem sets were derived for use in this test. Descriptions of the actual testing methodology start in Section 5.3.

Source Viewer Problem Set

To determine the problem sets experienced by the users of this system, a field study was conducted. Two interviews with the users of the large screen system, as well as observation were conducted.

This abbreviated field study produced some interesting insight into the usage and nature of the large screen and how it impacted daily job activities.

The interviews were conducted over a two day period with the control manager at the news station. They were informal in nature, and probed current usage of the large screen in supporting daily work activity. These interviews were recorded, using both written note taking and digital audio recordings. The digital recordings were then transcribed to allow for complete analysis, and identification of usability problems.

Observations occurred during a 4 hour time period, split over 2 days. The observer stayed out of the way, usually off to the side, and watched the people in the control room as they went about their daily work. Notes on work context, situational context, and interactions with the large screen were recorded. These observations served to provide evidence of the current usability issues that the users encountered with the large screen display.

Field Study Results Concurrently with the field study, a claims analysis [15] was performed on the Source Viewer. This analysis was based on typical usage context for the system, and was intended to capture the typical usability issues with it. We used this format with all three systems to ensure a common, easy to understand representation of the problem sets.

The claims analysis identified 11 claims, altogether covering about 30 tradeoffs with the Source Viewer. These claims were then verified with the field study results. Each of the claims captures a typical usability tradeoff in the design of the source viewer as reported from interviews with the system users. This verification ensures that the problems we have in the claims analysis are indeed a subset of the real problems that are experienced by the users of this system.

Plasma Poster Problem Set

Analytic evaluation augmented with developer feedback and literature review served as the method for determining the real problem set for the Plasma Poster. We employed the same claims analysis technique that we used in the creation process to identify typical usability tradeoffs for the Plasma Poster. After identifying the usability issues, we asked the developers of the system to verify the tradeoffs.

In all there were 14 tradeoffs for the Plasma Poster and most were validated by the lead developer of the system. In addition, we used the literature available on the Plasma Poster to verify certain claims that represented behavior the developers could not recall or had not observed. Thus, we used all 14 claims in the experiment.

Notification Collage Problem Set

To validate the problem set for the Notification Collage, we contacted the developers of the system and asked them to check each tradeoff as it pertained to the behavior of real users. The developers were given a list of the tradeoffs found in our claims analysis (from Chapter 4) and asked to verify each tradeoff according to their observations of real user behavior. This problem set validation insured that the problems we would be using in the test were at least a subset of the real problems that users experienced in their daily use of the system.

Method	*Ordering*		
Nielsen	PSN	PNS	SNP
	SPN	NPS	NSP
Berry	PSN	PNS	SNP
	SPN	NPS	NSP
Somervell	PSN	PNS	SNP
	SPN	NPS	NSP

Table 5.1: Latin Square balanced ordering for the test setup used in the comparison study. P stands for Plasma Poster, S stands for Source Viewer, and N stands for Notification Collage.

5.3 Testing Methodology

This experiment involves a 3x3 mixed factors design. We have three levels of heuristics (Nielsen, Berry, and Somervell) and three systems (Source Viewer, Plasma Poster, and Notification Collage). The heuristics variable is a between-subjects variable because each evaluator sees only one set of heuristics. The system variable is within-subjects because each participant sees all three systems. For example, evaluator 1 saw only Nielsen's heuristics, but used those to evaluate all three systems.

We used a balanced Latin Square to ensure learning effects from system presentation order would be minimized. Thus, we needed a minimum of 18 participants (6 per heuristic set) to ensure coverage of the systems in the Latin Square balancing. Table 5.1 shows the presentation order resulting from the balanced Latin Square. The first entry in the first column indicates which system the participant would see first, second, and third; or in this case, Plasma Poster (P), Source Viewer (S), then Notification Collage (N). We did not expect the system presentation order to impact the study, but using the Latin Square ordering effectively eliminates any possible effects from presentation order.

5.3.1 Participants

As shown in Table 5.1, we needed a minimum of 18 evaluators for this study. Twenty-one computer science graduate students who had completed a course on Usability Engineering volunteered for participation as inspectors. Six participants were assigned to each heuristic set, to cover each of the order assignments. Three additional students volunteered and they were randomly assigned a presentation order.

These participants all had knowledge of usability evaluation, as well as analytic and empirical methods. Furthermore, each was familiar with heuristic evaluation. Some of the participants were not familiar with the claim structure used in this study, but they were able to understand the tradeoff concept immediately.

Unfortunately, one of the participants failed to complete the experiment. This individual apparently decided the effort required to complete the test was too much, and thus filled out the questionnaire using a set pattern. For example, this participant answered the questions exactly the

same for every claim, for all three systems. It is obvious that they did not look at or think about the issue and the heuristic applicability, but chose to simply mark the circles as they saw fit. It was only obvious after about 10 claims that the participant was simply marking the answers exactly the same as on the previous sheet. As a result, we were forced to disregard this person's answers and data. This makes the final number of participants 20, with seven for Nielsen's heuristics, seven for Berry's heuristics, and six for Somervell's heuristics.

5.3.2 Materials

Each target system was described in one to three short scenarios, and screen shots were provided to the evaluators. The goal was to provide the evaluators with a sense of the display and its intended usage. This material is sufficient for the heuristic inspection technique according to Nielsen and Mack [70]. This setup ensured that each of the heuristic sets would be used with the same material, thereby reducing the number of random variables in the execution of this experiment.

A description of the heuristic set to be used was also provided to the evaluators. This description included a listing of the heuristics and accompanying text clarification. This clarification helps a person understand the intent and meaning of a specific heuristic, hopefully aiding in assessment. These descriptions were taken from [70] and [9] for Nielsen and Berry respectively.

Armed with the materials for the experiment, the evaluator then proceeded to rate each of the heuristics using a 7-point Likert scale, based on whether or not they felt that the heuristic applied to a claim describing a design tradeoff in the interface. Thus they are judging whether or not a specific heuristic applies to the claim, and how much so. Marks of four or higher indicate agreement that the heuristic applies to the claim, otherwise the evaluator is indicating disagreement that the heuristic applies.

To fully understand this setup, one needs to understand the presentation used to provide the evaluator with the necessary comprehension of a potential usability problem in the target system. Instead of listing a specific problem in the interface, we used a claims analysis technique of showing a design decision, along with its associated upside and downside tradeoffs. This presentation format gets the evaluator to think about the claim (design decision) in terms of the rationale behind it, as well as the potential negative results of using that design decision. This format allows the evaluator to make their own judgment as to whether a particular heuristic would apply to the issue described in the claim. Another way to think of this would be that the evaluator speculates as to whether a heuristic would lead to the discovery of the issue in the claim (if the claim had not been presented to the evaluator). He or she then indicates how much they think a heuristic applies to the claim by marking the corresponding point on the Likert scale.

5.3.3 Questionnaire

As mentioned earlier, the evaluators in this experiment provided their feedback through a Likert scale, with agreement ratings for each of the heuristics in the set. In addition to this feedback, each evaluator also rated the claim in terms of how much they felt it actually applied to the interface in question. By indicating their agreement level with the claim to the interface, we get feedback on whether usability experts actually think the claim is appropriate for the interface in question. This feedback helps us in identifying the issues that truly seem to apply to the interface, according

to multiple experts. They provided this claim applicability feedback before rating the heuristics. Appendix G provides a copy of the questionnaire.

After rating each of the heuristics for the claims, we also asked each evaluator to rank the severity that the claim would hold, if the claim were indeed a usability problem in the interface. This question asked the evaluators to rely on their expertise in usability evaluation in order to rank the claim as a usability problem. Rankings ranged from "no problem" to "most severe", with the latter indicating the problem must be fixed and the former indicating that the claim does not represent a usability issue in the interface.

5.3.4 Measurements Recorded

The data collected in this experiment consists of each evaluator's rating of the claim applicability, each heuristic rating for an individual claim, and the evaluator's assessment of the severity of the usability problem. This data was collected for each of the thirty claims across the three systems.

In addition to the above measures, we also collected data on the evaluator's experience with usability evaluation, heuristics, and large screen information exhibits. This evaluator information was collected through survey questions before the evaluation was started.

After the evaluators completed the test, they recorded the amount of time they spent on the task. This was a self reported value as each evaluator worked at his/her own pace and in their own location.

Data collection was done through pen and paper. Each claim was presented on a single sheet of paper, along with the questions about the applicability of the claim to the interface and the severity of the problem. In addition, each of the heuristics was listed on the same sheet, adjacent to the claim. This setup allowed the evaluator to consider the claim for each of the heuristics and subsequently rate how much they felt the heuristic applied to the claim. As mentioned earlier, each heuristic was rated on a 7-point Likert scale, indicating the evaluators level of agreement that the heuristic applied to the claim. Agree ratings meant that the heuristic somehow related to the issue in the claim, through the associated upside or downside tradeoffs. Disagree ratings meant that the heuristic did not really relate to the claim.

5.4 Results

Twenty-one evaluators provided feedback on 33 different claims across three systems. Each evaluator ended up providing either 10 or 12 question responses per claim, depending on the heuristic set used (Nielsen's set has 10 in it, whereas the others only have 8). This means we have either 330 or 396 answers to consider, per evaluator. Fortunately, this data was separable into manageable chunks, dealing with applicability, severity, and heuristic ratings; as well as evaluator experience levels and time to complete for each method.

5.4.1 Participant Experience

As for individual evaluator abilities, the average experience level with usability evaluation, across all three systems, was "amateur". This means that overall, for each heuristic set, we had comparable experience for the evaluators assigned to that set. This feedback was determined from a

Participant	Set	Usability	Heuristic	LSIE
1	Nielsen	expert	amateur	novice
2	Nielsen	expert	amateur	amateur
3	Nielsen	expert	expert	novice
4	Nielsen	amateur	amateur	novice
5	Nielsen	amateur	amateur	novice
6	Nielsen	amateur	novice	novice
7	Nielsen	expert	amateur	novice
8	Berry	amateur	novice	novice
9	Berry	expert	amateur	novice
10	Berry	expert	amateur	novice
11	Berry	amateur	expert	novice
12	Berry	amateur	amateur	novice
13	Berry	expert	amateur	expert
14	Berry	expert	amateur	amateur
15	Somervell	expert	amateur	amateur
16	Somervell	amateur	amateur	novice
17	Somervell	amateur	amateur	novice
18	Somervell	amateur	amateur	novice
19	Somervell	amateur	amateur	novice
20	Somervell	amateur	amateur	novice

Table 5.2: Evaluator experience with usability evaluation. Amateur means they had knowledge of usability evaluation and had performed at least one such evaluation. Novice means that the evaluator was only familiar with the concept of usability evaluation. Expert means the evaluator had performed two or more evaluations.

question asking the evaluator to rate his/her experience with usability evaluation. They answered the question with either novice, amateur, or expert. This answer was then coded with a 1, 2, or 3 so that averages and standard deviations could be taken. Table 5.2 lists the self-reported experience level for each evaluator, with their assignments to the systems. Thus, we are confident that the overall usability evaluation experience levels of the evaluators was equal or near equal across the three heuristic sets.

What about experience with heuristic evaluation or experience with large screen information exhibits? Feedback on these questions was given as responses to similar questions as the usability evaluation experience. Again, evaluators gave their self-assessment rating of their experience level with heuristic evaluation and with large screen information exhibits. Table 5.2 gives the ratings of each evaluator. Heuristic experience was equal or near equal across all three heuristic sets. Experience with large screen information exhibits was likewise equal or near equal. Figure 5.4 provides a summary of the evaluators' experience levels with respect to usability evaluation, heuristic evaluation, and large screen information exhibits by system.

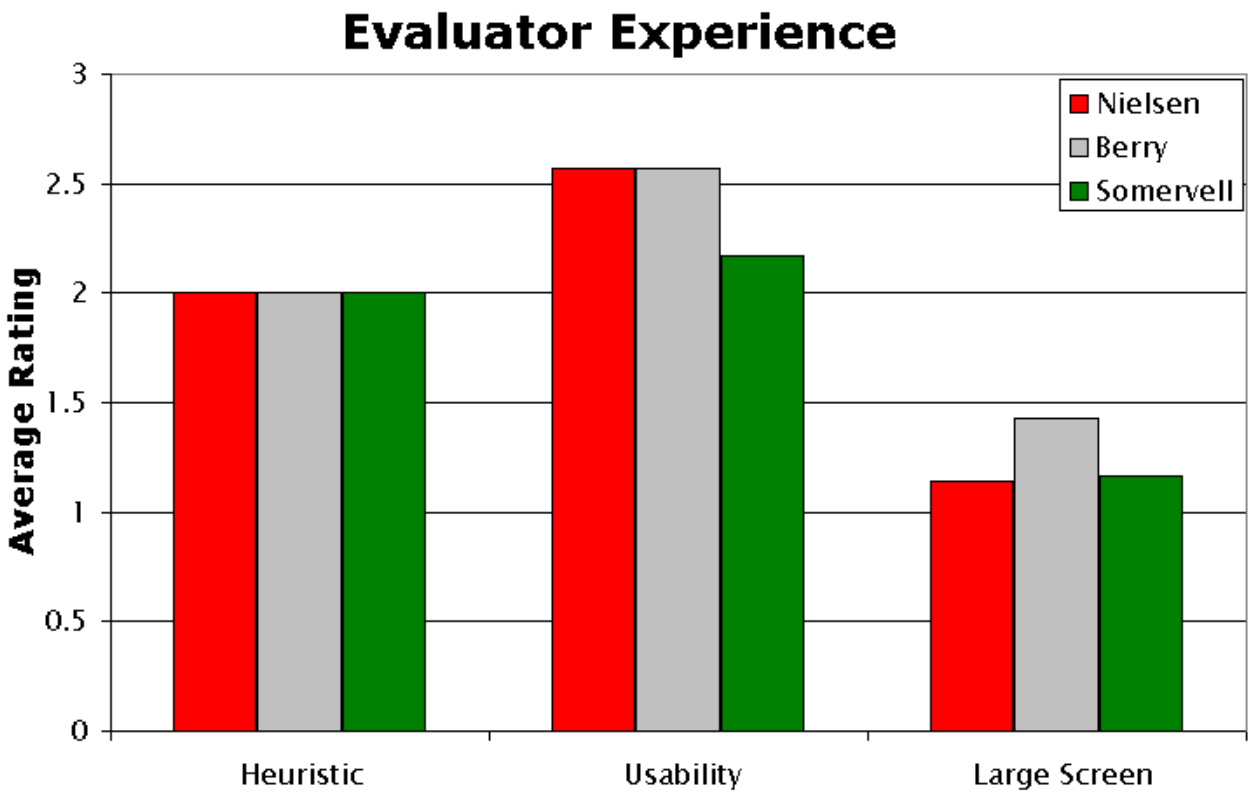

Figure 5.4: Summary of evaluator experience with usability evaluation, heuristic, evaluation, and large screen information exhibits.

5.4.2 Applicability Scores

To indicate whether or not a heuristic set applied to a given claim (or problem), evaluators marked their agreement with the statement "the heuristic applies to the claim". This agreement rating indicates that a specific heuristic applied to the claim. Each of the heuristics was marked on a 7-point Likert scale by the evaluators, indicating his/her level of agreement with the statement.

Using this applicability measure, the responses were averaged for a single claim across all of the evaluators. Averaging across evaluators allows assessment of the overall "applicability" of the heuristic to the claim. This applicability score is used to determine whether any of the heuristics applied to the issue described in the claim. If a heuristic received an "agree" rating, average greater than or equal to five, then that heuristic was thought to have applied to the issue in the claim.

Overall Applicability

Considering all 33 claims together (found in all three systems), one-way analysis of variance (ANOVA) indicates significant differences among the three heuristic sets for applicability ($F(2, 855) = 3.0, MSE = 49.7, p < 0.05$). Further pair-wise t-tests reveal that Somervell's set of heuristics had significantly higher applicability ratings over both Berry's ($df = 526, t = 3.32, p < 0.05$) and Nielsen's sets ($df = 592, t = 11.56, p < 0.05$). In addition, Berry's heuristics had significantly higher applicability scores over Nielsen's set ($df = 592, t = 5.94, p < 0.05$). Table 5.3 provides

Anova: Single Factor

SUMMARY

Groups	*Count*	*Sum*	*Average*	*Variance*
Berry	264	1056.831	4.003148	1.441069
Nielsen	330	1150.933	3.487677	0.836816
Somervell	264	1133.033	4.291793	0.550076

ANOVA

Source of Variation	*SS*	*df*	*MS*	*F*	*P-value*	*F crit*
Between Groups	99.40269	2	49.70134	53.18589	1.7E-22	3.006249
Within Groups	798.9835	855	0.934484			
Total	898.3862	857				

Table 5.3: Summary of ANOVA for overall applicability. This includes all 33 claims from the three systems.

the ANOVA summary.

Plasma Poster

Somervell's heuristics had the highest applicability scores of the three sets. One-way analysis of variance (ANOVA) indicates a significant difference in scores for the three heuristic sets for each of the systems ($F(2, 361) = 3.02, MSE = 34.16, p < 0.05$ for Plasma Poster). Pairwise t-tests indicate that Somervell's heuristics have significantly higher applicability scores over both Nielsen's ($df = 250, t = 10.33, p < 0.05$) and Berry's sets ($df = 222, t = 3.30, p < 0.05$). Similarly, Berry's set was rated significantly higher than Nielsen's set ($df = 250, t = 4.75, p < 0.05$). Thus, Somervell's heuristics have the highest applicability for the Plasma Poster. Figure 5.5 shows the applicability scores for the three heuristic sets for the Plasma Poster.

Notification Collage

Similar analysis for the Notification Collage yields slightly different results. One-way ANOVA indicates significant differences across the three heuristic sets ($F(2, 205) = 13.93, MSE = 12.77, p < 0.05$). Pairwise t-tests show that both Somervell's and Berry's heuristics have significantly higher applicability scores over Nielsen's set ($df = 142, t = 5.14, p < 0.05$ and $df = 142, t = 3.80, p < 0.05$ respectively). However, there was no significant difference between the applicability scores for Somervell's and Berry's ($df = 126, t = 0.76, p = 0.44$). Figure 5.5 shows this graphically.

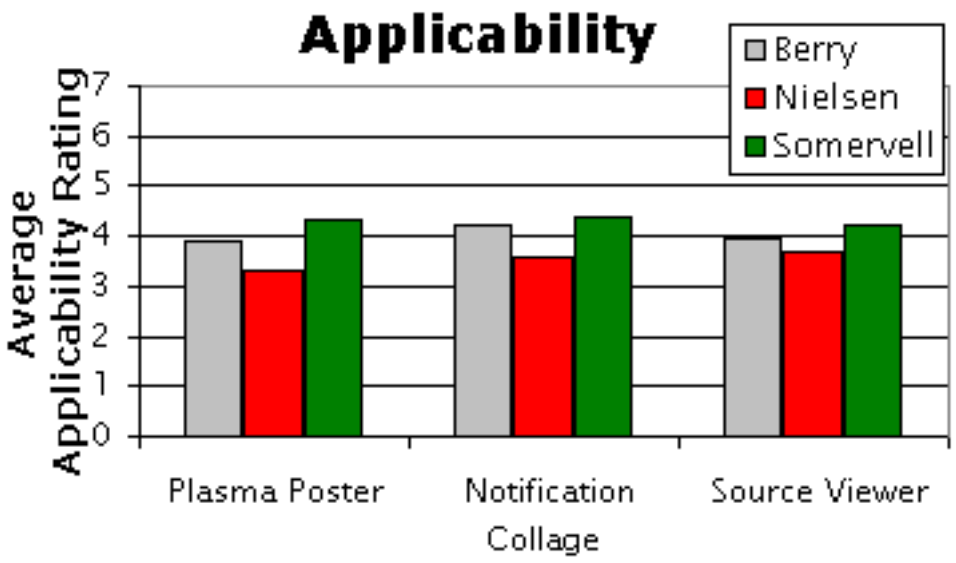

Figure 5.5: Applicability scores for each heuristic set by system.

Source Viewer

Source Viewer analysis is similar to the other two systems. One-way ANOVA indicates significant difference in applicability scores for the three heuristic sets ($F(2, 283) = 6.96), MSE = 7.0, p < 0.05$). However, we only find significant differences from t-tests for Somervell's compared to Nielsen's, with Somervell's significantly higher ($df = 196, t = 4.42, p < 0.05$). T-tests do not indicate significance between Somervell and Berry ($df = 174, t = 1.4, p = 0.16$) nor between Berry and Nielsen ($df = 196, t = 1.94, p = 0.05$). Figure 5.5 summarizes these results.

5.4.3 Thoroughness

Recall that thoroughness is measured as the number of problems found by a single method, divided by the number uncovered by all of the methods. This requires a breakdown of the total number of claims into the numbers for each system. Plasma Poster has 14 claims, Notification Collage has eight claims, and the Source Viewer has 11 claims. We look at thoroughness measures for each system. To calculate the thoroughness measures for the data we have collected, we count the number of claims "covered" by the target heuristic set. Here we are defining covered to mean that at least one of the heuristics in the set had an average agreement rating of at least five. Why five? On the Likert scale, five indicates somewhat agree. If we require that the average score across all of the evaluators to be greater than or equal to five for a single heuristic, we are only capturing those heuristics that truly apply to the claim in question.

Overall Thoroughness

Across all three heuristic sets, 28 of 33 claims had applicability scores higher than five. Somervell's heuristics had the highest thoroughness rating of the three heuristic sets with 96% (27 of 28 claims). Berry's heuristics came next with a thoroughness score of 86% (24 of 28) and Nielsen's heuristics

had a score of 61 (17 of 28)

Test of proportions [1] reveals significant differences between Somervell's heuristics and Nielsen's heuristics ($z = 3.26, p = 0.001$). Berry's heuristics also showed significance when compared to Nielsen's heuristics ($z = 2.11, p = 0.04$). No significant difference was found between Somervell's heuristics and Berry's heuristics ($z = 1.41, p = 0.16$)

The following subsections detail the thoroughness scores as broken down for each of the three systems.

Plasma Poster

With 14 claims, the Plasma Poster had the highest number of issues of the three systems. Across all three heuristic sets, 11 of 14 issues were covered. In other words, if we take the number of issues covered (average ratings higher than 5 for at least one heuristic) by all three heuristic sets, we come up with a total of 11. Somervell's heuristics applied to the most problems, with 11, Berry's heuristics were next with eight, and Nielsen's heuristics applied to three. So for thoroughness, this means that Somervell's heuristics had a thoroughness rating of 100%, Berry's heuristics had a rating of 73%. and Nielsen's heuristics had a thoroughness rating of 27%. Figure 5.6 shows the thoroughness scores for all three heuristic sets.

Test of proportions reveals significant differences between Somervell's heuristics and Nielsen's heuristics ($z = 3.55, p < 0.05$). Berry's heuristics also showed significantly higher thoroughness over Nielsen's set ($z = 2.13, p = 0.03$). No significant differences were found between Somervell's heuristics and Berry's heuristics ($z = 1.86, p = 0.06$).

Notification Collage

The Notification Collage had the least number of claims of the three systems with eight. Across all three heuristic sets, seven of these claims were covered. Nielsen's heuristics applied to six of the eight claims, yielding a thoroughness score of 86%. Berry's heuristics also applied to six of the eight claims, with an 86% thoroughness score. Somervell's heuristics applied to seven claims, hence it received a thoroughness score of 100%. Figure 5.6 shows these scores. Test of proportions reveals no significant differences in thoroughness scores among the three heuristic sets for the Notification Collage.

Source Viewer

The Source Viewer had 11 claims associated with it. Of these, 10 were determined to have applicability from across the three heuristic sets. Somervell's heuristics applied to nine of those 10 issues, for a thoroughness rating of 90%. Nielsen's heuristics applied to eight, for a thoroughness score of 80%. Berry's heuristics applied to 10 of the 10 claims, for a thoroughness score of 100%. Again, these scores are shown in Figure 5.6. Test of proportions indicates no significant differences in thoroughness scores for the Source Viewer.

[1] Test of proportions is an accepted statistical test for determining significant differences between proportions [49].

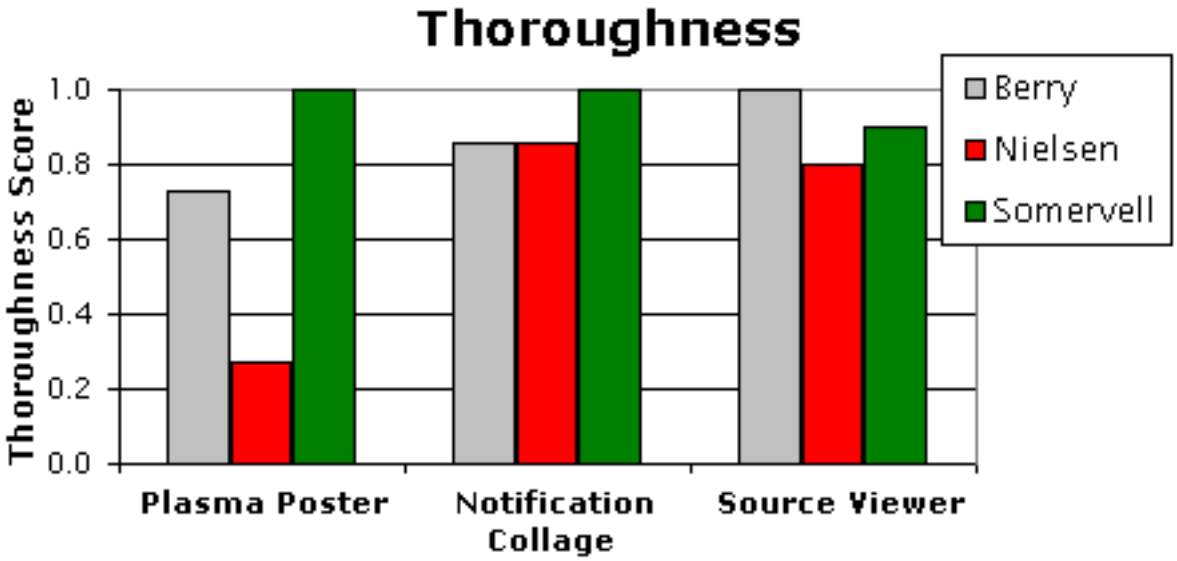

Figure 5.6: Thoroughness scores for each method and system.

5.4.4 Validity

Validity measures the UEM's ability to uncover real usability problems in a system [40]. Here the full set of problems in the system is used as the real problem set (as discussed in earlier sections). As with thoroughness, the applicability scores determine the validity each heuristic set held for the three systems. As before, we used the cutoff value of five on the Likert scale to indicate applicability of the heuristic to the claim. An average rating of five or higher indicates that the heuristic applied to the claim in question.

Overall Validity

Similar to thoroughness, validity scores were calculated across all three systems. Out of 33 total claims, only 28 showed applicability scores greater than five across all three heuristic sets. Somervell's heuristics had the highest validity, with 27 of 33 claims yielding applicability scores greater than five, for a validity score of 82%. Berry's heuristics had the next highest validity with 24 of 33 claims, for a validity score of 73%. Nielsen's heuristics had the lowest validity score, with 17 of 33 claims for a score of 52%.

Test of proportions reveals significant differences between Somervell's heuristics and Nielsen's heuristics ($z = 2.61, p = 0.01$). No significant differences were found between Berry's heuristics and Nielsen's heuristics ($z = 1.78, p = 0.08$), nor between Somervell's heuristics and Berry's heuristics ($z = 0.88, p = 0.38$).

The following subsections provide details on the breakdown in validity scores across the three systems.

Plasma Poster

Recall that Plasma Poster had 14 claims in the real problem set. Nielsen's heuristics applied to three of the 14 issues in the Plasma Poster, thus yielding a validity score of 21%. Berry's heuristics applied to eight of the 14, for a validity score of 57%. Somervell's heuristics applied to 11 of the 14, for a validity score of 79%. These scores are in alignment with what was found in the thoroughness measures. Figure 5.7 provides a graphical view of these scores.

Test of proportions reveals significant differences in validity between Somervell's heuristics and Nielsen's heuristics ($z = 3.02, p < 0.05$). However, no significant differences were found between Somervell's heuristics and Berry's heuristics ($z = 1.21, p = 0.22$) nor between Berry's heuristics and Nielsen's heuristics ($z = 1.93, p = 0.05$).

Notification Collage

The general trend reported in the Plasma Poster was also found in the Notification Collage. Nielsen's heuristics applied to six of the eight issues with the system, giving it a validity score of 75%. Berry's heuristics also applied to six of the eight issues, with a validity score of 75%. Somervell's heuristics performed best, applying to seven of the eight issues, yielding a validity score of 88%. See Figure 5.7 for a graphical comparison of these scores. Test of proportions does not indicate significant differences in these validity scores.

Source Viewer

As with the thoroughness measure, validity did not follow the pattern observed so far. Nielsen's set of heuristics applied to eight of the 11 claims, producing a validity score of 73%. Berry's set applied to 10 of the 11 issues, giving that set a validity score of 91%. Somervell's only applied to nine of the 11 issues, with a validity score of 82%. Figure 5.7 gives a graphical depiction of these scores. Test of proportions indicates no significant differences in these scores.

5.4.5 Effectiveness

Effectiveness is calculated by multiplying thoroughness by validity. UEMs that have high thoroughness and high validity will have high effectiveness scores. A low score on either of these measures will reduce the effectiveness score.

Overall Effectiveness

Considering the effectiveness scores across all three systems reveals that Somervell's heuristics had the highest effectiveness with a score of 0.79. Berry's heuristics came next with a score of 0.62. Nielsen's heuristics had the lowest overall effectiveness with a score of 0.31.

Plasma Poster

For the Plasma Poster, Somervell's heuristics had the highest effectiveness scores (0.79). Berry's had the next highest effectiveness score (0.42), and Nielsen's heuristics had the lowest effectiveness score (0.06). Figure 5.8 provides a graphical depiction.

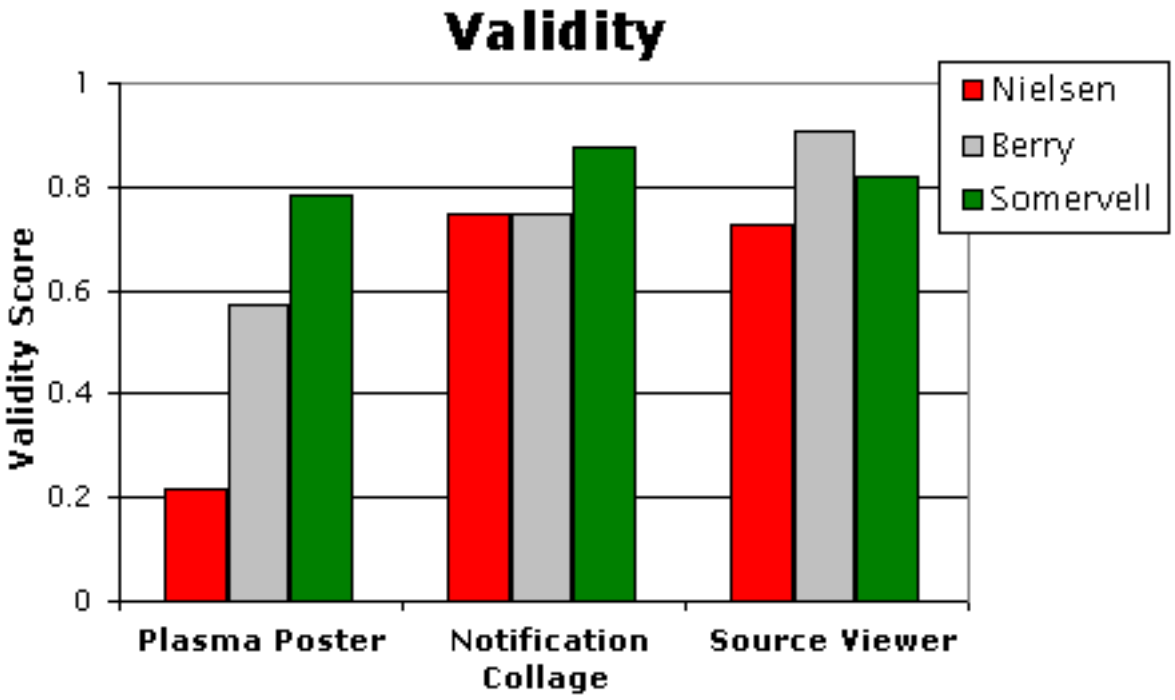

Figure 5.7: Validity scores for the three heuristics sets for each system.

Notification Collage

As expected, Somervell's heuristics had the highest effectiveness scores for the Notification Collage (0.88). Berry's heuristics and Nielsen's heuristics had the same effectiveness score (0.64) [see Figure 5.8].

Source Viewer

As observed in the thoroughness and validity measures, Berry's heuristics also had the highest effectiveness score for the Source Viewer (0.91). Somervell's heuristics had the next highest effectiveness score (0.74). Nielsen's heuristics had the lowest effectiveness score (0.58). Figure 5.8 shows the effectiveness scores for the Source Viewer.

5.4.6 Reliability – Differences

Recall that the reliability of each heuristic set is measured in two ways: one relying upon the actual differences among the evaluators, the other upon the average number of agreements among the evaluators. Here we focus on the former. For example, Berry's set has eight heuristics, so consider calculating the differences in the ratings for the first heuristic for the first claim in the Plasma Poster. This difference is found by subtracting the ratings of each evaluator from every other evaluator and summing up each of the differences, then dividing by the number of differences (or the average difference). Suppose that an evaluator rated the first heuristic with a 6 (agree) and another rated it as a 4 (neutral) and a third rated it as a 5 (somewhat agree). The difference in this

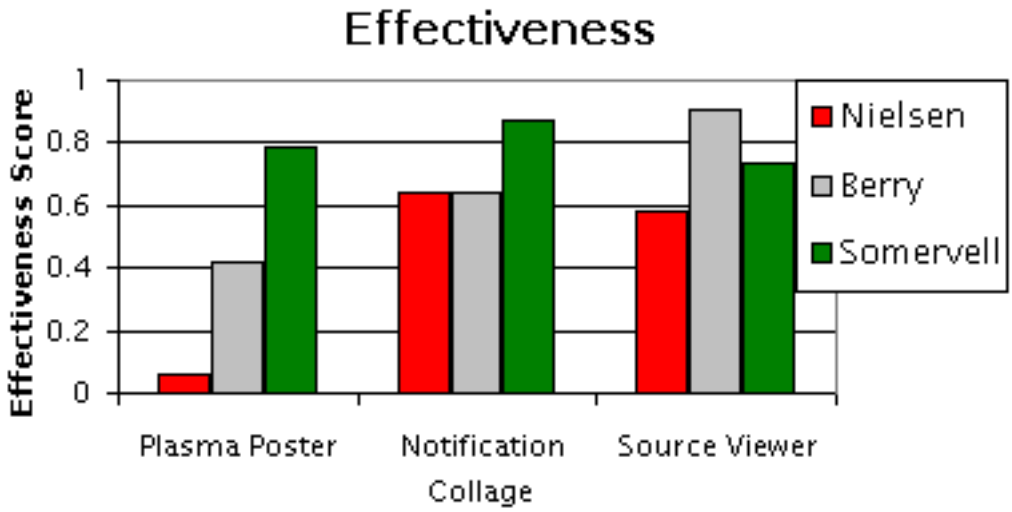

Figure 5.8: Effectiveness scores for each system. Somervell's heuristics had consistently high effectiveness.

particular instance would be:

$$\frac{(6-4)+(6-5)+(5-4)}{3} = 1.33$$

We then averaged the differences for every heuristic on a given claim to get an overall difference score for that claim, with a lower score indicating higher reliability (zero difference indicates complete reliability). These average differences provide a measure for the reliability of the heuristic set.

Overall Reliability Differences

Considering all 33 claims across the three systems gives an overall indication of the average differences for the heuristic sets. One-way ANOVA suggests significant differences among the three heuristic sets ($F(2,23) = 23.02, MSE = 0.84, p < 0.05$). Pair-wise t-tests show that Somervell's heuristics had significantly lower average differences than both Berry's heuristics ($df = 14, t = 4.3, p < 0.05$) and Nielsen's heuristics ($df = 16, t = 6.8, p < 0.05$). No significant differences were found between Berry's heuristics and Nielsen's heuristics ($df = 16, t = 1.43, p = 0.17$), but Berry's set had a slightly lower average difference ($M_B = 2.02, SD_B = 0.21$; $M_N = 2.14, SD_N = 0.13$). Figure 5.9 shows the overall average evaluator differences for the three heuristic sets.

Plasma Poster

Focusing on the Plasma Poster, one-way analysis of variance (ANOVA) suggests significant differences in the reliability of the three heuristic sets ($F(2,23) = 21.7, MSE = 0.83, p < 0.05$). Further pairwise t-tests show that Somervell's set had the lowest average difference ($M = 1.58, SD = 0.23$), with significance at the $\alpha = 0.05$ level over the other two methods [$df = 16, t = 6.5, p <$

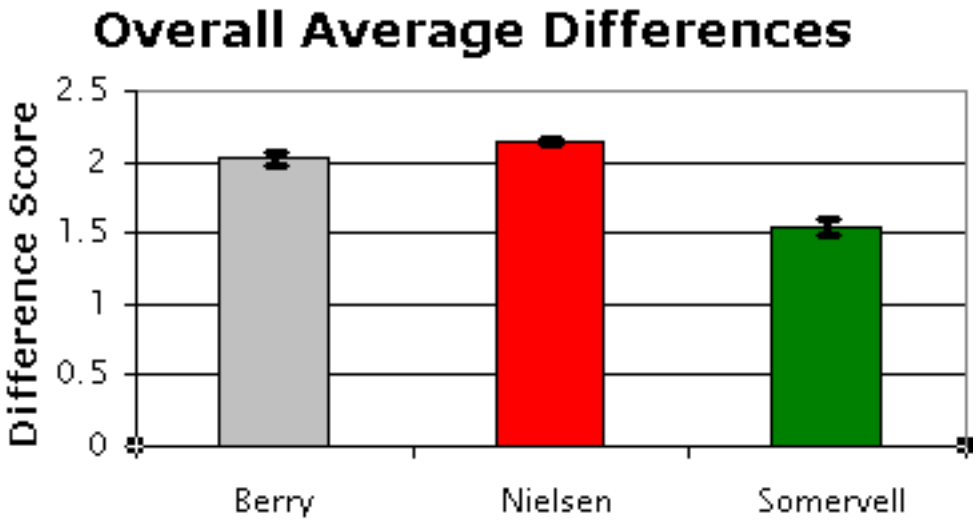

Figure 5.9: Overall average evaluator differences for each heuristic set, with standard deviation error bars. Somervell's heuristics had the lowest average difference, which means that set had better reliability when considering all of the claims across the three systems.

0.05 for Nielsen's set ($M = 2.18, SD = 0.16$) and $df = 16, t = 1.85, p = 0.08$ for Berry's ($M = 2.02, SD = 0.20$)]. Berry's set also showed significance over Nielsen's set (with $df = 26, t = 15.24, p < 0.05$). So for the Plasma Poster, Somervell's heuristics had the lowest average difference. See Figure 5.10 for a graphical representation.

Notification Collage

One-way ANOVA on the average differences of the three heuristic sets for the Notification Collage indicates significant differences ($F(2, 23) = 7.03, MSE = 0.44, p < 0.05$). Using pairwise t-tests, Somervell's set holds significantly higher reliability ($M = 1.65, SD = 0.29$) over both Nielsen's ($M = 2.062, SD = 0.26$) and Berry's ($M = 2.058, SD = 0.18$) with $df = 16, t = 3.06, p = 0.01$ for Nielsen and $df = 14, t = 3.29, p = 0.01$ for Berry. Berry's heuristics and Nielsen's heuristics had very similar reliability ratings and there is no significant difference in reliability between these two heuristic sets ($df = 16, t = 0.04, p = 0.97$). See Figure 5.10 for a graphical representation.

Source Viewer

One-way ANOVA suggests significant differences in reliability of the three heuristic sets for the Source Viewer ($F(2, 23) = 17.78, MSE = 1.22, p < 0.05$). Further analysis using pairwise t-tests reveals that Somervell's heuristics had significantly lower average differences ($M = 1.42, SD = 0.08$) than both Nielsen's ($M = 2.14, SD = 0.03$) and Berry's ($M = 2.0, SD = 0.10$) sets (with $df = 16, t = 6.48, p < 0.05$ for the former and $df = 14, t = 3.78, p < 0.05$ for the latter). T-tests do not show significant differences between Nielsen's and Berry's heuristic sets ($df = 16, t = 1.16, p = 0.26$). Figure 5.10 provides a graphical summary of the reliability scores.

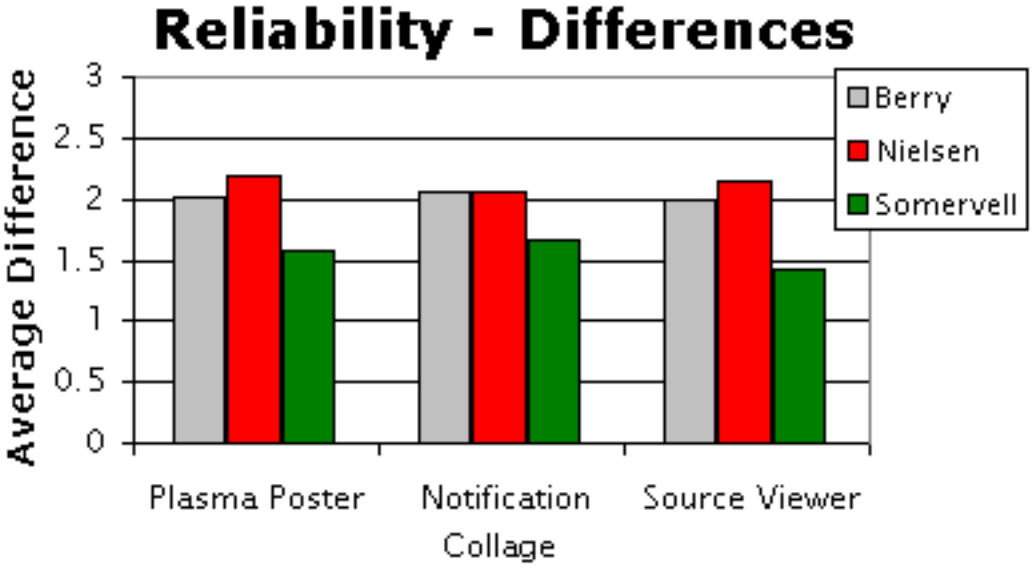

Figure 5.10: Average difference scores for each method by system. Lower differences indicate higher reliability.

5.4.7 Reliability – Agreement

In addition to the average differences, a further measure of reliability was calculated by counting the number of agreements among the evaluators, then dividing by the total number of possible agreements. This calculation provides a measure of the agreement rating for each heuristic. For example, consider the previous three evaluators and their ratings (6, 5, and 4). The agreement rating in this case would be:

$$agreement = \frac{0}{3} = 0$$

because none of the evaluators agreed on the rating, but there were potentially three agreements (if they had all given the same rating). Averages across all of the claims for a given system were then taken. This provides an assessment of the average agreement for each heuristic as it pertains to a given system.

Overall Agreement

Taking all 33 claims into consideration, one-way ANOVA indicates significant differences among the three heuristic sets for evaluator agreement ($F(2, 23) = 6.31, MSE = 0.01, p = 0.01$). Pairwise t-tests show that both Somervell's heuristics and Berry's heuristics had significantly higher agreement than Nielsen's set ($df = 16, t = 2.99, p = 0.01$ and $df = 16, t = 3.7, p < 0.05$ respectively). No significant differences were found between Somervell's and Berry's heuristics ($df = 14, t = 0.46, p = 0.65$). Figure 5.11 shows the average evaluator agreement for each heuristic set across all three systems.

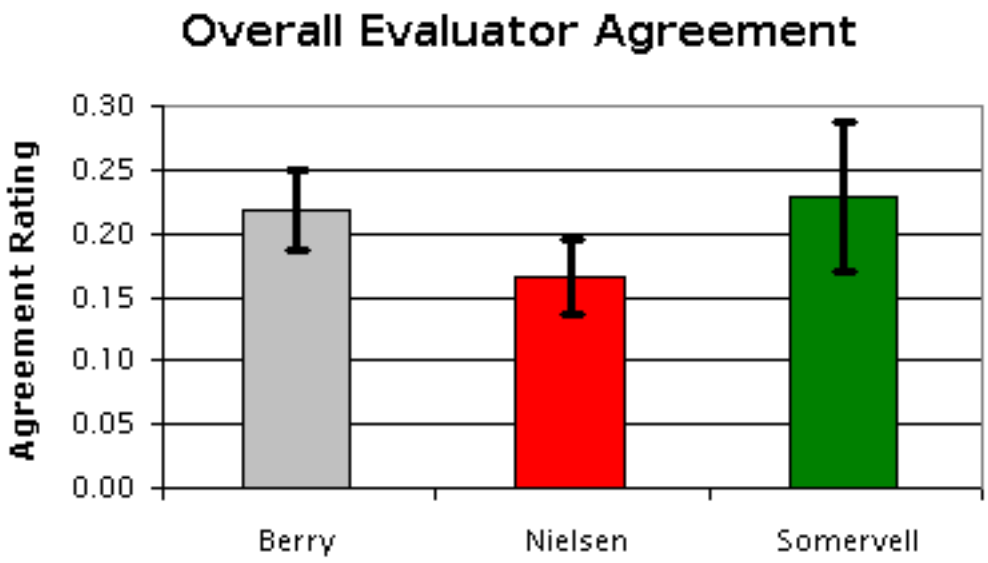

Figure 5.11: Overall average evaluator agreement for the three heuristic sets. Error bars represent one standard deviation from the means. Somervell's set had the best evaluator agreement, whereas Nielsen's set had the least.

Plasma Poster

One-way ANOVA for agreement on the Plasma Poster suggests significant difference among the three heuristic sets ($F(2,23) = 4.58, MSE = 0.01, p = 0.02$). Pair-wise t-tests show that Somervell's heuristics ($M = 0.22, SD = 0.07$) had significantly higher agreement ratings than Nielsen's heuristics ($M = 0.17, SD = 0.03$) with $df = 16, t = 2.17, p < 0.05$. Berry's heuristics also held higher agreement ratings ($M = 0.23, SD = 0.03$) with $df = 16, t = 4.23, p < 0.05$. No significant differences were found between Somervell's heuristics and Berry's heuristics. See Figure 5.12 for graphical depiction.

Notification Collage

One-way ANOVA for agreement on the Notification Collage does not reveal significant difference among the three heuristic sets ($F(2,23) = 2.82, MSE = 0.01, p = 0.08$). However, Somervell's heuristics had a slightly higher average agreement rate ($M = 0.21, SD = 0.06$) over Nielsen's ($M = 0.16, SD = 0.04$) and Berry's ($M = 0.21, SD = 0.03$). Figure 5.12 shows a graphical representation.

Source Viewer

One-way ANOVA for agreement on the Source Viewer suggests significant difference among the three heuristic sets ($F(2,23) = 6.81, MSE = 0.021, p < 0.05$). Pair-wise t-tests show that Somervell's heuristics ($M = 0.26, SD = 0.0.07$) had significantly higher agreement ratings than Nielsen's heuristics ($M = 0.16, SD = 0.04$) with $df = 16, t = 3.56, p < 0.05$. Berry's heuristics also held significantly higher agreement ratings ($M = 0.23, SD = 0.03$) over Nielsen's set, with

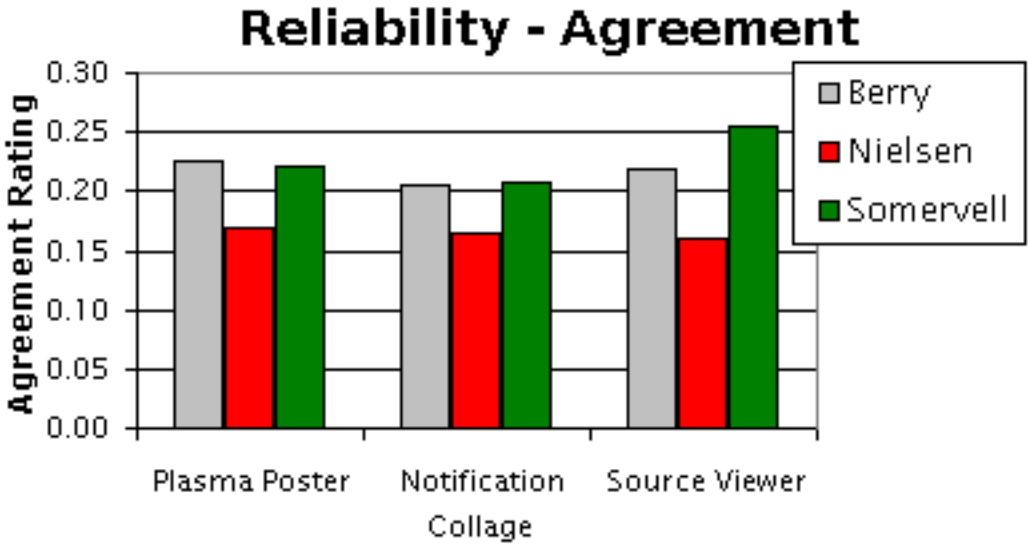

Figure 5.12: Evaluator agreements for the three heuristic sets, shown by system. Note that Somervell's heuristics had consistently high evaluator agreement across all three systems.

$df = 16, t = 2.62, p = 0.02$. No significant differences were found between Somervell's heuristics and Berry's heuristics. Figure 5.12 shows these findings.

5.4.8 Time Spent

Recall that we also asked the evaluators to report the amount of time (in minutes) they spent completing this evaluation. This measure is valuable in assessing the cost of the methods in terms of effort required. It was anticipated that the time required for each method would be similar across the methods.

Averaging reported times across evaluators for each method suggests that Somervell's set required the least amount of time ($M = 103.17, SD = 27.07$), but one-way ANOVA reveals no significant differences ($F(2, 17) = 0.26, p = 0.77$). Berry's set required the most time ($M = 119.14, SD = 60.69$) while Nielsen's set ($M = 104.29, SD = 38.56$) required slightly more than Somervell's. Figure 5.13 provides a graphical representation of these times.

5.5 Discussion

So what does all this statistical analysis mean? What do we know about the three heuristic sets? How have we supported or refuted our hypotheses through this analysis? We address these questions in the following sections.

5.5.1 Hypotheses Revisited

Recall that we had three hypotheses for this experiment:

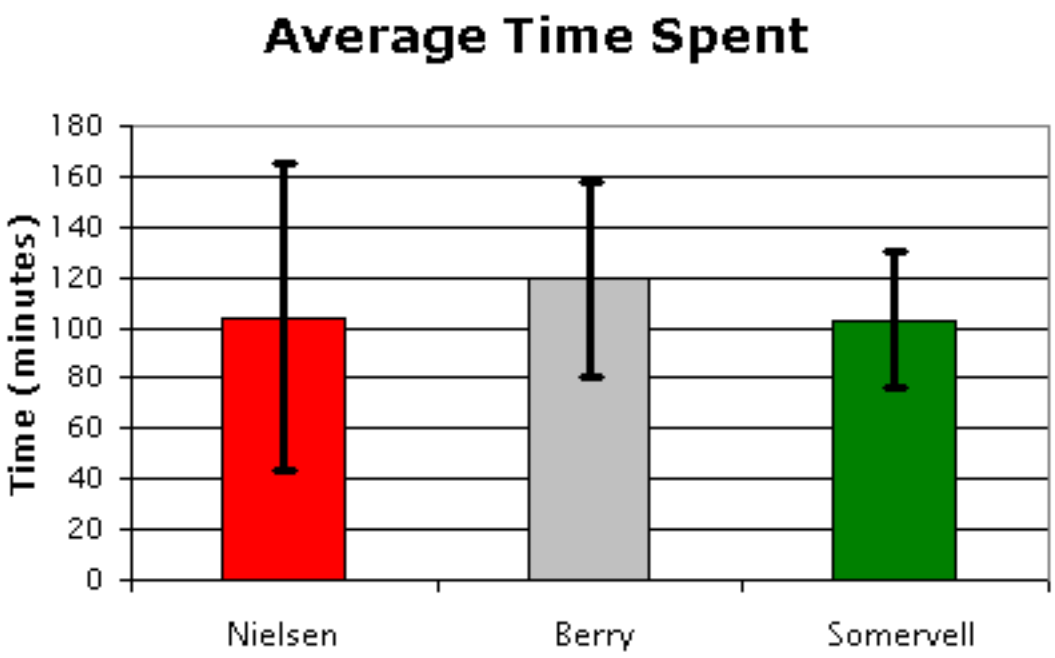

Figure 5.13: Average time to complete evaluations with each heuristic set.

1. **Somervell's set of heuristics will have a higher validity score for the Notification Collage.**
2. **More specific heuristics will have higher thoroughness, validity, and reliability measures.**
3. **Generic methods will require more time for evaluators to complete the study.**

We discuss the statistical analysis in terms of these three hypotheses in the following sections.

Hypothesis 1

For hypothesis one, we discovered that Somervell's heuristics indeed held the highest validity score for the Notification Collage (see Figure 5.7). However, this validity score was not 100%, as was expected. What does this mean? It simply illustrates the difference in the evaluators who participated in this study. They did not think that any of the heuristics applied to one of the claims from the Notification Collage. Although, it can be noted that the applicability scores for that particular claims were very close to the cutoff level we chose for agreement (that being 5 or greater on a 7-point scale). Still, evidence suggests that hypothesis 1 holds.

Implications Since Somervell's set had the highest validity of the three heuristic sets for the Notification Collage, we have further validation of that set. Why? In the creation process (Chapter 4) we used the Notification Collage as input. It seems logical that the heuristic set developed in that effort should find most or all of the problems with one of the systems used in that creation process. We found evidence of this case in this study. Had Somervell's method not produced the

highest validity score for the Notification Collage, one could raise questions about the soundness of Somervell's set of heuristics. As it stands, we have verified that Somervell's method works on at least a subset of the systems from which it came: a kind of validation of the method and the heuristics.

Hypothesis 2

We find evidence to support this hypothesis based on the scores on each of the three measures: thoroughness, validity, and reliability. In each case, more specific methods had the better ratings over Nielsen's heuristics for each measure. Overall one could argue that Somervell's set of heuristics is most suitable for evaluating large screen information exhibits, but must concede that Berry's heuristics could also be used with some effectiveness.

Specifically, the Source Viewer results suggest that there was a slight turn around in the otherwise consistent trend observed in the methods. For this system, we found that Berry's heuristics actually applied to more of the claims than both Somervell's and Nielsen's heuristics. Although we cannot be entirely sure why this anomaly was observed, it does not weaken the hypothesis much because the statistical differences were not significant. Possible reasons why this anomaly occurred could be due to evaluator differences, the actual problems for that particular system, or even the nature of problem collection. Since the differences were not significant, post-facto analysis can only suggest a possible cause, and it does not impact the overall findings of this work.

It should be noted that even though Somervell's heuristics came behind Berry's set for the Source Viewer in terms of thoroughness, validity, and effectiveness; Somervell's heuristics had better reliability, both in terms of evaluator differences, and evaluator agreement. So, even though Berry's heuristics had higher thoroughness and validity for the Source Viewer, Somervell's heuristics were very close and had higher reliability for that system. This allows us to claim that Somervell's heuristics are best suited for the LSIE system class.

Implications This result is by far the most compelling. We have shown through rigorous testing that Somervell's method is the best heuristic choice for evaluating large screen information exhibits. Coupled with the validation results from Chapter 6, we show that this method is both easy to use in analytic evaluations of these systems, as well as effective (in terms of thoroughness, reliability, and validity) at finding usability problems.

In other words, for a given system class, it is worthwhile and useful to create an evaluation method that is tailored to the goals of that system class, yet is still generic enough to apply to many systems within that class. This finding does not contradict efforts from Mankoff et al. [56] and Baker et al. [5] in which they have shown that similar evaluation tools, corresponding to system-class specificity, actually perform better than generic sets.

Hypothesis 3

We did not find evidence to support this hypothesis. As reported, there were no significant differences in the times required to complete the evaluations for the three methods. However, Somervell's and Nielsen's sets took about 15 fewer minutes, on average, to complete. This does not indicate that the more generic method (Nielsen's) required more time. So what would cause the evaluators to take more time with Berry's method? Initial speculation would suggest that this set uses

terminology associated with Notification Systems [62] (see Figure 5.2 for listing of heuristics), including reference to the critical parameters of interruption, reaction, and comprehension, and thus could have increased the interpretation time required to understand each of the heuristics.

Implications Since we did not find any statistical differences in the amount of time required to complete the evaluation with each of the three heuristic sets, we cannot claim that one method is more cost effective over the other methods (with respect to time spent). This is unfortunate, as it would have been compelling to have Somervell's method also the one to require the least amount of time. While on average, Somervell's method did have the lowest time, Nielsen's method on average only required about 1 minute longer. This difference is negligible.

It is encouraging however, that both Berry's and Somervell's methods required similar amounts of time to that of Nielsen's. As mentioned before, Nielsen's method has been around for almost 20 years and has seen extensive use and analysis. Therefore it is promising that the more specific methods do not require significantly more time when used in an analytic evaluation; suggesting that these more specific methods could reach acceptance levels comparable to Nielsen's.

5.5.2 Other Discussion and Implications

In addition to the insight provided through analysis of the hypotheses, we can also discuss some of the caveats and limitations of this experiment and its findings, in terms of how it can impact and support various groups, including usability professionals and UEM researchers.

Implications for Usability Specialists

This study has shown that system-class specific UEMs are desirable for formative usability evaluation. What this implies for evaluators, especially for those who are evaluating notification systems, is that some up-front effort should be expended for developing a set of tools that can be reused in multiple evaluations of similar systems.

Somervell's heuristics are an example of this level of specificity. They are tailored to the user goals associated with the LSIE system class, yet are applicable across multiple systems within that class. The other areas of the notification systems design space are other potential system classes that could benefit from tailored UEM development.

For example, consider the secondary display system class. These interfaces try to support rapid reaction to and high comprehension of information, while simultaneously being non-interruptive. A major news agency like CNN or Fox News would be highly interested in providing the most effective designs for their tickers and faders. If they had tailored heuristics to help evaluate their designs, they could maximize utility while minimizing cost. Each of the other system classes described by the notification design space could have a tailored UEM, thus improving the evaluation phases of notification system design.

Implication for UEM Researchers

This work has illustrated a new approach to UEM comparison. Instead of relying upon evaluators to produce disjoint problem reports which UEM researchers must then interpret, we have suggested

an alternative approach that strengthens consistency and eliminates some of the ambiguity inherent in UEM comparison studies.

This is achieved through a setup that requires the UEM researcher to provide a real problem set up front, to each of the evaluators, instead of trying to derive the set from the comparison test results. There are several advantages to this approach. Obviously, we can eliminate the ambiguity that arises from the inspector trying to interpret the problem reports from the system evaluators. Secondly, by using the same set of problems for each evaluator, we have tighter control of the experimental conditions. Lastly, by having the evaluators concentrate on the system and the targeted UEM instead of writing out problem descriptions, they can provide more robust analysis and give more reliable answers.

Furthermore, this particular setup allows for easy calculation of the desired metrics (thoroughness, validity, etc.). By using the structured presentation and data collection methods employed here, UEM researchers can quickly and easily calculate all of the metrics included in the Hartson et al. technique. A spreadsheet program or database application can easily perform the necessary calculations.

One other advantage to this setup is that it can be somewhat automated by connecting to a database of design knowledge. UEM researchers would be able to import design problems or claims from a database, as well as heuristics (if comparing heuristics). This would allow for rapid creation of multiple tests across different systems and UEMs. Efforts to support this are ongoing, as reported in [17].

This could be especially important for large companies that produce or focus on one type of system. Leveraging a new evaluation mechanism, like heuristics, could prove time consuming if a new evaluation has to be created for every system to be tested. Having an automated platform can facilitate multiple, rapid evaluation setup, execution, and data collection.

5.6 Summary

We have described an experiment to compare three sets of heuristics, representing different levels of generality/specificity, in their ability to evaluate three different LSIE systems. Information on the systems used, test setup, and data collection and analysis has been provided. This test was performed to illustrate the utility that system-class specific methods provide by showing how they are better suited to evaluation of interfaces from that class. In addition, this work has provided important validation of the creation method used in developing these new heuristics.

We have shown that a system-class specific set of heuristics provides better thoroughness, validity, and reliability than more generic sets (like Nielsen's). The implication being that without great effort to tailor these generic evaluation tools, they do not provide as effective usability data as a more specific tool.

While this experiment makes a compelling case for system-class specific UEMs, further validation of the actual heuristics is required to ensure faith in the creation method. Efforts to achieve this validation are described in the next chapter.

Chapter 6

Heuristic Application

This chapter reports on efforts to illustrate the utility and usefulness of the heuristic sets created through the development process reported in Chapter 4. While they were created from inspection of five example large screen information exhibits, and experimentally shown to be as good as or better than other heuristics, these heuristics need to be validated in the sense that the method is usable in evaluation efforts.

By using these heuristics in real system development efforts, we gain insight into the effectiveness and utility of this UEM. We are pushing the envelope on heuristic use by focusing on non-traditional users: novice HCI students and domain experts. These two groups often rely on analytic techniques, and using the new heuristics with these groups can illustrate the utility of the new set while providing support for the heuristic creation process. In addition, we seek expert opinion, providing feedback and commentary on the heuristics.

6.1 Introduction

Before we can expect practitioners to use the set of heuristics developed in Chapter 4, it is necessary to provide support for their existence. In other words, we need to show that these heuristics are indeed a "good" set and uncover a substantial portion of the usability problems associated with large screen information exhibits. Three approaches were taken to assess the utility of the heuristics for evaluating large screen information exhibits; one involves neophyte HCI students, one involves domain experts, and a final approach relies on international HCI expert opinion. Each of these efforts will be discussed in turn.

6.2 Novice HCI Students

An initial look at these heuristics was done by neophyte HCI students. These students come from an introductory undergraduate HCI class that I 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, but they had experience with usability engineering concepts from

the course content. The analytical evaluation stage occurred towards 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 described in chapter 4. The goals of the evaluation were to help the students with the design of their systems, and to gather feedback on the utility of the heuristic method for producing redesign guidance. 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.

6.2.1 Method

This test was conducted as part of course requirements for the Introduction to HCI class in which the students were enrolled. Sixteen students participated in this study in the form of 5 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 six week period, with summative evaluation occurring in the fifth week. None of the students were familiar with the systems used in the creation process reported in Chapter 4. Furthermore, they were not familiar with the new heuristics before the evaluation assignment. They were familiar with analytic evaluation and had performed a simple such evaluation in a class activity that used Nielsen's set (as found in [70]).

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.

6.2.2 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, but only 19 at most, with an average of 16 problems found per team. Figure 6.1 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.

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

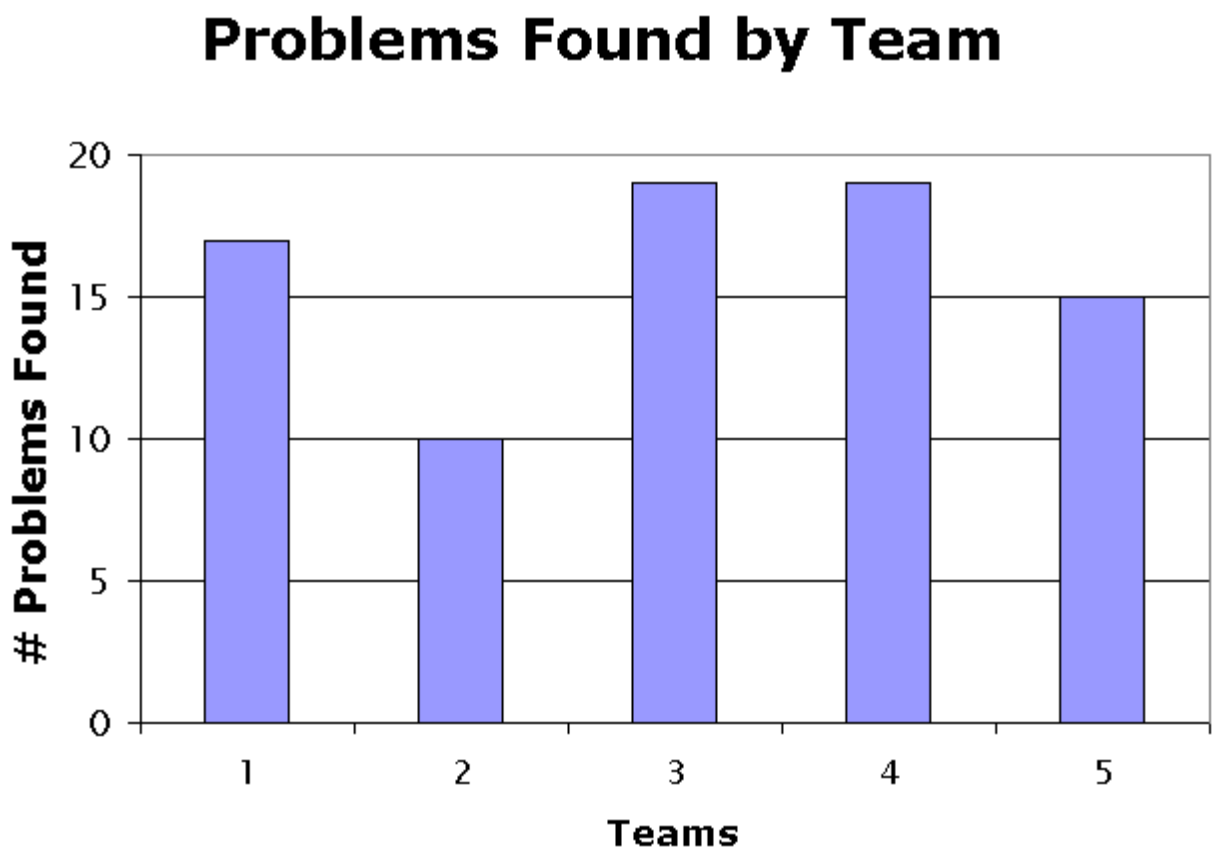

Figure 6.1: Total number of problems uncovered with the heuristics, shown by team.

The majority of the students felt the heuristics were "useful" and provided "much needed guidance for evaluation effort." Granted, this in itself would be expected, because as composed to no heuristics, having something to guide evaluation effort is indeed useful. 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. Figure 6.2 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 they would have produced better designs before doing any evaluation had they known about the issues contained in the heuristics.

6.2.3 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 quite good. In fact, each of the solutions given by

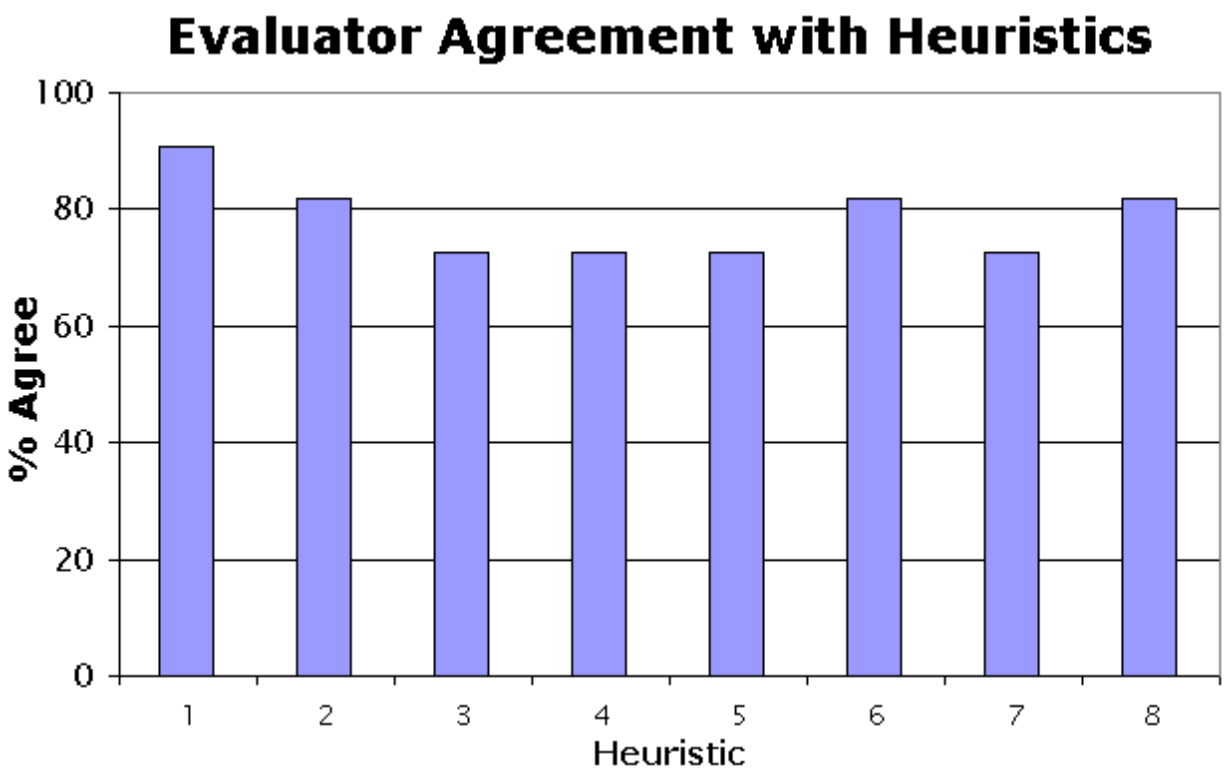

Figure 6.2: Percentage of students who agreed that the heuristic was applicable to large screen information exhibits.

the student groups consisted entirely of one screen, which typically employed some animation technique to show changes to information sources. Thus, 16 real problems identified in these systems allows for substantial improvements to the design.

We accept that the participants in this study were NOT expert HCI professionals, as is typically used in heuristic evaluation [70]. 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 indeed 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 indeed 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 of the 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—they can be used to guide design from the start of a project by identifying and illustrating potential trouble spots in the design of large screen information exhibits—thus they can truly be considered as heuristics. This apparent usefulness also strengthens the overall desirability of these heuristics for use in design and evaluation of large screen information exhibits.

	Total Problems	Overall IRC	interruption	reaction	comprehension
#	183	88	20	9	59
% of total	—	48	11	5	32

Table 6.1: Summary of problems found through student application of heuristics.

6.2.4 Post-Analysis of Problems

Because this summative-style evaluation occurred towards the end of the project development cycle, 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 knowledge of the IRC framework. All they knew about LSIEs was that they involved software running on large display surfaces. Hence, it is 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 the problems reported by the students related to some design artifact within the particular system that was evaluated. 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:

> The current temperature does not stand out well against the blue background.

Clearly this problem describes a specific piece of information (the current 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, he/ she would experience decreased understanding of that information, or a lower comprehension. Alternatively (or in addition) a user may experience an increase in interruption because it takes more time to look at the temperature and decode what it says.

Most of the problems reported by the students relate 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 IRC parameters. This is about 48% of the problems. Of the problems that correspond to critical parameters (88) there were 59 related to comprehension, 20 related to interruption, and nine related to reaction. This breakdown reflects the emphasis on the comprehension based tasks for which the systems are created to support. Table 6.1 provides a summary of this analysis.

6.2.5 Evaluator Ability

It is prudent to consider the ability of these students in regards to completing heuristic evaluation. While these students had little experience with evaluation and could be considered novices, they did understand the purpose of usability evaluation and understood the goals of the system that they evaluated. Given this understanding of their ability level, it is interesting to conceptually compare the results of a "good" evaluator and a "poor" evaluator from this pool.

As expected, the majority of the problems reported by the students dealt with specific design decisions and related impacts to the user goals of the display. There were only three students who provided robust problem reports that could be useful during redesign. The nature of the problem descriptions for these "good" evaluators consistently revolved around specific interface artifacts and the consequences of those decisions. One outstanding example even described the problems in terms of claims, listing both upsides and downsides of the design element as psychological tradeoffs for the user.

Nine of the students provided what we call "average" reports, consisting of a short description of a problem they thought would occur. The problem with these reports was that there was little re-design guidance provided in the wording of the problem to aid designers when fixing the problem.

Four of the students gave "poor" problem reports, merely consisting of simple statements describing how the system followed the heuristics (or not). These reports described very few problems with the target systems, and held no re-design value. In most cases, the so called problems that these students reported were in actuality a description of the system, not an analysis of the system.

Speculation suggests that the students in the "poor" category did not spend adequate time on the assignment, perhaps only attempting the exercise the night before they were due in the class. In contrast, the "good" examples likely involved several hours playing with the interface in question, assessing it in terms of typical tasks, and writing useful problem descriptions. We must also concede that the heuristics themselves could have posed a problem for the students in the "poor" category. Perhaps the students did not understand how to apply the heuristics due to the wording. Perhaps these students did not fully understand the idea encapsulated in each heuristic.

However, our evidence suggests that these HCI students were able to effectively apply the heuristics in evaluation of LSIE systems. Coupled with the positive feedback from these students, we feel confident that the heuristics provide structure and guidance for analytic evaluation efforts. However, this example did not illustrate how the problems could be used in system re-design. The following section describes a separate effort in which the heuristics were used and the problems fed into a re-design.

6.3 Education Domain Experts

A separate application of these heuristics consisted of having domain experts use the heuristics in an analytical evaluation of the GAWK system. Specifically, the teachers involved in the Classroom Bridge effort [31] used these heuristics to evaluate an updated version of the GAWK software.

One could ask why we wanted to use non-HCI people in a heuristic evaluation, especially since usability experts are required for effective results. 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 [70]. 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.

One other question about this effort involves the use of the GAWK system. Recall that the GAWK was one of the five systems in the heuristic creation process. If we use the heuristics from that process to evaluate the GAWK system, what would we really be finding? There are two

answers to this. One is that the GAWK system underwent significant design changes from the version that was reported in Chapter 3 and used in the creation process in Chapter 4. In reality, the new GAWK looks different from the original version and thus learning whether the set of heuristics can uncover the original problems would validate the set to a certain degree. Second, because the system performs the same functionality and only underwent cosmetic changes, any problems found in this effort should match up with the issues we identified in the earlier study (Chapter 4).

6.3.1 Method

Since these evaluators are not expert usability people, 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 [52] 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 were novice usability people, and thus had no clue about usability problems or how to report them. 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 they had executing those tasks, filling out problem reports for each problem encountered.

This method is slightly different from the suggested implementation of a heuristic evaluation as reported in [70] but we felt it was necessary given the novice experience level of our evaluators. In traditional heuristic evaluation, the evaluator does not typically use the system or try to complete tasks with it. Instead, they attempt to derive potential problems from a purely analytic analysis of the system and its intended usage. Here we have our evaluators actually using the system in specific, scripted scenarios. We felt that the structure provided through this approach would facilitate the problem identification and help with the reporting effort.

There were two forms of data collected in this study: the problem reports and interview feedback on the heuristics. Each evaluator provided their own problem reports, detailing the problem the 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.

6.3.2 Results

These non-HCI professionals, who had never heard of heuristic evaluation before, were able to take these heuristics, and identify several usability problems with the display. These two evaluators found a total of 23 problems with the system. While this number may seem low, considering their lack of experience with analytical evaluation techniques, it is a good number.

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 indeed 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 of the evaluators 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."

6.3.3 Discussion

Simple 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 Chapter 4. Furthermore, if we examine empirical findings on heuristic evaluation [69], 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 sheer 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.

6.3.4 Post-Analysis

Deeper 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.

Interruption	*Reaction*	*Comprehension*
6	7	17

Table 6.2: Number of problems identified by teachers that relate to critical parameters, shown by breakdown for each parameter. Some problems were related to multiple parameters, hence the total is greater than the number of problems found in the evaluation (23).

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. Table 6.2 provides the numbers for each parameter. We believe that this high correlation between the problems and the parameters is a direct result of the heuristic creation process that is based on the critical parameters. 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.

6.3.5 GAWK Re-Design

Because the programmers involved with the ClassroomBridge 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.

Comparing this new design to the earlier instance of the GAWK display, we see some important changes. Overall the structure is 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 space. 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 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. More discussion of this notion can be found in Section 8.3.2.

6.4 HCI Expert Opinions

Another effort to validate this set of heuristics was through international usability experts. Specifically, a terse description of the creation work and the heuristic list was submitted and accepted at the Ninth IFIP TC13 Conference on Human-Computer Interaction in Zurich, Switzerland. This paper was peer reviewed by an international community. Hence, acceptance suggests these heuristics are indeed useful for interface design and evaluation.

In addition to acceptance within the international community, interviews with several usability experts were also conducted. These interviews were informal in nature and were conducted at the above mentioned conference. The interviewees were attendees at the conference, and were in some way connected to large screen display design and/or evaluation; either through dedicated research or through usage contexts.

6.4.1 Method

The interview structure consisted of a few questions about the usage of large screen display technologies within the interviewee's current work setting, usually an industrial or academic research setting (like a lab). The majority of the interviewees were interested in designing and developing large screen display interfaces.

In all, ten Human-Computer Interaction experts were interviewed about large screen display technologies and their usage. Notes were taken on the comments given and were subsequently used to determine opinions on and about large screen information exhibits. The interviews were informal in nature and mostly revolved around their individual opinions on the heuristics created in this development effort. Other comments focused on LSIEs and the shifting computing paradigm (from desktop to more ubiquitous devices).

6.4.2 Results

There was a broad range in opinions about the heuristics from the interviews. Five of the 10 experts stated they "liked" them and would use them in their development efforts. The others were more skeptical and wanted to see evidence of the heuristics in use, or in comparison to other sets of heuristics. None of the experts felt that the heuristics were unnecessary, and all agreed that this was an interesting and useful approach to a difficult usability evaluation problem.

Four of the ten experts gave specific feedback on the individual heuristics. Of these four experts, they generally felt that the heuristics were simple enough and covered a significant portion of the potential trouble areas in LSIE systems. However, two of these four experts pointed out that three or four of the heuristics had more of a "guideline" feel to them, rather than a "heuristic" one. When prompted to explain this separation, they stated that the wordings of the heuristics sounded more like commandments rather than general rules. For example, the heuristic "Avoid the use of audio" struck them as too strict for wide applicability.

6.4.3 Discussion

Overall, the experts who reviewed and provided feedback on the heuristics had a positive outlook towards them. The overwhelming point from this effort was that the heuristics were "untried" and needed to be used in real evaluations and compared to other types of heuristics; both to show that they can uncover usability problems, and to show that they are better suited to the LSIE system class than other heuristics.

At the time of the interviews, none of the interviewees were aware of the validation efforts involving use of the heuristics (as described in Sections 6.2 and 6.3). They were informed of intent to compare this set to other types of heuristics and applauded that goal. All were curious to see the results of such a comparison study.

6.5 Overall Discussion

These three efforts provided feedback on the usability and effectiveness of the heuristics. We have evidence that the heuristics can be used in real evaluations, both from the neophyte HCI students and the domain experts. These heuristics uncover real usability problems and provide design feedback in formative evaluation efforts. In addition, expert usability people had a positive feel towards the heuristics, suggesting they are a step in the right direction.

The successes in these efforts indicate that the heuristics hold great potential for supporting formative usability evaluation. Coupled with the experimental findings from Chapter 5, we now have confidence in the heuristic creation method based on critical parameters.

6.6 Summary

In the process of creating a new set of heuristics for the large screen information exhibit system class, we had to show that the heuristics we came up with can be used in real evaluations and actually produce results. By having neophyte HCI students employ these heuristics in evaluations of new LSIE systems, we gain the confidence that the heuristics are effective for uncovering usability

problems. By having domain experts use these heuristics, we gain the confidence that the heuristics are usable; they are easy to understand and apply in heuristic evaluation. Usability experts also provided positive feedback about this set and felt the set was promising and should be compared to other types to illustrate the improvements from using this UEM method over other methods.

This evidence lends strength that these heuristics are suited for large screen information exhibits by supporting usability evaluation. This also gives credence to the creation method we reported in Chapter 4–that process produces usable and reliable heuristics.

Chapter 7 summarizes the work and provides some discussion of the implications of this work. Chapter 8 details the contributions of this work and possible areas of future work.

Chapter 7

Discussion

To fully appreciate the impacts of our work and the subsequent application of the results, we need to discuss potential areas where these efforts have important ramifications. After discussing important facets of the tool development work described in previous chapters based on critical parameters, we discuss interface evaluation, testing usability evaluation methods, and UEM generality vs. specificity. These topics directly relate to the contributions of this work (Section 8.2, and have important impacts for future work (Section 8.3).

7.1 Supporting UEM Creation Through Critical Parameters

As developers build knowledge on different system classes, focused on critical parameters, the techniques described in this work can allow systematic creation of heuristics for formative evaluation. Because system class specific evaluation methods hold great potential for guiding evaluation as well as allowing benchmark development and system comparison, the research community should focus tool development effort on system class specific methods.

The method described in this work (see Chapter 4) provides the necessary steps for creating and testing a new evaluation tool tailored to the critical parameters for a system class. To further aid those who may wish to implement this technique, Appendix H provides a guide, in which the process is extracted to high level steps and illustrated with an example.

By relying on the method described in this work, UEM developers can focus their effort on specific steps that lead to the creation of an analytic tool for formative evaluation. They no longer have to spend considerable time aimlessly evaluating and analyzing systems with no clear structure or process, in hopes that massive, disjoint effort will unveil "gems" of design knowledge that could lead to an evaluation tool. Instead, effort is structured and guided with specific goals and tasks, ensuring that tool creation effort results in a usable evaluation tool.

Our work has provided valuable insight into the development and testing of a new set of heuristics, tailored to the unique user goals associated with the large screen information exhibit system class. Through two empirical studies (see Chapters 3 and 6), we now have evidence to support a new UEM creation technique, as well as evidence to suggest that the resulting tools are indeed useful.

Traditionally, usability specialists and developers spend significant time and effort creating usability evaluation tools for a single system. Our efforts have shown that more generic methods

hold promise for designer because they reduce the amount of time required in formative evaluation. Focusing evaluation tool design on the user goals of a system class produces desirable results as well as allowing for re-use of the evaluation tools. Initial investment of tool development effort can yield better usability results in the long term.

A major problem with this approach arises from lack of clear methods for creating a new evaluation tool. By focusing on critical parameters, evaluation tool creation is proceduralized and simplified. Based on the notion of critical parameters, focused analysis and knowledge extraction produces detailed design guidance. This design guidance provides a foundation for creating high-level, generalized design concepts that can guide formative evaluation efforts in the form of heuristics.

An important aspect of the critical parameter approach to heuristic development is that the heuristics will be targeted to the most important user concerns with the system. In our case the heuristics indicate how to support the appropriate levels of interruption, reaction, and comprehension. Problems identified by these heuristics are readily approached because the heuristics indicate possible ways to improve the design, or at least suggest alternatives that may alleviate the symptoms of the problem.

The approach described in Chapter 4 details how one can develop heuristics based on critical parameters. Applying this approach to other system classes will provide the research community with valuable evaluation tools and reusable design knowledge. However, this process includes the assumption that the critical parameters are already defined for the target system class. Here the critical parameters of interruption, reaction, and comprehension were previously identified and accepted within the community. Other system types, like ubiquitous interfaces, may not have predefined critical parameters; hence, the method described in this work may not be immediately applicable.

However, there could be other ways to define a system class. Perhaps focusing on the types of tasks that the systems support could lead to a clearly defined system class. In this case the technique described herein would be applicable. Other system class definitions could result from investigating user goals. Again, once a set of design parameters are known, our method can be used to guide heuristic creation for that system class.

Whether one focuses on user tasks, user goals, or well-defined critical parameters, there is still the question of how to determine when a set of each actually defines a system class. This question is beyond the scope of this work but this work has illustrated the need for a structured process for identifying reliable critical parameters.

Another aspect of the creation method reported in this work involves the effort required to actually derive the heuristics through the process. Identifying systems, creating scenarios, extracting claims, and creating the problem tree took six weeks of dedicated effort. Extracting heuristics required an additional four weeks. 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 $\frac{1}{2}$ to $\frac{3}{4}$ of this time (say 3 to 5 weeks). While significant, this effort is worthwhile, as illustrated and discussed in previous sections. The question arises then about how much time other methods would require. If we consider an alternative set of heuristics, like Nielsen's, one can ask how much effort went into the creation of that set. It has not been clearly documented as to how and where the original heuristics actually came from, but it can be assumed that it took longer than 6 weeks to come up with that set of heuristics (and likely involved years of thought and experience). Because researchers need and want more specific heuristics, our proposed creation

method provides the necessary tools to facilitate this UEM creation effort, even if it may require six weeks of effort.

7.2 Supporting Interface Evaluation

This work has illustrated a gaping hole within the development space for interfaces. That hole is a lack of evaluation methods and tools for achieving effective formative evaluation results. We started to fill this hole by focusing in on LSIE systems but there are numerous other types of interfaces within the notification systems design space, and within other interface types, that could benefit from targeted evaluation tools.

This work supports this endeavor in several ways. First, evaluating notification systems, especially formative evaluation, often necessitates reliance upon analytic techniques. Supporting the creation of new analytic tools enhances notification systems researchers' abilities to produce effective designs. The method we have provided can ensure rapid development of re-usable heuristics for many types of systems.

Secondly, by promoting the use of critical parameters and the associated terminology, the research community is strengthened and solidified in advancement efforts. A community of researchers, discussing and talking about the same types of ideas, can more readily breach the research gaps within emerging fields.

Finally, this new set of heuristics provides immediately available tools for LSIE system designers and developers. Existing designs can be readily evaluated and emerging systems can benefit from early evaluation and design guidance.

7.3 Comparing Evaluation Methods

For researchers interested in comparing evaluation methods, our work provides a unique setup that supports these types of endeavors. Usability evaluation is one of the most important and costly steps in developing human-computer interfaces. The goal is to identify and fix problems in the interface, so as to improve the user experience with the interface.

What this means is that researchers are tasked with developing effective evaluation tools for many different system classes. A key part of this development work involves testing the new methods and comparing them to existing alternatives. Traditionally UEM comparison has been difficult and wrought with problems that lead to debate over validity and utility of previous comparison studies [32]. A large part of the problem with these comparison studies involves validity in problem sets and results. Hartson et al. suggest a set of metrics to use in UEM comparison studies that are designed to help with accurately comparing different methods [40]. However, calculating these measures relies upon knowledge of the real problem sets for the target system. This can be problematic for traditional comparison tests because it is not clear what constitutes the real problem set [40].

Our work addresses this problem in a novel way. Instead of having evaluators uncover usability problems in a traditional evaluation; we provide a list of problems for the target system and ask the evaluators to rate the applicability of specific heuristics to that problem. In other words, we ask the usability professionals to assess the heuristics in terms of how much the heuristic would

help in identifying the issue in a traditional evaluation. This relies on the evaluators' experience with usability problems and their ability to reason about the heuristics in an abstract manner. The strength of this approach comes from the fact that we have a specific set of problems to serve as the real problem set and we can easily calculate the Hartson et al. measures from this setup.

By providing a set of problems to the evaluators, we can more accurately determine the applicability of a set of heuristics to that problem set. This allows us to quickly calculate several measures of the method and compare different methods on the same basis. Other comparison studies usually must deal with validity issues that arise from evaluator differences, investigation of lengthy, wordy problem reports, and then mapping multiple descriptions of problems to an accepted set of problems.

These comparison studies are often also plagued with having questionable or weak "real" problem sets. For example, a common technique is to use the union of problems found by all the methods in a comparison study as the real problem set. One problem with this approach arises from the fact that this set of problems may not be the ones that real users would experience during typical system use. In our approach, we use actual problems encountered by the users of the systems, as found through system inspection and feedback with developers, or through direct user studies.

Our implementation of this new comparison technique suggests a better approach to UEM assessment. Instead of relying on highly variable problem sets from traditional evaluation approaches, we have a common base set to use in the calculation of comparison metrics. We reduce the variability in the calculations, ensuring that the comparison is fair and balanced.

Furthermore, this approach can be somewhat automated. By relying upon existing design knowledge, one can create a new testing setup by importing usability problems and heuristic sets to dynamically create new tests, either for evaluating the problems or for comparison tests. In fact, this particular effort is underway as part of the LINK-UP system for evaluating notification systems [17]. The testing platform used in this work can be automated to retrieve specific claims from a database, which can then be used in analytic evaluations.

7.4 Exploring Generality vs Specificity

Exploration of the generality/specificity question within usability evaluation method applicability occurred through two experiments. Covering the spectrum of specificity from system specific, through system class, then interface type, and finally to generic interfaces, allowed for deeper understanding of how the specificity level impacts usability problem coverage. Usability evaluation methods that are targeted for a specific system class provide the most promise for continued development and study. They seem to apply to the systems and produce more reliable results than more generic methods. In addition, evaluation tools targeted to system classes also provide enough generality that they can be applied to multiple systems within the class as well as help in identifying benchmark performance. What does this mean for other usability experts? How can they leverage this finding in their development and testing efforts?

By providing the HCI community with analytic evaluation tools, we are supporting the future development of notification systems. Because there are better alternatives for effective notification system analytic evaluation, formative evaluation design phases can be completed with higher efficiency and better results. Furthermore, the creation method reported in this work allows UEM researchers to create these system-class specific UEMs through a structured process, eliminating

ambiguity and uncertainty.

The tools need to be targeted to the specific user goals associated with the system class, yet need to be general enough to allow for reuse across multiple systems within the class. Our work has suggested that the best place to focus tool development effort is at the system-class level, which is defined by specific levels of critical parameters. Using our creation process, researchers can focus on the critical parameters that define a design space, thereby building new evaluation tools with focus and ease. By doing so, new design knowledge can be captured and reused in future efforts.

A long term effect of this type of work includes the strengthening of the notification systems research community; both through promotion of reusable tools and design knowledge, as well as through continued evaluation and system refinement. This effect can occur due to the reduction in time required for developing effective, reusable evaluation tools.

A possible tradeoff of using more specific tools rather than highly generic tools (like Nielsen's heuristics) could be loss of creativity. In other words, if a UEM deals with specific terms and concepts indigenous to the target system, it is possible that the evaluator would be less likely to rely upon his/her mind to assess the interface in ways that may not be obvious. By using focused terminology and concepts, imagination could easily be constrained.

While we concede that this is possible, we must point out that the experiment described in Chapter 5 does not support this suggestion. Here we almost totally relied upon the evaluators' ability to think about the claim and assess whether or not a heuristic had some applicability to that claim. If the above assertion were true, we would have expected Nielsen's heuristics to have the highest applicability of the three methods. This expectation arises from the structure of the test; evaluators would have been able to think at a high level whether the heuristics had any applicability to the claim. Without restrictions on their thinking and consideration, one might expect that the more generic heuristics would be rated with higher applicability. This was not the case in our experiment. The evaluators indicated that Somervell's heuristics had the highest applicability.

7.5 Lessons Learned Through Use

Both examples of using the heuristics in system evaluation provided valuable insight into how the heuristics support these types of efforts. There are several areas related to the development and use of these heuristics from which important ramifications arise: reporting problems to programmers, mapping problems to critical parameters, specificity in heuristics, work/benefit tradeoffs, and differences between critical parameters and usability metrics.

7.5.1 Reporting Problems to Developers

One of the most important aspects of usability evaluation is reporting the results to the programmers/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 programmers and/or 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 [80, 21]. Our heuristics are a step in the right direction, because they are closely related to the critical parameters of the LSIE system class. Indeed, this connection to the critical parameters can be strengthened through further research (see Section 8.3.2).

7.5.2 Mapping Problems to Critical Parameters

Though it is desirable for programmers to separate the problem reports from critical parameter terminology, system developers and researchers often need to know how well a system is performing its intended function, for comparing systems or benchmarking; thus they need to know which parameters are addressed by a system. This seems to suggest a conflict — on the one hand programmers want problem descriptions to revolve around interface artifacts and user impacts, on the other hand researchers want to know how well certain critical parameters are addressed by a system.

This conflict may not actually exist. Inspection of the problem reports from the domain experts reveals that most of the problems are inherently tied to the critical parameters, even if they do not explicitly use the terms interruption, reaction, and comprehension. In addition, these problem reports typically suggested the interface artifact that, when changed, could alleviate the problem. We seem to be getting the best of both worlds. Granted, a small amount of analysis is required to ascertain to which parameters a particular problem pertains, but this effort is minimal in most cases.

How are we achieving this robust problem reporting? It seems that the heuristics that came from the critical parameter based creation process allow problem reporting that bridges the aforementioned conflict. Because the heuristics are tightly coupled to the critical parameters (through the claims analysis process), problem reports seem to revolve around the underlying user goals associated with each parameter. Furthermore, the heuristics suggest specific interface artifacts (like use of animation) that programmers could immediately identify for change. Hence, the heuristics provide the robust analysis that supports both the programmer and the researcher.

7.5.3 Specificity in Heuristics

It is worthwhile to discuss the heuristics at a conceptual level, especially with respect to how specific they are. The students from our second application effort and the expert reviewers suggested that some of the heuristics were “too strict” and needed to be more generic in nature. This raises the question of what constitutes a heuristic. Are heuristics just “vague checklists” as argued by Sauro

[80]? How vague is vague? What are guidelines then? Are guidelines not checklists, perhaps only slightly less vague than heuristics? Understanding the difference between what is considered a heuristic and what is considered a guideline is non-trivial. Individual opinions will vary, but in general a guideline is thought to be more "specific" than a heuristic. In other words, a guideline suggests how to design while a heuristic raises questions about a design. However, clearly defining the separation point is the difficult task, and because so many different opinions surround this notion, no clear answer is available.

Still, we are faced with the question of what "specific" means for a heuristic. It is obvious that the new set of heuristics created in this work are more specific than both Nielsen's and Berry's sets, with respect to the LSIE system class. However, are the new heuristics still considered heuristics, or are they just guidelines? Does this question even matter? Our comparison experiment has shown that the more specific heuristics had better thoroughness, validity, effectiveness, and reliability. Would that not suggest that even if they are "guidelines", they are better suited for analytic evaluation? In addition, our application examples both illustrated that these heuristics find usability problems in typical analytic evaluations. This provides strong evidence that the creation process produces sound evaluation tools. What this implies is that we need to further investigate analytic evaluation through differing levels of tool specificity.

7.5.4 Development 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 six 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 hours 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 hour 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 $\frac{1}{2}$ to $\frac{3}{4}$ of this time (say 3 to 5 weeks).

All this work amounts to significant investment. So why is it worthwhile? There are three major reasons. First, the targeted heuristics had higher thoroughness and validity scores in the comparison test. 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.

Secondly, because these heuristics had the highest reliability scores, 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.

Thirdly, 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 insures 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. We expand on this in the next section.

7.5.5 Critical Parameters vs. Usability Metrics

It is interesting and worthwhile to discuss how critical parameters and usability metrics are related. Because critical parameters are universally accepted indicators of whether at system serves its intended purpose, they are often confused with typical usability metrics. There is a distinction however. Usability metrics are typically described *after* design and before testing, whereas critical parameters are described "at the outset of design" [68]. Also, usability metrics focus solely on whether or not a system will be "usable" by the consumer. While this goal is desirable, it does not entirely focus evaluation on whether a system actually performs its function correctly, which critical parameters provide [68]. Usability metrics are not "utility metrics". For example, consider the GAWK system as described earlier. Its purpose is to allow teachers and students to understand information about lengthy class projects. A typical usability test may show that the interface is easy to use and easy to learn. What these usability metrics do not show is that the program is supporting the long term understanding of the information.

This example shows the difference in critical parameters and usability metrics. A critical parameter is an attribute that captures an important functional aspect of a system [68]. In contrast, a usability metric is how a person could measure a critical parameter through testing. Take the example of our critical parameters for notification systems. Each of interruption, reaction, and comprehension can be measured through separate metrics. Interruption is measured by primary task degradation. Reaction is measured in response time or hit rate. Comprehension can be measured by questions about the information source, with percent correct being the metric.

In addition, describing interfaces in terms of concrete parameters supports testing and long term benchmarking. Because critical parameters are closely tied to the typical user tasks associated with a system, benchmark performances can be determined and described through the parameters. If a newly created system performs poorly in one or more of the parameters, then the designer knows where to focus re-design effort. For example, say a newly created LSIE system has weak performance on a benchmark comprehension test, then the designer can focus re-design effort on helping the user understand the information (maybe a different layout, maybe textual formatting, maybe including a legend). The same utility is not found in standard usability metrics. If an interface performs poorly on "ease of use", what does that tell the designer? There would be too many possible areas where a re-design could occur. The critical parameters suggest to designers where improvements can be made, focusing design effort on fixes that result in better systems.

7.6 Discussion Summary

We have illustrated the utility of our new heuristic creation process through experimental comparison and application. Researchers now have the techniques to help close the evaluation gap present in all human interfaces. By focusing on critical parameters, structured UEM creation can proceed. Coupled with our process, usable heuristics result, pushing notification systems evaluation forward. The following chapter provides a summary of the work, describes some of the major contributions, and then concludes with potential extensions and future work.

Chapter 8

Conclusion

Here we provide a summary of the work completed, as well as descriptions of the contributions of the work and discussion of potential future work to extend and build off of the efforts described in previous chapters.

8.1 Summary of the Work

Supporting the creation of heuristics for system classes is accomplished through a process that involves system inspection, claims analysis, classification, categorization, and design knowledge extraction. The impetus for this work comes from the desire and need for tailored UEMs, providing targeted evaluation and reliable results.

Early work (Chapter 3) involved the comparison of generic to specific UEMs in evaluating LSIE systems. That work provided the background and motivation for pursuing a structured heuristic creation process, based on critical parameters. Results of the early work suggested that system-class specific methods are probably best, which coincided with findings from earlier research. Furthermore, this background work illustrated the utility in formative analytic evaluation for notification systems.

Chapter 4 describes the heuristic creation process. The method relies upon critical parameters, which define a design space, for effectively deriving usable heuristics for a target system class. Claims analysis and Scenario Based Design are tools used to facilitate the process. After identifying 333 claims, 22 design issues were extracted, then eight heuristics were synthesized. The appropriate levels of the critical parameters–interruption, reaction, and comprehension–allowed us to derive the final heuristics.

After creating a set of heuristics tailored to the LSIE system class, we performed an experiment to compare the new heuristics to existing, more generic alternatives (Chapter 5). This comparison study illustrated the effectiveness of system-class specific heuristics and validated the heuristic creation method. In addition, a new UEM comparison technique was developed and implemented to facilitate the calculation of comparison metrics. This new testing platform allows UEM researchers to compare multiple methods, quickly and easily; reducing ambiguity and increasing validity in the process.

In addition to the comparison study, we also illustrated the effectiveness of the new heuristics through two real-world applications of the method (Chapter 6. In one instance we had undergrad-

uate HCI students perform analytic evaluations on newly created LSIE systems by incorporating the new heuristics. Findings from this application suggest that the new heuristics were easy to use, reliable, and produced important design problems that were fixed in subsequent implementations of the systems. The second application involved domain-expert evaluators using the heuristics to evaluate an LSIE system. Again we found that the heuristics were highly applicable to the system and facilitated these domain-experts in uncovering usability problems with the interface. These validation efforts illustrate the utility of the new heuristics, as well as provide support for the creation method.

8.2 Contributions

This work impacts several important research areas within the Human Computer Interaction branch of the Computer Sciences, including: usability evaluation method creation and testing, usability evaluation method applicability, notification systems evaluation, notification systems design and development, large screen information exhibit design and use, and knowledge re-use. Specifically, the contributions of this work include:

- **Critical parameter based creation of system class heuristics**
- **Heuristics tailored to the LSIE system class**
- **LSIE system design guidance**
- **UEM comparison tool**
- **Deeper understanding of the generality vs. specificity tradeoff**

8.2.1 Critical Parameter Based Creation of System Class Heuristics

The main contribution of this work comes from the creation method leading to the set of heuristics for large screen information exhibits. This method is based on solid methods of analysis of existing systems (scenario based design and claims analysis) and leverages the notification systems framework (critical parameters) in the classification and categorization of claims into usable heuristics. This method has a specific structure and can be repeated for other system classes. One example is the notification systems design space (as described in [62]). In that model, different levels of the three critical parameters–interruption, reaction, and comprehension–define subsets of the design space for notification systems. Large screen information exhibits are one such subset of the entire notification system space, but other types of systems (like alerts, secondary displays, ambient displays, etc.) can benefit from this creation process. We simply focus the analysis on the particular user goals for the target system class. For example, instead of focusing on self-defined interruption and high comprehension (as for LSIEs), we could look for low interruption and low comprehension and focus on *indicators* [62]. We could then identify three to five example systems from this class, perform claims analysis on them to identify current design tradeoffs, then use the method described herein to systematically extract high level design issues and potential heuristics for the indicator system class.

This work illustrates the strength and utility that critical parameters hold for guiding usability evaluation method creation. As alluded to in Section 7.1, determining the critical parameters for a system type may not be obvious. Speculation on methods for guiding this process are discussed in Section 8.3.3.

8.2.2 Heuristics Tailored to the LSIE System Class

Another important contribution of this work is the set of heuristics. LSIE designers and evaluators now have an effective formative evaluation tool. Evaluation resources can now be devoted to running analytic tests and analyzing results, as opposed to developing tools for every system to be tested.

In addition, other notification systems researchers can potentially leverage these heuristics in evaluation of related system classes. For example, some of these heuristics would facilitate the evaluation of secondary displays, as well as ambient displays. By having more tools in their usability toolkit, evaluators can more readily find important usability problems in their designs.

The new set of heuristics developed in this effort provides enough detail to uncover important usability issues with a target system, yet it is generic enough to be applicable across many different LSIE systems. Armed with this new tool, system developers and usability engineers will be able to include formative evaluation of their systems *earlier* in the process. Why earlier? Because they will not have to spend time creating a new evaluation tool, tailored to the system they are studying. Instead they can focus time and effort on performing the evaluations and analyzing feedback. Furthermore, these heuristics can be used early in design (before testing) to suggest possible design elements or to provide evidence for omitting certain technologies.

The real strength of this contribution becomes apparent when one considers alternatives to evaluating these types of systems. Often one would have to rely on existing tools and try to manipulate or change them to suit the current need. Now, evaluators and designers can use the method developed in this work to derive heuristics targeted for their particular needs.

8.2.3 LSIE System Design Guidance

A third contribution comes from the extensive analysis we performed in the creation process used to develop the new set of heuristics. We have on average 50 claims for the five systems used in the creation method that can be used by designers as evidence of good and bad design choices. These claims can serve as idea material for generating designs or as support for specific design decisions. In addition to the 253 total claims, we have identified 22 high level issues that capture the underlying design problems associated with the claims. These issues are not exactly worded to be used as heuristics, but they certainly provide design advice for large screen information exhibits.

In addition, these claims can feed into ongoing efforts within the local research community on understanding design re-use. The LINK-UP system [17] attempts to incorporate design knowledge gleaned from various system development efforts into the design of new systems. Part of this system requires analytic evaluation techniques within a system of exploring and testing various claims for use in new design efforts [17]. The analytic evaluation provides users a method for creating a set of claims that he/she is interested in for re-using in his/her development project but needs to get feedback on those claims [17]. The user can select a subset of the claims in a database or library and then have experts evaluate those claims according to some analytic method

(like heuristics) in an automated interface. The claims produced through our inspection of the five systems can feed directly into the growing claims database used in the LINK-UP system.

8.2.4 UEM Comparison Tool

Another important contribution comes from the testing methodology we used in the comparison study (see Chapter 5). For this test, we created a new way to calculate the UEM comparison metrics required for comparing evaluation methods. Instead of relying on problem sets from the evaluators (as recommended in [40]), we identify potential problem sets *before* the evaluation and have the test participants assess the applicability of the UEM to the problem. This new approach to UEM comparison eliminates variability in problem set identification, and provides control over metric calculation.

8.2.5 Generic vs. Specific UEM Tradeoffs

Finally, we have probed the issue of whether to use generic or specific heuristic evaluation methods for large screen information exhibits. We reached the conclusion that system level specific evaluation tools hold the most promise for providing effective evaluation of systems within a well defined class of systems. This was accomplished through an extensive comparison of three LSIE systems using three different types of heuristics, each representing a different level on the generic/specific scale; coupled with a study comparing system specific surveys to system class surveys. These two studies provide new insight into the generality/specificity tradeoff in usability evaluation method applicability.

The findings impact overall UEM research endeavors by allowing other researchers to refine their focus in creating new evaluation tools; increasing the potential effectiveness of their methods while reducing the amount of time spent exploring alternatives. This is achieved through applying the heuristic creation method we used in this work. Our specific technique could be modified to create questionnaires or surveys for particular system classes, but one needs an understanding of the key task goals for the system class for easy application.

8.2.6 Contribution Summary

These contributions are important to the emerging field of Notification Systems by aiding future design and evaluation of these displays and for facilitating comparison among systems that deal with similar problems. Systems that share design models, even those that are not designed for large screens, could benefit from the design guidelines and heuristics. Each area of the notification systems design space could be explored in this manner, identifying specific user goals and how systems support those goals, leading to the development of consistent and re-usable design recommendations for all notification systems.

The more general field of Human Computer Interaction will benefit from the sound methodology employed in investigating this area. Future researchers can take this approach and evaluate other types of methods for different classes of systems. Sound methodology and empirical practices illustrate strong science upon which others can build and refine, advancing the field.

8.3 Future Work

Even though this work has provided methods and techniques for improving notification system design and evaluation, there are several areas that can be identified where future work will yield important contributions, both within the notification systems design community and in the larger Human-Computer Interaction community.

8.3.1 Extend Method to Other System Classes

One obvious extension would be to use this method on each of the system classes defined by the IRC framework [62]. By applying the heuristic creation process described in this work to the other areas, two major advances result:

1. Further validation and refinement of the creation process and
2. Extensive coverage of the notification system design space.

There are basically seven areas of the notification design space that have not received extensive study, like that of this thesis work. Doing that work will flesh out the body of design knowledge surrounding notification systems and support future system development efforts. This knowledge can and should be stored in the emerging LINK-UP system [17] so that the effort to create evaluation tools is not lost to the design community.

Completing the system classes in the notification design space also provides a pre-existing test bed for refining the creation process. This is possible because the design space has been laid out and clearly defined system classes exist. Each class could benefit from targeted evaluation tools, hence application of our creation method seems the logical next step.

As researchers develop evaluation tools for each of these system classes, the process can become streamlined and even benefit from reuse of artifacts from the creation process itself. For example, scenarios created in the early phases of system inspection could be recycled and re-tooled to apply to different systems from different system classes. This reuse can even extend beyond the boundaries of notification systems and reach into the ubiquitous computing, CSCW (computer supported collaborative work), distance communication, and educational realms.

8.3.2 Automate Comparison Platform

Another logical extension of the work would be in automating the UEM comparison tool. Instead of relying upon pen and paper for data collection, the tool structure could be automated as simple web pages or as a survey through online survey tools[1]. A more robust implementation could directly tie in with design knowledge databases, allowing the UEM researcher to define problem sets and heuristics for specific tests, with quick and efficient data collection and analysis. Furthermore, existing evaluations can be saved and reused multiple times, with feedback instantly available to the entire design community involved with the knowledge repository.

In addition, important strides can be made in developing direct mappings from the heuristics to the critical parameters. The LINK-UP system [17, 19] allows designers to assess the design model

[1] www.survey.vt.edu is an example

in terms of specific levels of the critical parameters. The design model is a notion put forth by Norman in [72] that captures the designers understanding of how a system will work. In contrast, the users model represents how the user understands the system and how it works [72]. A goal for designers is to get the designers' and users' models to be as close as possible (ideally they would be the same). The LINK-UP system helps designers formulate and quantify the design models in terms of critical parameters [17]. What we also need is a method for quantifying the user models so that comparisons can be made to the design model representations. Lee et al. describe an early attempt at quantifying the user model through analytic evaluation [53]. However, they do not describe how they quantify the analytic results to the user model. Mapping our heuristics to the critical parameters is one possibility for providing this quantification. This mapping would 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.

A possible way to accomplish this mapping would be to reverse-engineer the creation process we used and trace back to the claims classifications. Because the heuristics are derived from claims with IRC classifications, it would be possible to trace back and find which ratings correspond to the heuristics. As an example, the "avoid the use of audio" heuristic could be mapped to the interruption parameter. Indeed, these mappings could then be verified through application of the heuristics, like in Chapter 6.

Another possible way to map the heuristics to the critical parameters involves quantifying each of the heuristics in terms of interruption, reaction, and comprehension. Doing so would allow a researcher 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. It is not clear how this quantification should proceed. One suggestion could be to rate the heuristics for a given system and different ratings indicate on which areas to focus re-design effort. For example, if heuristics 2, 5, and 7 received high ratings, then that could mean the designer needs to focus more on supporting comprehension. Identifying these mappings is non-trivial but could provide benefits to address some of the downsides to traditional heuristic evaluation discussed by Sauro and others [80, 21].

8.3.3 Critical Parameters

This work has also illustrated an important constraint on the application of the creation method to other system types. To be effective and useful, clearly defined system classes are a must. To define a system class, one needs to know the critical parameters for the design space, so that he/she can focus on the correct levels of each parameter. Identifying the critical parameters for a system type is not necessarily straightforward.

Future work needs to examine the process of identifying critical parameters for system types. One technique would involve extensive literature survey to determine initial taxonomies that describe the target systems. From this taxonomy, one could then determine the underlying important aspects that the systems hold for the end users. This user-focused approach is likely the best way to determine the critical parameters that hold the most importance in system design.

Another possible approach is to use the method described herein for deriving heuristics and tweak it to derive critical parameters. Much of the effort in the early stages (system inspection, claims analysis) can assist researchers in organizing disjoint systems based on commonalities

among them. A user-centered method like Scenario Based Design can provide some structure to the categorization of various claims, thereby focusing researchers' effort towards determining critical parameters. The entire method would not be applicable in this case but the analytic approach would at least give guidance for assessing claims and perhaps lead to identification of underlying parameters for the system class.

This realization could manifest due to the wording of claims. For example, if several claims deal with aspects of design causing distraction to a user, perhaps an important aspect of these systems is managing interruption. While this example is based on the critical parameters for notification systems, the underlying principle could work for other system types.

Another direction with critical parameters would be to investigate the "level" of the parameter. Indeed, as shown in this work, perhaps there are sub-classes of systems within a corner of the NS cube. Perhaps the physical instantiation of the software system (i.e. the platform) has important ramifications for design that impact the user goals associated with interruption, reaction, and comprehension. This would imply an extension of the current critical parameters to include "platform" or "medium" to capture this idea.

8.3.4 Design Knowledge Reuse

Creating heuristics by using the method described in this work produces significant amounts of design knowledge on multiple systems. Scenarios, claims, design issues, and the final heuristics are all reusable packets of design knowledge. Capturing this design knowledge and storing it for future use would be an important contribution to the notification systems research community.

Efforts exist in which this goal is being pursued. Specifically, significant effort is being put into the development of the LINK-UP system [17]. This system will provide notification systems researchers with claims, scenarios, and artifacts to guide design and testing. However, a clear connection to this system is critical for the method developed in this work. Tool support for the creation of scenarios and claims, as well as classifying and categorizing claims would allow broader application of our creation method. Some of the tools in the LINK-UP system could be used to support these tasks, but some changes would be necessary.

One example is apparent in the analytic module [17] of the LINK-UP system. This module guides the designer in creating and executing analytic formative evaluations. This involves testing claims and can be done through heuristic evaluation (as illustrated in Chapter 5). Supporting test creation and execution can easily be accomplished through software implementations of the testing procedure used in our experiment. However, tool support would need to be created for importing heuristics for inclusion in testing, as well as setting up the analytic test. A simple program could provide the researcher with access to existing sets of heuristics and he/she could then pick and choose which heuristics to include in the assessment of the chosen claims.

Future work efforts can target the inclusion of the creation method described herein as part of the analytic module of the LINK-UP system. One important aspect of this inclusion process will be tying the heuristics to the critical parameters associated with the system class. While the creation process relied upon the critical parameters for establishing the correct classifications for various claims, the resulting heuristics are more generic and do not immediately reflect the underlying critical parameters. Developing the relationship that each heuristic has with the critical parameters for the system class would be a valuable addition to the analytic module.

Bibliography

[1] Gregory Abowd and Elizabeth Mynatt. Charting past, present, and future research in ubiquitous computing. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 7(1):29–58, 2000.

[2] Gregory D. Abowd, Christopher G. Atkeson, Jason Brotherton, Tommy Enqvist, Paul Gulley, and Johan LeMon. 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)*, pages 440–447, April 1998.

[3] Gregory D. Abowd, Maria da Graca Pimentel, Bolot Kerimbaev, Yoshihide Ishiguro, and Mark Guzdial. Anchoring discussions in lecture: An approach to collaboratively extending classroom digital media. In *Proceedings of the Computer Support for Collaborative Learning (CSCL)1999 Conference*, pages 11–19, December 1999.

[4] Brian P. Bailey, Joseph A. Konstan, and John V. Carlis. The effects of interruptions on task performance, annoyance, and anxiety in the user interface. In *Proceedings of the 8th IFIP TC.13 International Conference on Human-Computer Interaction (INTERACT 2001)*, pages 593–601, Tokyo, Japan, July 2001.

[5] Kevin Baker, Saul Greenberg, and Carl Gutwin. Empirical development of a heuristic evaluation methodology for shared workspace groupware. In *ACM Conference on Computer Supported Cooperative work (CSCW'02)*, pages 96–105, New Orleans, LA, November 2002.

[6] Linda Ruth Bartram. *Enhancing Information Visualization with Motion*. PhD thesis, Simon Fraser University, Canada, 2001.

[7] Lyn Bartram. Enhancing visualizations with motion. In *Proceedings of the IEEE Symposium on Information Visualization (InfoVis '98)*, pages 13–16, Raleigh, NC, 1998.

[8] Lyn Bartram, Colin Ware, and Tom Calvert. Moving icons: Detection and distraction. In *Proceedings of the 8th IFIP TC.13 International Conference on Human-Computer Interaction (INTERACT 2001)*, pages 157–165, Tokyo, Japan, July 2001.

[9] Brandon Berry. Adapting heuristics for notification systems. In *Proceedings of 41st Annual ACM Southeast Conference (ACMSE'03)*, pages 144–149, Savannah, GA, March 2003.

[10] Richard Bowden, Pakorn Kaewtrakulpong, and Martin Lewin. Jeremiah: the face of computer vision. In *Proceedings of the 2nd international symposium on Smart graphics*, pages 124–128. ACM Press, 2002.

[11] Doug Bowman, J. Gabbard, and Deborah Hix. A survey of usability evaluation in virtual environments: Classification and comparison of methods. *Presence: Teleoperators and Virtual Environments*, 11(4):404–424, 2002.

[12] JJ Cadiz, Gina Danielle Venolia, Gavin Jancke, and Anoop Gupta. Sideshow: Providing peripheral awareness of important information. Technical Report MSR-TR-2001-83, Microsoft Research, Collaboration, and Multimedia Group, September 2001.

[13] John M. Carroll. *Making Use: Scenario-Based Design of Human-Computer Interactions*. The MIT Press, Cambridge, MA, 2000.

[14] John M. Carroll, Dennis Neale, Philip Isenhour, Mary B. Rosson, and D. Scott McCrickard. Notification and awareness: Synchronizing task-oriented collaborative activity. *International Journal of Human-Computer Studies, Special Edition on Design and Evaluation of Notification User Interfaces*, 58:605–632, 2003.

[15] John M. Carroll and Mary Beth Rosson. Getting around the task-artifact cycle: how to make claims and design by scenario. *ACM Transactions on Information Systems (TOIS)*, 10(2):181–212, April 1992.

[16] Jarinee Chattratichart and Jacqueline Brodie. Extending the heuristic evaluation method through contextualisation. In *Proceedings of the Human Factors and Ergonomics Society 46th Annual Meeting*, pages 641–645, 2002.

[17] C. M. Chewar, Edwin Bachetti, D. Scott McCrickard, and John Booker. 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, January 2004.

[18] C. M. Chewar, D. Scott McCrickard, Ali Ndiwalana, Chris North, Jon Pryor, and David Tessendorf. Secondary task display attributes: Optimizing visualizations for cognitive task suitability and interference avoidance. In *Proceedings of the Symposium on Data Visualization (VisSym '02)*, pages 165–171, Barcelona, Spain, 2002. Eurographics Association.

[19] C. M. Chewar, D. Scott McCrickard, and Alistair G. Sutcliffe. 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), to appear*, August 2004.

[20] Elizabeth F. Churchill, Les Nelson, Laurent Denoue, and Andreas Girgensohn. The plasma poster network: Posting multimedia content in public places. In *Ninth IFIP TC13 International Conference on Human-Computer Interaction (INTERACT03)*, pages 599–606, Zurich, Switzerland, September 2003.

[21] Gilbert Cockton and Alan Woolrych. Sale must end: Should discount methods be cleared off hci's shelves? *interactions*, september + october:13–18, 2002.

[22] SMART company site. The history of smart technologies. [ONLINE] http://www.smarttech.com/company/aboutus/history.asp.

[23] Edward Cutrell, Mary Czerwinski, and Eric Horvitz. Notification, disruption, and memory: Effects of messaging interruptions on memory and performance. In *Proceedings of the 8th IFIP TC.13 International Conference on Human-Computer Interaction (INTERACT 2001)*, pages 263–269, Tokyo, Japan, July 2001.

[24] Mary Czerwinski, Edward Cutrell, and Eric Horvitz. Instant messaging and interruption: Influence of task type on performance. In *Proceedings of OzCHI 2000*, pages 356–361, Sydney, Australia, December 2000.

[25] Mary Czerwinski, Edward Cutrell, and Eric Horvitz. Instant messaging: Effects of relevance and timing. In *Proceedings of HCI 2000*, September 2000.

[26] Mary Czerwinski, Desney S. Tan, and George G. Robertson. Women take a wider view. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 195–202. ACM Press, 2002.

[27] Paul Dourish and Sara Bly. Portholes: Supporting awareness in a distributed work group. In *Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI '92)*, pages 541–547, May 1992.

[28] Robert J. Dufresne, William J. Gerace, William J. Leonard, Jose P. Mestre, and Laura Wenk. *Classtalk*: A classroom communication system for active learning. *Journal of Computing in Higher Education*, 7(3–47), March 1996.

[29] Scott Elrod, Richard Bruce, Rich Gold, David Goldberg, Frank Halasz, William Janssen, David Lee, Kim McCall, Elin Pedersen, Ken Pier, John Tang, and Brent Welch. 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)*, pages 599–607. ACM Press, 1992.

[30] Geraldine Fitzpatrick, Tim Mansfield, Simon Kaplan, David Arnold, Ted Phelps, and Bill Segall. Augmenting the workaday world with elvin. In *Proceedings of European Conference on Computer Supported Cooperative Work(ECSCW'99)*, pages 431–451, Copenhagen, Denmark, September 1999. Kluwer Academic Publishers.

[31] Craig Ganoe, Jacob Somervell, Dennis Neale, Philip Isenhour, John M. Carroll, Mary Beth Rosson, and D. Scott McCrickard. 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)*, pages 21–30, November 2003.

[32] Wayne Gray and Marilyn Salzman. Damaged merchandise? a review of experiments that compare usability evaluation methods. *Human-Computer Interaction*, 13(4):203–261, 1998.

[33] Saul Greenberg. Peepholes: Low cost awareness of one's community. In *Conference Companion for the ACM Conference on Human Factors in Computing Systems (CHI '96)*, pages 206–207, Vancouver, BC, April 1996.

[34] Saul Greenberg and Chester Fitchett. Phidgets: easy development of physical interfaces through physical widgets. In *Proceedings of the 14th annual ACM symposium on User Interface Software and Technology (UIST'01)*, pages 209–218. ACM Press, 2001.

[35] Saul Greenberg, Geraldine Fitzpatrick, Carl Gutwin, and Steve Kaplan. Adapting the locales framework for heuristic evaluation of groupware. In *Proceedings of OZCHI*, pages 28–30, 1999.

[36] Saul Greenberg and Michael Rounding. The notification collage: Posting information to public and personal displays. In *Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI '01)*, pages 515–521, Seattle, WA, April 2001.

[37] Jonathan Grudin. Partitioning digital worlds: Focal and peripheral awareness in multiple monitor use. In *Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI'01)*, pages 458–465, Seattle, WA, April 2001.

[38] Carl Gutwin and Saul Greenberg. The mechanics of collaboration: Developing low cost usability evaluation methods for shared workspaces. In *IEEE 9th International Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises (WET-ICE'00)*, June 2000.

[39] Richard Hanowski and Barry Kantowitz. Driver memory retention of in-vehicle information system messages. [ONLINE] http:
citeseer.nj.nec.com/274172.html, 2000.

[40] H. Rex Hartson, Terence S. Andre, and Robert C. Williges. Criteria for evaluating usability evaluation methods. *International Journal of Human-Computer Interaction*, 13(4):373–410, 2001.

[41] Christopher G. Healey and James T. Enns. Large datasets at a glance: Combining textures and colors in scientific visualization. *IEEE Transactions on Visualization and Computer Graphics*, 5(2):145–167, 1999.

[42] D.A. Henderson and S. Card. Rooms: The use of multiple virtual workspaces to reduce space contention in a window-based graphical user interface. *ACM Transactions on Graphics*, 5(3):211–243, 1986.

[43] James M. Hudson, Jim Christensen, Wendy A. Kellogg, and Thomas Erickson. "I'd be overwhelmed, but it's just one more thing to do": Availability and interruption in research management. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 97–104. ACM Press, 2002.

[44] Hiroshi Ishii and Minoru Kobayashi. Clearboard: a seamless medium for shared drawing and conversation with eye contact. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 525–532. ACM Press, 1992.

[45] Hiroshi Ishii, Minoru Kobayashi, and Jonathan Grudin. Integration of inter-personal space and shared workspace: Clearboard design and experiments. In *Proceedings of the 1992 ACM conference on Computer-supported cooperative work*, pages 33–42. ACM Press, 1992.

[46] Hiroshi Ishii, Minoru Kobayashi, and Jonathan Grudin. Integration of interpersonal space and shared workspace: Clearboard design and experiments. *ACM Transactions on Information Systems (TOIS)*, 11(4):349–375, 1993.

[47] Hiroshi Ishii and Brygg Ulmer. Tangible bits: Towards seamless interfaces between people, bits, and atoms. In *Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI '97)*, pages 234–241, Atlanta, GA, March 1997.

[48] Robin Jeffries, James R. Miller, Cathleen Wharton, and Kathy Uyeda. User interface evaluation in the real world: a comparison of four techniques. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 119–124. ACM Press, 1991.

[49] Robert Johnson and Patricia Kuby. *Elementary Statistics*. Duxbury, 2000.

[50] Claire-Marie Karat, Robert Campbell, and Tarra Fiegel. Comparison of empirical testing and walkthrough methods in user interface evaluation. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 397–404. ACM Press, 1992.

[51] Kara A. Latorella. Investigating interruptions: Implications for flightdeck performance. In *Proceedings of the Human Factors and Ergonomics Society 40th Annual Meeting*, pages 249–253, Santa Monica, CA, 1996.

[52] Darryn Lavery, Gilbert Cockton, and Malcolm Atkinson. Comparison of evaluation methods using structured usability problem reports. *Behaviour and Information Technology*, 16(4/5):246–266, 1997.

[53] Jason Chong Lee, Sirong Lin, C. M. Chewar, and D. Scott McCrickard. 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), to appear*, November 2004.

[54] William Luebke, Michael Richmond, Jacob Somervell, and D. Scott McCrickard. An extensible framework for information visualization and collection. In *Proceedings of the 41st Annual ACM Southeast Conference (ACMSE'03)*, pages 365–370, March 2003.

[55] Paul P. Maglio and Christopher S. Campbell. Tradeoffs in displaying peripheral information. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 241–248. ACM Press, 2000.

[56] Jennifer Mankoff, Anind K. Dey, Gary Hsieh, Julie Kientz, Scott Lederer, and Morgan Ames. Heuristic evaluation of ambient displays. In *Proceedings of the conference on Human factors in computing systems*, pages 169–176. ACM Press, 2003.

[57] D. Scott McCrickard. Maintaining information awareness with Irwin. In *Proceedings of the World Conference on Educational Multimedia/Hypermedia and Educational Telecommunications (ED-MEDIA '99)*, pages 552–557, Seattle, WA, June 1999.

[58] D. Scott McCrickard. *Maintaining Information Awareness in a Dynamic Environment: Assessing Animation as a Communication Mechanism*. PhD thesis, Georgia Institute of Technology, Atlanta GA, 2000.

[59] D. Scott McCrickard, Richard Catrambone, C. M. Chewar, and John T. Stasko. Establishing tradeoffs that leverage attention for utility: Empirically evaluating information display in notification systems. *International Journal of Human-Computer Studies, Special Edition on Design and Evaluation of Notification User Interfaces*, 58:547–582, 2003.

[60] D. Scott McCrickard, Richard Catrambone, and John T. Stasko. Evaluating animation in the periphery as a mechanism for maintaining awareness. In *Proceedings of the IFIP TC.13 International Conference on Human-Computer Interaction (INTERACT 2001)*, pages 148–156, Tokyo, Japan, July 2001.

[61] D. Scott McCrickard and C. M. Chewar. Attuning notification design to user goals and attention costs. *Communications of the ACM*, 46(3):67–72, March 2003.

[62] D. Scott McCrickard, C. M. Chewar, Jacob P. Somervell, and Ali Ndiwalana. A model for notification systems evaluation: assessing user goals for multitasking activity. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 10(4):312–338, 2003.

[63] D. Scott McCrickard, David Wrighton, and Dillon Bussert. 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)*, pages 54–56, Arlington VA, September 2002.

[64] Daniel C. McFarlane. *Interruption of People in Human-Computer Interaction*. PhD thesis, George Washington University, Washington DC, 1998.

[65] Daniel C. McFarlane. Comparision of four primary methods for coordinating the interruption of people in human-computer interaction. *Human Computer Interaction*, 17(3):63–139, 2002.

[66] Rolph Molich and Jakob Nielsen. Improving a human-computer dialogue: What designers know about traditional interface design. *Communications of the ACM*, 33(3):338–348, 1990.

[67] Ali Ndiwalana, C. M. Chewar, Dillon Bussert, Jacob Somervell, and D. Scott McCrickard. Ubiquitous computing: By the people, for the people. In *Proceedings of the 41st Annual ACM Southeast Conference*, pages 24–29, Savannah, GA, March 2003.

[68] William M. Newman. Better or just different? on the benefits of designing interactive systems in terms of critical parameters. In *Proceedings of Designing Interactive Systems (DIS'97)*, pages 239–245, 1997.

[69] Jakob Nielsen and Thomas K. Landauer. A mathematical model of the finding of usability problems. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 206–213. ACM Press, 1993.

[70] Jakob Nielsen and R. L. Mack. *Usability Inspection Methods*. John Wiley and Sons, New York, NY, 1994.

[71] Jakob Nielsen and Rolph Molich. Heuristic evaluation of user interfaces. In *Proceedings of the ACM Conference on Human Factors and Computing Systems (CHI'90)*, pages 249–256, Seattle, WA, April 1990.

[72] Donald A. Norman. *The Design of Everyday Things*. MIT Press, 1998.

[73] Donald A. Norman and Stephen W. Draper, editors. *User Centered System Design: New Perspectives on Human-Computer Interaction*. Lawrence Erlbaum Associates, Hillsdale, New Jersey, 1986.

[74] Graham Pang and Hugh Liu. Led location beacon system based on processing of digital images. *IEEE Transactions on Intelligent Transportation Systems*, 2(3):135–150, 2001.

[75] Elin Rønby Pedersen, Kim McCall, Thomas P. Moran, and Frank G. Halasz. Tivoli: an electronic whiteboard for informal workgroup meetings. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 391–398. ACM Press, 1993.

[76] Johan Redström, Tobias Skog, and Lars Hallnäs. Informative art: Using amplified artworks as information displays. In *Proceedings of DARE 2000 conference on Designing Augmented Reality Environments*, pages 103–114. ACM Press, 2000.

[77] Mary Beth Rosson and John M. Carroll. *Usability Engineering: Scenario-Based Development of Human-Computer Interaction*. Morgan Kaufman, New York, NY, 2002.

[78] Daniel M. Russell, Clemens Drews, and Alison Sue. Social aspects of using large public interactive displays for collaboration. In *Ubiquitous Computing (UbiComp 2002)*, pages 229–236, Goteburg, Sweden, September 2002.

[79] Daniel M. Russell, Jay Trimble, and Roxana Wales. Two paths from the same place: Task driven and human-centered evolution of a group information surface. http://www.almaden.ibm.com/software/user/BlueBoard/index.shtml.

[80] Jeff Sauro. Premium usability: getting the discount without paying the price. *interactions*, 11(4):30–37, 2004.

[81] A. Sears. Heuristic walkthroughs: Finding the problems without the noise. *International Journal of Human Computer Interaction*, 9(3):213–234, 1997.

[82] Tobias Skog and Lars Erik Holmquist. 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*, pages 351–352, The Hague, Netherlands, April 2000.

[83] Markus Sohlenkamp and Greg Chwelos. Integrating communication, cooperation, and awareness: the diva virtual office environment. In *Proceedings of the Conference on Computer Supported Cooperative Work (CSCW'94)*, pages 331–343. ACM Press, 1994.

[84] Jacob Somervell, C. M. Chewar, and D. Scott McCrickard. Evaluating graphical vs. textual displays in dual-task environments. In *Proceedings of the 40th Annual ACM Southeast Conference (ACMSE'02)*, pages 153–160, Raleigh, NC, April 2002.

[85] Jacob Somervell, C. M. Chewar, D. Scott McCrickard, and Ali Ndiwalana. Enlarging usability for ubiquitous displays. In *Proceedings of the 41st Annual ACM Southeast Conference (ACMSE'03)*, pages 24–29, March 2003.

[86] Jacob Somervell, D. Scott McCrickard, Chris North, and Maulik Shukla. An evaluation of information visualization in attention-limited environments. In *Joint Eurographics/IEEE TCVG Symposium on Visualization (VisSym'02)*, pages 211–216, May 2002.

[87] Jacob Somervell, Ragavan Srinivasan, Omar Vasnaik, and Kim Woods. Measuring distraction and awareness caused by graphical and textual displays in the periphery. In *Proceedings of the 39th Annual ACM Southeast Conference (ACMSE'01)*, Athens, GA, March 2001.

[88] Jacob Somervell, Shahtab Wahid, and D. Scott McCrickard. Usability heuristics for large screen information exhibits. In *Proceedings of the Ninth IFIP TC13 International Conference on Human Computer Interaction (INTERACT'03)*, pages 904–907, Zurich, Switzerland, September 2003.

[89] Norbert A. Streitz, Jörg Geiβler, Torsten Holmer, Shin'ichi Konomi, Christian Müller-Tomfelde, Wolfgang Reischl, Petra Rexroth, Peter Seitz, and Ralf Steinmetz. i-land: an interactive landscape for creativity and innovation. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 120–127. ACM Press, 1999.

[90] Wei-Song Tan and R. R. Bishu. Which is a better method of web evaluation? a comparison of user testing and heuristic evaluation. In *Proceedings of the Human Factors and Ergonomics Society 46th Annual Meeting*, pages 1256–1260, 2002.

[91] J. Gregory Trafton, Erik Altmann, Derek Brock, and Farilee Mintz. Preparing to resume an interrupted task: effects of prospective goal encoding and retrospective rehearsal. *International Journal of Human-Computer Studies, Special Edition on Design and Evaluation of Notification User Interfaces*, 58:583–603, 2003.

[92] Maarten van Dantzich, Daniel Robbins, Eric Horvitz, and Mary Czerwinski. Scope: Providing awareness of multiple notifications at a glance. In *Proceedings of the 6th International Working Conference on Advanced Visual Interfaces (AVI '02)*. ACM Press, 2002.

[93] Mark Weiser and John Seely Brown. Designing calm technology. *PowerGrid Journal*, 1.01, July 1996.

[94] Q. Alex Zhao and John T. Stasko. What's Happening? the community awareness application. In *Conference Companion of the ACM Conference on Human Factors in Computing Systems (CHI'00)*, pages 253–254, The Hague, the Netherlands, April 2000.

[95] Q. Alex Zhao and John T. Stasko. What's Happening?: Promoting community awareness through opportunistic, peripheral interfaces. In *Proceedings of the Working Conference on Advanced Visual Interfaces (AVI'02)*, pages 69–74, Trento, Italy, May 2002.

Appendix A

Surveys Used in Preliminary Study

This appendix contains the statements used in the study of generic and specific evaluation methods. Each statement was rated by the participant using a 7-point Likert scale, ranging from strongly disagree to strongly agree.

We used this rating to assess how much the statement applied to the interface in question.

A.1 Generic Survey (used for both systems)

1. I could find natural break points in my task to look at the display so I wouldn't miss important information.
2. The interface did not distract my attention from my current task.
3. I was able to notice when new information appeared on the display without stopping my current work.
4. The interface provides an overall sense of the information.
5. The interface provides an ability to detect and understand clusters in the information.
6. The interface supports easy understanding of how information changes over time.
7. The interface supports easy understanding of links between different types of information.
8. The interface supports rapid reaction to the information.
9. Appropriate reactions to the information are obvious and intuitive.

A.2 GAWK Specific Survey

1. The interface provides an overall sense of the status of all groups.
2. I could tell how each group was doing in the project.
3. I could tell if the group work seemed one-sided.

4. I could tell which groups needed help.
5. I got a general sense of what the groups had been doing before today.
6. I could tell how the groups had worked over time.
7. I was quickly able to tell when useful information was available that I could look at more carefully when I had time.
8. If I were busy with something, changes in the display would NOT distract me.
9. If I were a part of the class, the information would be useful to me.

A.3 Photo News Board Specific Survey

1. The four quadrants were easily discernible and indicated the separate news areas.
2. I gained an overall sense of what events where happening in each area.
3. I could easily tell which news stories were recent and which stories were older.
4. The movement of the pictures allowed me to know when new stories appeared.
5. The movement of the pictures did not distract me from my other tasks.
6. I could tell what category was most interesting to the users in the room.
7. I like the access to current news stories provided by the interface.
8. When someone came into the room, I could tell what they were interested in from the highlighting.
9. Seeing a news story for a different person’s interest would cause me to start a conversation with that person about the story.

Appendix B

Scenarios for Systems

This appendix includes the scenarios that were created in the analysis of each of the systems used in the creation process. We have also included scenarios for the two other systems we used in the comparison study.

B.1 GAWK

B.1.1 Ms. Lang Surveys Student Groups

Ms. Lang, the sixth grade teacher, just finishes helping group one with posting their experiment results to the Virtual School project notebook. She needs to determine which group needs help next. Several students have hands in the air and two are lined up waiting to talk to her. She quickly looks at the GAWK and surveys the status of her groups. She sees that group five has not been as active as the others and decides to see if she can help them after taking care of the current line.

B.1.2 Karen Checks For Due Dates

Karen, an eighth grader, is busy working on typing out the processes her team used in their science experiment. She has several things to do and needs to prioritize by when things are due. As she finishes typing a sentence, she glances at the GAWK and notes that the introduction materials are due the next day. She decides to save work on the process description, and begins editing the introduction, hoping she can finish it before class is over so she won't have to do homework.

B.1.3 Mr. Bosk Assesses Progress

Mr. Bosk, the eighth grade teacher, is curious about how far along the sixth graders are in their parts of the projects. As he is giving a quiz, reading the questions to the class, he surveys the GAWK and notes that most of the sixth-grade groups are not as far along as he would hope, in regards to the upcoming deadline. He decides to contact Ms. Lang to discuss the situation after class.

B.2 Photo News Board

B.2.1 Jill Learns About Sports

Jill goes to the break room to relax for a few minutes. It is early in the day and she is the only one in the break room. She gets her cup of coffee and sets at the table to read her newspaper. While reading a few headlines she notices movement on the Photo News Board. She looks at the large screen display and watches as photos of interesting news stories get added to the display. She finds it interesting to see the display highlight different types of stories, from world news to entertainment to sports. She also notes that the display highlights more stories about sports than the other categories. She then recalls that she filled out a form when she came here that stated what she was most interested in (sports) and realizes that the large screen is simply highlighting stories that pertain to her interests.

B.2.2 Ted Learns About Jill

Ted goes into the break room to get a cup of coffee and pauses to watch the large screen display. Since he is the only one in the room, the display focuses on the entertainment section (that is what he is most interested in). After a few seconds, Jill a new coworker enters the break room. The display starts highlighting stories from the sports section (since Jill likes sports most). Ted notices the change in highlighting during a glance up from his paper, and starts some light conversation about the possible baseball strike. Jill is surprised to learn that Ted knew she liked sports just from the screen. They continue their conversation as they leave the break room.

B.2.3 Joe Breaks the Ice

Joe, the company boss, makes his way to the break room. He wants to get to know his coworkers a little better, so they do not always get quiet when he is around. He wants his employees to be able to come to him with problems if the need arises. Upon entering the break room, he notices Ted drinking some coffee and reading his newspaper. Joe proceeds to get a cup of coffee, and suddenly notices highlighting changes on the large screen. He notices a recent story on the new winner of the “American Idol” competition so he asks Ted what he thinks about the show. Surprised at first, Ted responds tersely then warms up and starts a more rich conversation with his boss. After the short conversation, Ted feels like he knows his boss a little better than he did before. Joe thinks the same about Ted.

B.3 Notification Collage

B.3.1 Bob Checks on Alice

Bob is working with Alice on some paperwork in their lab. Bob must frequently go over to Alice’s workstation to get her to look at the paperwork and sign documents that are needed. Alice, however, is often not at her workstation. Bob can uses the Notification Collage so he asks Alice to post a video feed of her workstation area. Bob will be able to continue doing his work, but stay aware

of Alice's presence by glancing at the Notification Collage. By glancing at the system when he takes a break or has a need to talk to Alice, Bob will be able to quickly realize if Alice is present at her workstation. He will no longer waste time walking across the lab to find Alice.

B.3.2 Bob Keeps Tabs

Bob is leading a project and is currently working at a different location for the rest of the month. All the members of the project team would like to keep Bob updated on what is completed. The team uses the Notification Collage to post notes about their work. Bob uses the Notification Collage to monitor notes that are posted by the members so that he can stay on top of what is going on. Bob can now continue to work with new workers at his new location and when he has time, he can look at the display and see if any notes are posted.

B.3.3 Dock Shares His Work

Dock, a professor running a lab, is proposing changes to the layout of the lab in order to account for new people using the lab constantly. Dock has drawn out a few layouts that could work well for the lab. He wants the others in the lab to give him opinions on the layouts. Knowing the lab users all use the NC, he posts his layouts as a slideshow on the Notification Collage for others to see. As users come by to the lab to work, they see the Notification Collage on their personal displays and the common large screen display. The Notification Collage initiates casual interaction among users who are around the large screen display and leads them to posting notes about the layouts. Users working at workstations glance at the slideshow and post their opinions. While Dock is busy writing a proposal, he notices the feedback about the lab layout, considers the opinions, and updates his proposal budget to include more equipment.

B.4 What's Happening?

B.4.1 Dill Checks on Traffic

Dill is working in his office, but must leave for home in time to make it to an invitation. Unfortunately, the time he plans to leave is the usual time for high traffic in nearby areas. To be able to plan his journey home, he can use the What's Happening? system running on the large screen display in the office. Every few minutes, Dill can glance at the display to see if the traffic information is being displayed as he continues to do his work. Once it is displayed, Dill can quickly determine the best route for him to take without much effort and will be able to make it home in time.

B.4.2 Alice Learns about Research

Alice is an active researcher in her department. Along with her work, she wishes to be able to have an idea about other research that other colleagues are working on. She knows that the What's Happening? goes through research pages that many of the researchers have. When Alice is in her lab, she can use the What's Happening? on the large screen display. While working, the display updates periodically with new content; she can glance at the pictures that are displayed and see

screenshots of interfaces that are posted at the researchers' sites. The system allows her to continue doing her work and at the same time facilitate her research interest by showing the ideas from other researchers' work.

B.4.3 Trudy Checks the Weather

Trudy works in a lab that is located in the center of her building. She usually works in this location, but some of her work involves doing to another building. During the winter's harsh weather conditions, Trudy would like to be informed of current weather conditions outside. The What's Happening? system can inform her of such information. Trudy can now continue to do her work, and a quick glance at the display will indicate the weather. When she must leave to go to another building, she can briefly glance at the What's Happening? display and quickly interpret the weather information. Now she can prepare herself for the weather outside.

B.5 Blue Board

B.5.1 Trudy Posts a Presentation

Trudy works for many people and must often make presentations to groups of people. Often, she needs to stay aware of scheduling changes in the available meeting rooms. As Trudy prepares a presentation, she notices a change in room assignments on the BlueBoard. She finishes her current powerpoint slide, then looks at the display to see if her room has been changed. Fortunatly, she still has the large conference room with the digital projector.

B.5.2 Alice Stays Informed

Alice works in a lab at her workstation. As she works throughout the day, she wishes to gain an understanding of announcements and news about the lab and company she works for. Alice knows that when the BlueBoard is not used, the system goes through the attract loop to show webpages with such announcements. As she works at her workstation, Alice can glance at the display at her own will and see which webpage is being displayed. If she sees a page that may be interesting, she can walk up and use the BlueBoard to browse through the page. The system allows her to continue doing her work and at the same time be informed of the current news.

B.5.3 Alice Checks Her Schedule

Alice is walking down the hallway, arms full of papers and books, when she suddenly realizes that she must check her schedule to make sure of the time her next meeting is. Alice is far from her workstation knows that it would be a waste of time to walk back. Fortunately, she notices the BlueBoard. She goes to the BlueBoard and it automatically badges-in and gives access to her posted information. She quickly sees her schedule and confirms the time of her next meeting. In this case, the BlueBoard provides an answer to a spontaneous need for information. She leaves, automatically badging-out and continues going on with her work.

B.6 Plasma Poster

B.6.1 Elizabeth Schedules a Presentation

Elizabeth goes to the kitchen to get some coffee. She glances at the Plasma Poster while pouring a cup and sees a new announcement for an upcoming presentation by her friend on his recent research effort. She goes over to the display and reads the date, time, and location of the presentation and makes a mental note to write it down in her schedule.

B.6.2 Alex and Kathy Make Plans

Alex is walking down the hall to his office, reading a memo, when he sees Kathy looking at the Plasma Poster. He stops by and sees that she is viewing a posting from a mutual friend in the building about an informal get together later in the week. He stops, asks her if she is going, and they make plans to car-pool. He then remembers they have a meeting and suggests they go over some information beforehand.

B.6.3 Jeff Enjoys Daily Humor

Jeff need a cup of coffee. He heads to kitchen to get a cup. He notices the empty pot and brews some more. While he fills up the pot with water, his attention falls on the Plasma Poster. He has seen it before and he looks at the bottom to see what the upcoming content may be. He turns back to his coffee-making task, sets the coffee-maker, then looks back at the Plasma Poster just in time to see the news article he noted from the preview list. He reads the article and chuckles to himself about it as he pours himself a cup of coffee.

B.7 Source Viewer

B.7.1 John Switches Source Content

John is busy with the noon newscast, listening for information from the producer and communicating timing information back. He is focused on his control board which is situated in front of him. He also must pay close attention to two oscilloscopes representing the volume levels on the commercial feed and the in-house news cast. A scheduled commercial break is coming up and he needs to make sure the station logo is played first. He glances at the large screen and sees that the logo is cued up on DSK1. He selects this source for the PRESET then waits for the cue from the on-screen talent. At the cue he switches the content. He then loads the next source in the PRESET window.

B.7.2 Bill Keeps Accurate Records

Bill is busy tracing the station breaks and marking them in his ledger. He gets a call from the station manager and he continues making the station breaks by glancing at the Source Viewer. He has just found out that there is an emergency break-in from CBS, posting the results of the New

Hampshire primaries. He easily switches the content but he must keep track of the planned station breaks. He periodically glances at the clock and the DSK and PRESET sources while viewing the highly interesting broadcast; so he can keep accurate records of when scheduled breaks were to occur. This information is vital for the station as they will have to settle accounts with various customers who had purchased air time that was pre-empted. The Source Viewer allows Bill to keep track of the commercials and the times at which they normally would have been broadcast.

B.7.3 Sarah Catches a Problem

Sarah is interning at WDBJ 7 in Roanoke. She wants to go into producing but must start out in the control room. Today she gets to do the source switching. She has watched John do it for a few days and she knows how to use the control panel but she is still nervous. As the noon newscast approaches, she must ensure that the sound levels are correct, that commercial breaks are switched correctly, and the live newscast gets broadcast. The Source Viewer provides all of the commercial and local news information and all she has to do is select the correct source and make the switch. After a few minutes into the newscast, she forgets to switch sources from the live newscast to an on-location agent. She missed the sound cue from the on-screen talent, and didn't hit the button on time. As a result, the switch was delayed about 10 seconds and the on-screen talent had to cover and move on. It wasn't a big deal because Sarah was able to make the switch, and cue up the next story. The Source Viewer allowed her to catch the problem very quickly and fix it.

Appendix C

System Claims

The following sections contain design tradeoffs in the form of upsides and downsides. These are the psychological impacts of specific design features. We abbreviate the typical claim format to include the feature in line with the tradeoff.

C.1 GAWK Upsides and Downsides

- +comparing groups/relative effort helps a teacher decide who needs help
- +showing deadlines helps students form goals
- +showing deadlines helps teachers focus students on tasks
- +showing work history shows group reputation for success
- +showing types of work completed gives a sense of what contributions are still necessary
- -public comparison of efforts may be embarrassing for students or grade groups
- +use of timeline to convey history, present status, upcoming requirements is a strong metaphor
- +3D rep of “today” in front helps people understand what’s closest
- +Banner suggests late-breaking changes and adds excitement
- -Timeline constrains lateral description space
- -3d Metaphor may be missed since it looks like a clickable object
- -banner info is often old, not exciting
- +stacking groups and separating with bold line allows easy comparison
- +“today” day is referred to most and should be in center and largest
- +showing work effort according to group and grade is most meaningful

- +banner on top allows people to notice changed items first
- +days are distinct as a thin vertical line
- -one-line banner is difficult to parse
- -space constrained for past items
- -entire 6th/8th grade performance difficult to realize
- -wasted space for future days
- +Red deadlines stand out and imply importance
- +green highlight stands out as item being described in banner
- +green highlight is not overly obtrusive
- +lack of other color reduces visual clutter and avoids confusion (unintended meaning)
- +blue highlight for new item flashing retains low obtrusiveness
- +dashed green highlight associates item versions
- -heavy use of red draws focus away from past and current screen areas
- -green highlight may not be visible on white background
- +San serif font and large size for easy reading from far away
- +icons show what work was a document, chat, or photo without requiring much space
- +chat icons show direction of chat and allow teachers to infer who's leading efforts
- -size constrains message length to 70-80 characters
- -small icons difficult to distinguish (from a distance)
- +lack of audio prevents distraction/annoying noises
- -rely on visual features totally to convey presence of new info, alerts, etc.
- +fading banner minimizes distraction
- +animated banner allows a lot of info to be cycled
- +transition of icon highlight corresponds with banner update to suggest association
- +new item flashing allows quick recognition of changes
- -flashing duration may be too short and go unnoticed
- -flashing with highlighting changes may be confusing

- -must watch whole cycle before getting text (banner) info for new item, causing frustration/confusion/interruption
- +Icons appear clickable and allow access to work item details
- -3d day presentation may appear clickable, when it is not
- -clickability of deadlines not conveyed
- +single screen is good for quickly recognizing changes or noticing changes over time
- +single screen is easy to learn
- +major screen changes occur at the very beginning of a session/day and aren't interruptive
- +new icons appear as work is completed, showing dynamic snapshot of progress
- -icons may be unexplained until banner cycle updates
- -difficult to anticipate highlighting sequence
- +icon selection is validated by banner updating and highlight movement
- +new icon flashing and presence provides acknowledgement of item submission
- +accommodates addition of groups and days to allow increase in project awareness
- +add deadlines or banner messages to promote new activities, actions, plans
- -too many groups/days makes interface too cluttered
- -messages added to banner may not be noticed quickly

C.2 Photo News Board Upsides and Downsides

- +seeing news summaries allows people to know the current status of news areas
- +seeing photos triggers curiosity about topics
- +new items arriving indicates when news is happening
- +showing history of photos allows tracking over time
- +showing interests of room occupants triggers conversation among them
- +quadrants allow awareness of news categories that people are interested in
- -showing interests publicly may be embarrassing or controversial
- -people may not recognize photos but are still interested in the topic (missed information)
- +lack of metaphor use allows for less prerequisite knowledge

- +pictorial representation of story will draw interest to the story
- +collage metaphor suggests loose connection between stories
- -may not associate photos with stories
- -collage metaphor may give disorderly haphazard appearance
- +categorizing stories and arranging them in quadrants allows for comparison of relative amount of news coverage for a topic
- +showing new items as larger and near center facilitates recognition
- +keeping older photos on edges gives sense of relative age of stories
- +showing older photos with new allows for tracking stories over time
- +banner design optimizes screen space for photos
- +small amount of white space separates individual photos
- -may not be obvious which quadrants are associated with which news topics
- -may not be clear that larger, center items are new; could be construed as importance
- -may not notice banner information immediately
- +color photos are appealing to look at
- +blue boundaries on quadrants is pleasing color, produces a calming effect
- +gray background on banner separates it from photo area
- +low contrast in all but one item focuses attention to the item (transparency)
- +using sans serif font facilitates reading
- +different sized pictures indicates relative age of pictures
- -font may be too small to read from long distances
- -smallest pictures may not be recognizable (on outer edges)
- +lack of audio prevents distraction/annoyance
- -relies on visual system for information changes
- +showing movement of pictures when new items arrives facilitates recognition of new items
- +highlighting disappears/reappears to allow quick recognition of change
- +text changes with highlighting aids association of the banner info to the pictures

- -multiple movements (transitioning photos and changed highlighting) can be confusing
- -abrupt changes in highlighting can cause distraction
- -movement of pictures prevents tracking of favorite stories (see where it goes)
- +showing options menu with arrow symbol implies you can click it to get the options
- -the fact that you can click on a picture is not immediately clear
- +single screen facilitates recognition of changes to information over time (easy to notice a difference)
- +single interface promotes learnability
- +options appear near option menu
- +movement pattern has pleasing effect and is not interruptive
- +clicking a picture brings it to center with story for easy reading/viewing
- -once a quadrant is full of pictures, older ones are moved off and lost
- -highlighting pattern is random which introduces uncertainty
- -a selected photo hides the photos behind it
- +photo selection is shown by making the photo appear in full color in center of screen with news story directly below it
- +options appear when options button is clicked
- -no immediate indication for when set of preferences changes (person leaves or enters room)
- -lack of association of preferences to a person may inhibit spontaneous communication (no way to tell who causes preference highlighting)
- +selecting speed, animation, and fonts promotes use
- +allows people to find most/least distracting settings
- -multiple users may not agree on settings
- -high speed setting slows processing (animation)

C.3 Notification Collage Upsides and Downsides

- + collage metaphor allows users to informally post information without any regards to organization
- + background supports the idea of graffiti. ie: put anything you want for everyone to see
- - lack of organization because of collage metaphor can hinder efforts to find an artifact
- + posting of live video, sticky notes, slide shows, etc. afford a wide variety of media forms
- + live video allows a quick and easy way of showing presence
- - lack of option limiting number of artifacts does not allow client to control clutter
- + filter options to hide artifacts can reduce some clutter
- - live video broadcast reduces privacy for users
- - full screen forces the user to use a secondary display
- + adjustable vertical bar lets user take control of the space
- + right side allows user to identify important artifacts
- - users define screen space
- + scattered arrangement of artifacts across screen accurately reflects the collage metaphor
- + background affords graffiti-like use
- - the background of the area on the right of the vertical bar does not convey the absence of artifact competition
- + font size is readable at large screen display
- - font size is too big for a personal display. a smaller size could save screen space
- - use of a fancier font decreases clarity
- + lack of audio decreases interruption and information overload, avoiding sensory overload
- - transition of the slideshow can distract users
- + customizing the rate at which the video feed updates allows user to control interruption
- - rapid animated updates to artifacts cause the artifact to move to the front at a constant rate
- - lack of organization frustrates users when trying to look for an artifact
- + lack of organization creates an informal virtual environment for users
- + artifacts placed on the right side of the vertical bar allows users to do their own organization

- + vertical bar clearly defines where artifacts must be arranged in order to preserve them
- + the system affords an environment in which users can be aware of each other
- + slideshow artifact allows multiple images to be shown in limited space
- - photo artifact uses a lot of space
- - chatting by using the post-it notes creates a distraction for others not chatting
- - system does not stop users from posting what may not be appropriate
- + vertical bar allows user to control what artifacts must remain in clear view
- - users hiding certain artifacts may miss important information
- - the design of the vertical bar is not intuitive. users may not know they can drag it
- - raising of context menu to initiate direct communication is not intuitive
- - left clicking on a webpage artifact does not take user directly to the page
- - left clicking on a picture does not open then picture in a picture editor
- + right clicking on an artifact raises the context menu to be able to contact the user that posted the artifact
- - competition between artifacts result in artifacts suddenly appearing on top, creating a distraction
- - slideshow artifact does not have smooth transitions
- - the system does not support receiving a receipt once an artifact is viewed by intended users
- + the video feed allows a user to constantly be aware of people working in the lab
- + the system allows users to filter out artifacts that aren't needed
- - the system does not allow a user to limit the number of artifacts viewable at any point in time
- - users do not have the ability to control the refresh rate of a video feed another person is posting
- - users can not configure transition of slideshow
- - users can not resize artifacts to tailor their use of space

C.4 What's Happening? Upsides and Downsides

- + collage metaphor allowed the system to place pictures in an unorganized fashion to use more screen space
- - system does not allow users to post information at their will
- - the system does not allow users to access the page from which the pictures were taken from
- - users do not have a way of going back to check what was on the system
- + since the system is designed to be opportunistic, users are not forced to regularly check the pictures on the system
- - the system did not allow users to filter certain types of images
- + using pictures as a single form of information delivery reduces the information clutter
- - full screen use in a screen saver does not allow the user to use their personal computer and monitor the system at the same time on a personal display
- + scattered arrangement of pictures across screen accurately reflects the collage metaphor
- + not using text descriptions allows more pictures to be shown
- - pictures uses a lot of space
- - bits of text displayed do not use as much space as they could for users to be able to read from greater distances
- - the black background of the system does not convey the collage metaphor
- - the background is a single color that may blend with pictures that may use the same color at the edges
- - font size is too small for a large screen display
- + smaller font size accurately conveys the system's concentration on pictures instead of text
- - use of a fancier font (italic) decreases clarity
- + lack of audio conveys the system's concentration on visual information
- - system does not use audio to alert users when new information is posted
- + lack of animated pictures or video decreases interruption produced by the system
- - lack of organization frustrates users if pictures are covered by others
- + lack of organization expresses the variations in types of pictures the system can display

- - users do not have a way of moving pictures around to be able to see pictures that may be partially covered
- - the system does not group pictures in the regions according to their source or type of information
- - pictures that are covered do not resurface to the front
- + the system affords an environment in which users can be aware of each other and their community
- - using an 18 month threshold for webpages allows the system to show information that may be up to 18 months old and irrelevant
- - the validity of the information displayed on the system depends on the validity of the websites used
- - the system does not afford posting information
- + the system affords glancing at the display to retrieve information opportunistically
- - lack of filters do not allow users to stop certain pieces of information
- - the system doesn't not allow any direct interaction since any type of input exits the screen-saver
- - trying to click on a picture or text does not take the user to the page it was found at
- - pictures appearing/disappearing suddenly may distract users since there is no apparent use of a fade in/out feature
- - the system does not provide any feedback on progress towards personal goals
- - there is no support for error recovery since the information displayed is driven by the system
- - irrelevant or old information can not be changed or removed
- - users can not change the amount of time a picture is displayed
- - users can not request a specific picture again
- - the system does not allow a user to limit the number of pictures viewable at any point in time
- - users can not configure the rate at which pictures from a source appear

C.5 Blue Board Upsides and Downsides

- + the finger-painting metaphor is accurately conveyed in the whiteboard
- + system allows rapid exchange of information by dragging and dropping
- + not using a keyboard or mouse simplifies all activities
- - lack of keyboard does not allow users to log into site they may want to see
- + system allows users to display calendars to quickly schedule activities
- + whiteboard provides a quick space in which users can write and/or draw
- + badging-in allows users to access their own posted information
- - system browser does not support all standard browsing features (ie. ctrl-f)
- + system displays web pages tailored to the location in the attract loop
- + dock clearly displays users that are badged-in on the right side of the screen
- + main screen area provides enough space for browsing and whiteboard use
- + background of the dock clearly separates the dock from the rest of the space
- + lack of a background in the main screen area affords using all the screen space at all times.
- - readability of fonts used will depend on information that is posted by users
- - lack of keyboard may lead to writing on the whiteboard that is illegible
- + lack of audio conveys the system's concentration on visual information
- - system does not use audio when an updated site is displayed in the attract loop to notify people of the update
- + use of alpha-blend to transition between webpage in the attract loop decreases interruption
- + badged-in users are clearly arranged on the right side
- - the system does not arrange information based on their type. they are all shown in the same screen area
- + the system afford quick exchange of information
- + users can be notified of webpages tailored to their interests based on their location
- + the back button afford quick retrieval of previously displayed pages in the attract loop
- + badge-in process allow quick access to personal information space
- + whiteboard affords quick and informal sketches

- - whiteboard affords illegible writing since there isn't a pen
- - system does not force a user to badge-out, leaving their account logged in so that others may pass information to fill their inbox
- + use of fingers makes the system interface more intuitive
- - the system does not allow a user to see another users page unless the other user is actually badged-in
- - the system does not allow a user to pass information they find on the BB to a user who is not badged-in
- + touching screen while using the whiteboard draws a line
- + touching a link on a page allows you to go to where the link points
- + touching and moving an object to the dock is an intuitive way of sending the information to a user
- + system uses alpha-blend to transition between webpages
- + emails that are sent about information exchanged act as progress receipts
- + social interaction around the display can result in feedback on information displayed
- - the system does not allow you to stop a transfer of information done to to another user by mistake
- - users can not configure the interface to accommodate their own needs
- + lack of configuration options allows all users to know exactly how the system will behave at all times since everyone will use the same interface
- - users can not change the amount of time a webpage is displayed in the attract loop

Appendix D

Electronic Problem Tree

This appendix contains the electronic problem tree containing all of the claims for the five systems (GAWK, Photo News Board, Notification Collage, What's Happening?, and Blue Board) as they were categorized and classified in the heuristics creation process.

Key	
GAWK =	GAWK
PNB =	Photo News Board
WH =	What's Happening?
NC =	Notification Collage
BB =	Blue Board

D.1 Activity Design

Activity design encompasses the capabilities of the system, the tasks that users can accomplish through the interface. Sub-categories include use of metaphors and supported/unsupported activities.

D.1.1 Metaphors

Metaphors in interface design refer to leveraging existing knowledge about other real world objects in the design to strengthen the user's understanding of the system. An example is the "desktop" metaphor found in most single-user computer systems.

High Interruption

- +Banner suggests late-breaking changes and adds excitement (GAWK)
- +pictorial representation of story will draw interest to the story (PNB)

- -collage metaphor may give disorderly haphazard appearance (PNB)
- -lack of organization because of collage metaphor can hinder efforts to find an artifact (NC)
- +collage metaphor allowed the system to place pictures in an unorganized fashion to use more screen space (WH)

Low Interruption

None

High Reaction

- +Banner suggests late-breaking changes and adds excitement (GAWK)
- +3D rep of "today" in front helps people understand what's closest (GAWK)
- +use of timeline to convey history, present status, upcoming requirements is a strong metaphor (GAWK)
- +pictorial representation of story will draw interest to the story (PNB)

Low Reaction

- -banner info is often old, not exciting (GAWK)

High Comprehension

- +background supports the idea of graffiti. ie: put anything you want for everyone to see (NC)
- +collage metaphor suggests loose connection between stories (PNB)
- +lack of metaphor use allows for less prerequisite knowledge (PNB)
- +3D rep of "today" in front helps people understand what's closest (GAWK)
- +use of timeline to convey history, present status, upcoming requirements is a strong metaphor (GAWK)

Low Comprehension

- -3d Metaphor may be missed since it looks like a clickable object (GAWK)
- -Timeline constrains lateral description space (GAWK)
- -banner info is often old, not exciting (GAWK)
- -may not associate photos with stories (PNB)

- -collage metaphor may give disorderly haphazard appearance (PNB)
- -lack of organization because of collage metaphor can hinder efforts to find an artifact (NC)
- +collage metaphor allowed the system to place pictures in an unorganized fashion to use more screen space (WH)

Unclassified

- +collage metaphor allows users to informally post information without any regards to organization (NC)
- +the finger painting metaphor is accurately conveyed in the whiteboard (BB)

D.1.2 Supported/Unsupported Activities

High Interruption

- +seeing photos triggers curiosity about topics (PNB)
- +new items arriving indicates when news is happening (PNB)
- -people may not recognize photos but are still interested in the topic (missed information) (PNB)
- +live video allows a quick and easy way of showing presence (NC)
- +posting of live video, sticky notes, slide shows, etc. allows information sharing (NC)
- -chatting by using the post-it notes creates a distraction for others not chatting (NC)

Low Interruption

- +using pictures as a single form of information delivery reduces the information clutter (WH)
- +since the system is designed to be opportunistic, users are not forced to regularly check the pictures on the system (WH)
- +filter options to hide artifacts can reduce some clutter (NC)
- -lack of filters do not allow users to stop certain pieces of information (WH)
- +the system affords glancing at the display to retrieve information opportunistically (WH)
- +vertical bar allows users to control what artifacts must remain in clear view (NC)
- -users hiding certain artifacts may miss important information (NC)
- +the video feed allows a user to constantly be aware of people working in the lab (NC)

High Reaction

- +showing deadlines helps students form goals (GAWK)
- +showing deadlines helps teachers focus students on tasks (GAWK)
- +showing types of work completed gives a sense of what contributions are still necessary (GAWK)
- +comparing groups/relative effort helps a teacher decide who needs help (GAWK)
- +showing interests of room occupants triggers conversation among them (PNB)
- +quadrants allow awareness of news categories that people are interested in (PNB)
- +seeing photos triggers curiosity about topics (PNB)
- +seeing news summaries allows people to know the current status of news areas (PNB)
- +filter options to hide artifacts can reduce some clutter (NC)
- +live video allows a quick and easy way of showing presence (NC)
- +posting of live video, sticky notes, slide shows, etc. allows information sharing (NC)
- +system allows rapid exchange of information by dragging and dropping (BB)
- +users can be notified of webpages tailored to their interests based on their location (BB)
- +the video feed allows a user to constantly be aware of people working in the lab (NC)

Low Reaction

- -people may not recognize photos but are still interested in the topic (missed information) (PNB)
- +since the system is designed to be opportunistic, users are not forced to regularly check the pictures on the system (WH)
- +filter options to hide artifacts can reduce some clutter (NC)
- +the system affords glancing at the display to retrieve information opportunistically (WH)

High Comprehension

- +comparing groups/relative effort helps a teacher decide who needs help (GAWK)
- +showing deadlines helps students form goals (GAWK)
- +showing deadlines helps teachers focus students on tasks (GAWK)
- +showing work history shows group reputation for success (GAWK)

- +showing types of work completed gives a sense of what contributions are still necessary (GAWK)
- -public comparison of efforts may be embarrassing for students or grade groups (GAWK)
- +showing history of photos allows tracking over time (PNB)
- +new items arriving indicates when news is happening (PNB)
- +showing interests of room occupants triggers conversation among them (PNB)
- +quadrants allow awareness of news categories that people are interested in (PNB)
- -showing interests publicly may be embarrassing or controversial (PNB)
- +seeing news summaries allows people to know the current status of news areas (PNB)
- +using pictures as a single form of information delivery reduces the information clutter (WH)
- +filter options to hide artifacts can reduce some clutter (NC)
- +live video allows a quick and easy way of showing presence (NC)
- +posting of live video, sticky notes, slide shows, etc. allows information sharing (NC)
- +system allows rapid exchange of information by dragging and dropping (BB)
- +system displays web pages tailored to the location in the attract loop (BB)
- -system does not force a user to badge-out, leaving their account logged in so that others may pass information to fill their inbox (BB)
- +users can be notified of webpages tailored to their interests based on their location (BB)
- +the system affords an environment in which users can be aware of each other (NC)
- +slideshow artifact allows multiple images to be shown in limited space (NC)
- +system affords an environment in which users can be aware of each other and their community (WH)
- +the video feed allows a user to constantly be aware of people working in the lab (NC)

Low Comprehension

- -people may not recognize photos but are still interested in the topic (missed information) (PNB)
- -the system did not allow users to filter certain types of images (WH)
- +filter options to hide artifacts can reduce some clutter (NC)
- -lack of filters do not allow users to stop certain pieces of information (WH)
- -using an 18 month threshold for webpages allows the system to show information that may be up to 18 months old and irrelevant (WH)
- - users hiding certain artifacts may miss important information (NC)
- -the system does not support receiving a receipt once an artifact is viewed by intended users (NC)
- -lack of association of preferences to a person may inhibit spontaneous communication (no way to tell who causes preferences highlighting) (PNB)
- -no immediate indication for when set of preferences changes (person leaves or enters room) (PNB)
- -the system does not provide any feedback on progress towards personal goals (WH)

Unclassified

- -the system does not allow users to access the page from which the pictures were taken from (WH)
- -system does not allow users to post information at their will (WH)
- -users do not have a way of going back to check what was on the system (WH)
- -live video broadcast reduces privacy for users (NC)
- -lack of option limiting number of artifacts does not allow client to control clutter (NC)
- +system allows users to display calendars to quickly schedule activities (BB)
- +whiteboard provides a quick space in which users can write and/or draw (BB)
- +badging-in allows users to access their own posted information (BB)
- -lack of keyboard does not allow users to log into site they may want to see (BB)
- -system browser does not support all standard browsing features (BB)

D.2 Information Design

D.2.1 Screen Space

High Interruption

- -one-line banner is difficult to parse (GAWK)
- +scattered arrangement of artifacts across screen accurately reflects the collage metaphor (NC)
- +scattered arrangement of pictures across screen accurately reflects the collage metaphor (WH)
- -photo artifact uses a lot of space (NC)
- -pictures uses a lot of space (WH)
- +adjustable vertical bar lets user take control of the space (NC)
- -users define screen space (NC)

Low Interruption

- +right side allows user to identify important artifacts (NC)
- +small amount of white space separates individual photos (PNB)
- +adjustable vertical bar lets user take control of the space (NC)
- -users define screen space (NC)

High Reaction

- +“today” day is referred to most and should be in center and largest (GAWK)

Low Reaction

none

High Comprehension

- +“today” day is referred to most and should be in center and largest (GAWK)
- +showing work effort according to group and grade is most meaningful (GAWK)
- +right side allows user to identify important artifacts (NC)
- +dock clearly displays users that are badged-in on the right side of the screen (BB)
- +small amount of white space separates individual photos (PNB)
- +banner design optimizes screen space for photos (PNB)

Low Comprehension

- -space constrained for past items (GAWK)
- -wasted space for future days (GAWK)
- +scattered arrangement of artifacts across screen accurately reflects the collage metaphor (NC)
- +scattered arrangement of pictures across screen accurately reflects the collage metaphor (WH)
- -bits of text displayed do not use as much space as they could for users to be able to read from greater distances (WH)

Unclassified

- -full screen forces the user to use a secondary display (NC)
- +main screen area provides enough space for browsing and whiteboard use (BB)
- -full screen use in a screen saver does not allow the user to use their personal computer and monitor the system at the same time on a personal display (WH)

D.2.2 Object and Background Colors

High Interruption

- -the background is a single color that may blend with pictures that may use the same color at the edges (WH)
- +low contrast in all but one item focuses attention to the item (transparency) (PNB)
- +color photos are appealing to look at (PNB)
- +green highlight stands out as item being described in banner (GAWK)
- +Red deadlines stand out and imply importance (GAWK)
- -heavy use of red draws focus away from past and current screen areas (GAWK)

Low Interruption

- +background of the dock clearly separates the dock from the rest of the space (BB)
- +blue boundaries on quadrants is pleasing color, produces a calming effect (PNB)
- +gray background on banner separates it from photo area (PNB)
- +green highlight is not overly obtrusive (GAWK)

- +blue highlight for new item flashing retains low obtrusiveness (GAWK)
- -green highlight may not be visible on white background (GAWK)
- +lack of other color reduces visual clutter and avoids confusion (unintended meaning) (GAWK)

High Reaction

- +low contrast in all but one item focuses attention to the item (transparency) (PNB)
- +color photos are appealing to look at (PNB)
- +green highlight stands out as item being described in banner (GAWK)

Low Reaction

- -green highlight may not be visible on white background (GAWK)

High Comprehension

- +low contrast in all but one item focuses attention to the item (transparency) (PNB)
- +gray background on banner separates it from photo area (PNB)
- +lack of other color reduces visual clutter and avoids confusion (unintended meaning) (GAWK)
- +Red deadlines stand out and imply importance (GAWK)
- +dashed green highlight associates item versions (GAWK)

Low Comprehension

- -the background is a single color that may blend with pictures that may use the same color at the edges (WH)
- -the background of the area on the right of the vertical bar does not convey the absence of artifact competition (NC)
- -heavy use of red draws focus away from past and current screen areas (GAWK)

Unclassified

- -the black background of the system does not convey the collage metaphor (WH)
- +lack of a background in the main screen area affords using all the screen space at all times (BB)
- +background affords graffiti-like use (NC)

D.2.3 Use of Fonts

High Interruption

- -use of a fancier font decreases clarity (NC)
- -readability of fonts used will depend on information that is posted by users (BB)
- -font size is too small for a large screen display (WH)
- -use of a fancier font decreases clarity (WH)
- -font may be too small to read from long distances (PNB)
- +icons show what work was a document, chat, or photo without requiring much space (GAWK)
- -small icons difficult to distinguish (from a distance) (GAWK)
- -smaller font size is harder to read on the large screen display (WH)

Low Interruption

- +font size is readable at large screen display (NC)
- -readability of fonts used will depend on information that is posted by users (BB)
- +using sans serif font facilitates reading (PNB)
- +San serif font and large size for easy reading from far away (GAWK)

High Reaction

none

Low Reaction

none

High Comprehension

- +using sans serif font facilitates reading (PNB)
- +different sized pictures indicates relative age of pictures (PNB)
- +San serif font and large size for easy reading from far away (GAWK)
- +icons show what work was a document, chat, or photo without requiring much space (GAWK)
- +chat icons show direction of chat and allow teachers to infer who’s leading efforts (GAWK)

Low Comprehension

- -font size is too small for a large screen display (WH)
- -font may be too small to read from long distances (PNB)
- -smallest pictures may not be recognizable (on outer edges) (PNB)
- -small icons difficult to distinguish (from a distance) (GAWK)
- -smaller font size is harder to read on the large screen display (WH)
- -size constrains message length to 76 characters (GAWK)

Unclassified

- -font size is too big for a personal display. a smaller size could save screen space (NC)
- -lack of keyboard may lead to writing on the whiteboard that is illegible (BB)

D.2.4 Use of Audio

High Interruption

- -rely on visual features totally to convey presence of new info, alerts, etc. (GAWK)
- -relies on visual system for information changes (PNB)

Low Interruption

- +lack of audio prevents distraction/annoying noises (GAWK)
- +lack of audio prevents distraction/annoyance (PNB)
- +lack of audio decreases interruption and information overload, avoiding sensory overload (NC)
- -system does not use audio to alert users when new information is posted (WH)
- -system does not use audio when an updated site is displayed in the attract loop to notify people of the update (BB)

High Reaction

none

Low Reaction

- -system does not use audio to alert users when new information is posted (WH)
- -system does not use audio when an updated site is displayed in the attract loop to notify people of the update (BB)

High Comprehension

- none

Low Comprehension

none

Unclassified

- +lack of audio conveys the system's concentration on visual information (BB)
- +lack of audio conveys the system's concentration on visual information (WH)

D.2.5 Use of Animation

High Interruption

- -rapid animated updates to artifacts cause the artifact to move to the front at a constant rate (NC)
- -transition of the slideshow can distract users (NC)
- +showing movement of pictures when new items arrives facilitates recognition of new items (PNB)
- +highlighting disappears/reappears to allow quick recognition of change (PNB)
- +text changes with highlighting aids association of the banner info to the pictures (PNB)
- -multiple movements (transitioning photos and changed highlighting) can be confusing (PNB)
- -abrupt changes in highlighting can cause distraction (PNB)
- +new item flashing allows quick recognition of changes (GAWK)
- -must watch whole cycle before getting text (banner) info for new item, causing frustration/confusion/interruption (GAWK)
- -flashing with highlighting changes may be confusing (GAWK)
- -highlighting pattern is random which introduces uncertainty (PNB)

- -competition between artifacts result in artifacts suddenly appearing on top, creating a distraction (NC)
- -pictures appearing/disappearing suddenly may distract users since there is no apparent use of a fade in/out feature (WH)
- -slideshow artifact does not have smooth transitions (NC)
- -difficult to anticipate highlighting sequence (GAWK)

Low Interruption

- +use of alpha-blend to transition between webpage in the attract loop decreases interruption (BB)
- +lack of animated pictures or video decreases interruption produced by the system (WH)
- +fading banner minimizes distraction (GAWK)
- -flashing duration may be too short and go unnoticed (GAWK)
- +major screen changes occur at the very beginning of a session/day and aren't interruptive (GAWK)
- +movement pattern has pleasing effect and is not interruptive (PNB)
- +system uses alpha-blend to transition between webpages (BB)

High Reaction

- +showing movement of pictures when new items arrives facilitates recognition of new items (PNB)
- +highlighting disappears/reappears to allow quick recognition of change (PNB)
- +new item flashing allows quick recognition of changes (GAWK)

Low Reaction

- -flashing duration may be too short and go unnoticed (GAWK)

High Comprehension

- +showing movement of pictures when new items arrives facilitates recognition of new items (PNB)
- +text changes with highlighting aids association of the banner info to the pictures (PNB)
- +animated banner allows a lot of info to be cycled (GAWK)
- +transition of icon highlight corresponds with banner update to suggest association (GAWK)

Low Comprehension

- -multiple movements (transitioning photos and changed highlighting) can be confusing (PNB)
- -movement of pictures prevents tracking of favorite stories (see where it goes) (PNB)
- -flashing with highlighting changes may be confusing (GAWK)
- -highlighting pattern is random which introduces uncertainty (PNB)
- -difficult to anticipate highlighting sequence (GAWK)
- -icons may be unexplained until banner cycle updates (GAWK)
- -once a quadrant is full of pictures, older ones are moved off and lost (PNB)

Unclassified

none

D.2.6 Grouping of Information Items

High Interruption

- -lack of organization frustrates users if pictures are covered by others (WH)
- -lack of organization frustrates users when trying to look for an artifact (NC)
- +showing new items as larger and near center facilitates recognition (PNB)
- -may not be obvious which quadrants are associated with which news topics (PNB)
- -may not be clear that larger, center items are new; could be construed as importance (PNB)
- +cyclic banner on top allows people to notice changed items first (GAWK)

Low Interruption

- +artifacts placed on the right side of the vertical bar allows users to do their own organization (NC)
- +vertical bar clearly defines where artifacts must be arranged in order to preserve them (NC)
- +categorizing stories and arranging them in quadrants allows for comparison of relative amount of news coverage for a topic (PNB)
- -may not notice banner information immediately (PNB)
- +stacking groups and separating with bold line allows easy comparison (GAWK)

High Reaction

- +showing new items as larger and near center facilitates recognition (PNB)
- +stacking groups and separating with bold line allows easy comparison (GAWK)
- +cyclic banner on top allows people to notice changed items first (GAWK)

Low Reaction

- none

High Comprehension

- +lack of organization expresses the variations in types of pictures the system can display (WH)
- +badged-in users are clearly arranged on the right side (BB)
- +artifacts placed on the right side of the vertical bar allows users to do their own organization (NC)
- +vertical bar clearly defines where artifacts must be arranged in order to preserve them (NC)
- +categorizing stories and arranging them in quadrants allows for comparison of relative amount of news coverage for a topic (PNB)
- +keeping older photos on edges gives sense of relative age of stories (PNB)
- +showing older photos with new allows for tracking stories over time (PNB)
- +stacking groups and separating with bold line allows easy comparison (GAWK)
- +weeks are distinct as a thin vertical line (GAWK)

Low Comprehension

- -the system does not group pictures in the regions according to their source or type of information (WH)
- -pictures that are covered do not resurface to the front (WH)
- -the system does not arrange information based on their type. they are all shown in the same screen area (BB)
- -lack of organization frustrates users when trying to look for an artifact (NC)
- -may not be clear that larger, center items are new; could be construed as importance (PNB)
- -may not notice banner information immediately (PNB)
- -entire 6th/8th grade performance difficult to realize (GAWK)

Unclassified

- -users do not have a way of moving pictures around to be able to see pictures that may be partially covered (WH)
- +lack of organization creates an informal virtual environment for users (NC)

D.3 Interaction Design

D.3.1 Recognition of Affordances

High Interruption

none

Low Interruption

none

High Reaction

none

Low Reaction

none

High Comprehension

none

Low Comprehension

none

Unclassified

- +use of fingers makes the systems interface more intuitive (BB)
- -whiteboard causes illegible writing since there isn't any pen (BB)
- +badge-in process allows quick access to personal information spaces (BB)
- -the system does not allows a user to pass information they find on the BB to a user who is not badged-in (BB)
- +the system affords quick exchange of information (BB)
- +whiteboard affords quick and informal sketches (BB)

- +the back button affords quick retrieval of previously displayed pages in the attract loop (BB)
- +icons appear clickable and allow access to work item details (GAWK)
- -clickability of deadlines not conveyed (GAWK)
- -they system does not allow a user to see another users page unless the other user is actually badged-in (BB)
- -the system does not afford posting information (WH)
- -the fact that you can click on a picture is not immediately clear (PNB)
- +showing options menu with arrow symbol implies you can click it to get the options (PNB)
- -raising of context menu to initiate direct communication is not intuitive (NC)
- -system does not stop from posting what may not be appropriate (NC)
- -the design of the vertical bar is not intuitive. users may not know they can drag it (NC)
- -the validity of the information displayed on the system depends on the validity of the web-sites used (WH)
- -3D day presentation may appear clickable, when it is not (GAWK)
- -users hiding certain artifacts may miss important information (filtering) (NC)

D.3.2 Behavior of Interface Control

High Interruption

none

Low Interruption

none

High Reaction

none

Low Reaction

none

High Comprehension

none

Low Comprehension

none

Unclassified

- +touching screen while using the whiteboard draws a line (BB)
- +touching and moving an object to the dock is an intuitive way of sending the information to a user (BB)
- +right clicking on an artifact raises the context menu to be able to contact the user that posted the artifact (NC)
- -the system does not allow any direct interaction since any type of input exists the screensaver (WH)
- -trying to click on a picture or text does not take the user to the page it was found at (WH)
- -left clicking on a picture does not open the picture in a picture editor (NC)
- -left clicking on a webpage artifact does not take the user directly to the page (NC)
- +not using a keyboard or mouse simplifies all activities (BB)

D.3.3 Expected Transition of State

High Interruption

none

Low Interruption

- +single screen is good for quickly recognizing changes or noticing changes over time (GAWK)

High Reaction

- +single screen facilitates recognition of changes to information over time (easy to notice a difference) (PNB)
- +new icons appear as work is completed, showing dynamic snapshot of progress (GAWK)

Low Reaction

none

High Comprehension

- +single screen facilitates recognition of changes to information over time (easy to notice a difference) (PNB)
- +single screen is good for quickly recognizing changes or noticing changes over time (GAWK)
- +new icons appear as work is completed, showing dynamic snapshot of progress (GAWK)

Low Comprehension

- -a selected photo hides the photos behind it (PNB)

Unclassified

- +clicking a picture brings it to the center with the story for easy reading/viewing (PNB)
- +options appear near option menu (PNB)
- +single screen is easy to learn (GAWK)
- +single interface promotes learnability (PNB)

D.3.4 Support for Undo/Error Recovery

High Interruption

none

Low Interruption

none

High Reaction

none

Low Reaction

none

High Comprehension

none

Low Comprehension

none

Unclassified

- -irrelevant or old information can not be changed or removed (WH)
- -the system does not allow you to top a transfer of information done to another user by mistake (BB)
- -there is no support for error recovery since the information displayed is driven by the system (WH)

D.3.5 Feedback about Progress on Task Goals

High Interruption

none

Low Interruption

none

High Reaction

none

Low Reaction

none

High Comprehension

- +icon selection is validated by banner updating and highlight movement (GAWK)
- +new icon flashing and presence provides acknowledgement of item submission (GAWK)

Low Comprehension

none

Unclassified

- +photo selection is shown by making the photo appear in full color in the center of the screen with news story directly below it (PNB)
- +options appear when options button is clicked (PNB)
- +e-mails that are sent about information exchanged act as progress receipts (BB)
- +social interaction around the display can result in feedback on information (BB)

D.3.6 Configurability Level for Usage Experience

High Interruption

- +the system allows users to filter out artifacts that aren't needed (NC)
- -the system does not allow a user to limit the number of pictures viewable at any point in time (WH)
- -users can not change the amount of time a picture is displayed (WH)
- -users can not configure the rate at which pictures from a source appear (WH)
- -users can not configure transition of slideshow (NC)
- -the system does not allow a user to limit the number of artifacts viewable at any point in time (NC)
- -users do not have the ability to control the refresh rate of a video feed another person is posting (NC)
- +allows people to find the most distracting settings (PNB)
- -too many groups/days makes interface too cluttered (GAWK)
- +customizing the rate at which the video feed updates allows user to control interruption (NC)
- -high speed settings slows processing (animation) (PNB)

Low Interruption

- -users can not change the amount of time a webpage is displayed in the attract loop (BB)
- +allows people to find the least distracting settings (PNB)
- -messages added to banner may not be noticed quickly (GAWK)
- +customizing the rate at which the video feed updates allows user to control interruption (NC)

High Reaction

none

Low Reaction

- -messages added to banner may not be noticed quickly (GAWK)
- -high speed settings slows processing (animation) (PNB)

High Comprehension

- -users can not change the amount of time a webpage is displayed in the attract loop (BB)
- +lack of configuration options allows users to know exactly how the system will behave at all times since everyone will use the same interface (BB)
- +accommodates addition of groups and days to allow increase in project awareness (GAWK)

Low Comprehension

- +the system allows users to filter out artifacts that aren't needed (NC)
- -the system does not allow a user to limit the number of pictures viewable at any point in time (WH)
- -users can not change the amount of time a picture is displayed (WH)
- -users can not resize artifacts to tailor their use of space (NC)
- -the system does not allow a user to limit the number of artifacts viewable at any point in time (NC)
- -too many groups/days makes interface too cluttered (GAWK)
- -high speed settings slows processing (animation) (PNB)

Unclassified

- -users can not configure the interface to accommodate their own needs (BB)
- -users can not request a specific picture again (BB)
- -multiple users may not agree on settings (PNB)
- -high speed settings slows processing (animation) (PNB)
- +selecting speed, animation, and fonts promotes use (PNB)
- +add deadlines or banner messages to promote new activities, actions, plans (GAWK)

Appendix E

High Level Issues

This appendix contains the high level issues identified from the problem tree. These issues capture some of the underlying causes in the nodes of the tree but are not general enough to be heuristics. They can serve as design guidelines and are available here for that purpose.

- Employ highly recognizable metaphors that use/stress organizational layout.
- Avoid metaphors that suggest haphazard or random layouts.
- Show the presence of information, but not the details of the information source.
- The magnitude or density of the information dictates visual representation.
- Avoid the use of audio.
- Introduce changes (new items) with slower, smooth transitions.
- Highlighting is an effective technique for showing relationships among data.
- Use cyclic displays with caution. If used, indicate “where” the display is in the cycle.
- Multiple, separate animations should be avoided.
- Indicate current and target locations if items move around the display.
- Text-based banner information should be located on the top or bottom, depending on focus and use of the information.
- Information grouped by time should be sequential.
- Information grouped by type can use random layouts within sub-groupings.
- Appropriate screen space should be delegated according to information importance.
- Important areas should be clearly recognizable.
- Use cool colors (blues, greens) for borders and backgrounds.
- Use warm colors (reds, yellows, oranges) for important information pieces and highlighting.

- Avoid heavy use of bright colors.
- Use sans serif fonts, in large size to facilitate reading.
- Use meaningful icons to show information, avoid text descriptions or delegate them to edges of the display.
- Avoid using multiple, small fonts.
- Eliminate or hide configurability controls.

Appendix F

Process Walkthrough

This appendix has a complete walkthrough of transforming the claims into heuristics, including the classification (Section 4.6.1) and categorization (Section 4.6.3) processes to form the problem tree. We also describe the extraction of high level issues from the problem tree (Section 4.7.2) as well as the final synthesis into heuristics (Section 4.7.3).

F.1 Classifying Claims

Extraction of claims from scenarios for each system was done through standard claims analysis techniques as described in [15] and [77]. Once these claims were identified, we needed to determine the impacts each claim had on the user goals of self-defined interruption, high comprehension, and appropriate reaction. This process is described in Section 4.6.1. Here we provide the results of that classification for all of the claims. The claim is listed on the left, with its associated classification on the right. I, R, and C refer to interruption, reaction, and comprehension respectively. "↑" and "↓" show how the particular parameter is effected by the claim. For example, ↑ *I* would indicate that the claim increased or otherwise caused higher allocation of attention from the user's primary task to the notification, whereas, ↓ *C* would suggest the claim caused or contributed to a lower understanding of the information. Italicized words indicate the reason for the classification. Multiple classifications arise from separate scenarios, and sometimes the classifications are in direct competition (↑ I and ↓ I).

Key	
GAWK =	GAWK
PNB =	Photo News Board
WH =	What's Happening?
NC =	Notification Collage
BB =	Blue Board

Claim **IRC classification**

+Banner suggests *late-breaking changes* and *adds excitement* (GAWK)	↑ I, ↑ R
+*pictorial representation* of story will *draw interest* to the story (PNB)	↑ I, ↑ R
-collage metaphor may give *disorderly haphazard appearance* (PNB)	↑ I, ↓ C
-lack of organization because of collage metaphor can *hinder efforts* to find an artifact (NC)	↑ I, ↓ C
+collage metaphor allowed the system to place pictures in an *unorganized fashion* to use more screen space (WH)	↑ I, ↓ C
+3D rep of “today” in front *helps people understand* what’s closest (GAWK)	↑ R, ↑ C
+use of timeline to *convey history*, present status, upcoming requirements is a strong metaphor (GAWK)	↑ R
-banner *info is often old, not exciting* (GAWK)	↓ R, ↑ C
+background supports the idea of graffiti. ie: put anything you want for *everyone to see* (NC)	↑ C
+collage metaphor suggests *loose connection between stories* (PNB)	↑ C
+lack of metaphor use allows for *less prerequisite knowledge* (PNB)	↑ C
-3d metaphor *may be missed* since it looks like a clickable object (GAWK)	↓ C
-timeline *constrains lateral description* space (GAWK)	↓ C
-people may not *associate photos* with stories (PNB)	↓ C

+collage metaphor allows users to informally post information without any regards to organization (NC)	
+the finger painting metaphor is accurately conveyed in the whiteboard(BB)	
+seeing photos *triggers curiosity* about topics (PNB)	↑ I, ↑ R
+new items *arriving indicates* when *news is happening* (PNB)	↑ I, ↑ C
-people may not *recognize photos* but are still *interested in the topic* (*missed information*) (PNB)	↑ I, ↓ R, ↓ C
+*live video* allows a quick and easy way of *showing presence* (NC)	↑ I, ↑ R
+*posting of live video*, sticky notes, slide shows, etc. allows *information sharing* (NC)	↑ I, ↑ R, ↑ C
-chatting by using the post-it notes *creates a distraction* for others not chatting (NC)	↑ I
+using pictures as a single form of information delivery *reduces the information clutter* (WH)	↓ I, ↑ C
+since the system is designed to be *opportunistic*, users are not forced to *regularly check* the pictures on the system (WH)	↓ I, ↓ R
+filter options to *hide artifacts* can *reduce some clutter* (NC)	↓ I, ↑ R, ↓ R, ↑ C, ↓ C
-lack of filters do not allow users to *stop certain pieces of information* (WH)	↓ I, ↓ C
+the system affords glancing at the	↓ I, ↓ R

display to *retrieve information opportunistically* (WH)	
+vertical bar allows *users to control* what artifacts must remain in *clear view* (NC)	↓ I
-users hiding certain artifacts may *miss important information* (NC)	↓ I, ↓ C
+the *video feed* allows a user to constantly be *aware of people* working in the lab (NC)	↓ I, ↑ R, ↑ C
+*showing deadlines* helps students form goals (GAWK)	↑ R, ↑ C
+*showing deadlines* helps teachers focus students on tasks (GAWK)	↑ R, ↑ C
+*showing types of work complete*d gives a *sense of* what contributions are still necessary (GAWK)	↑ R, ↑ C
+*comparing groups/relative effort helps a teacher decide* who needs help (GAWK)	↑ R, ↑ C
+showing *interests of room occupants triggers* conversation among them (PNB)	↑ R, ↑ C
+quadrants allow *awareness of news* categories that people are interested in (PNB)	↑ R, ↑ C
+seeing *news summaries* allows people to *know the current status* of news areas (PNB)	↑ R, ↑ C
+live video allows a quick and easy way of *showing presence* (NC)	↑ R, ↑ C
+system allows *rapid exchange of information* by dragging and dropping (BB)	↑ R, ↑ C
+users can be *notified of webpages*	↑ R, ↑ C

tailored to their interests based on their location (BB)	
+showing work history *shows group reputation for success* (GAWK)	↑ C
-public *comparison of efforts* may be embarrassing for students or grade groups (GAWK)	↑ C
+showing history of photos allows *tracking over time* (PNB)	↑ C
-*showing interests* publicly may be embarrassing or controversial (PNB)	↑ C
+system displays web pages *tailored to the location* in the attract loop (BB)	↑ C
-system does not force a user to badge-out, leaving their account logged in so that others may *pass information* to fill their inbox (BB)	↑ C
+the system affords an environment in which users can be *aware of each other* (NC)	↑ C
+slideshow artifact allows multiple *images to be shown* in limited space (NC)	↑ C
+system affords an environment in which users can be *aware of each other* and their community (WH)	↑ C
-the system did not allow users to *filter certain types of images* (WH)	↓ C
-using an 18 month threshold for webpages allows the system to show *information that may be up to 18 months old and irrelevant* (WH)	↓ C
-the system does not support *receiving*	↓ C

a receipt once an artifact is viewed by intended users (NC)

-lack of *association of preferences* to a person may inhibit spontaneous communication (no way to *tell who causes preferences highlighting*) (PNB) ↓ C

-no immediate indication for when set of *preferences changes* (person leaves or enters room) (PNB) ↓ C

-the system does not provide any *feedback on progress* towards personal goals (WH) ↓ C

-users *hiding certain artifacts* may *miss important information* (filtering) (NC) ↓ C

-the system does not allow users to access the page from which the pictures were taken from (WH)

-system does not allow users to post information at their will (WH)

-users do not have a way of going back to check what was on the system (WH)

-live video broadcast reduces privacy for users (NC)

-lack of option limiting number of artifacts does not allow client to control clutter (NC)

+system allows users to display calendars to quickly schedule activities (BB)

+whiteboard provides a quick space in which users can write and/or draw (BB)

+badging-in allows users to access their own posted information (BB)

-lack of keyboard does not allow users to log into site they may want to see(BB)

-system browser does not support all standard browsing features (BB)

-the system does not allow a user to pass information they find on the BB to a user who is not badged-in (BB)

+the system allows quick exchange of information (BB)

+the back button allows quick retrieval of previously displayed pages in the attract loop (BB)

-they system does not allow a user to see another users page unless the other user is actually badged-in (BB)

-the system does not allow posting information (WH)

-system does not stop from posting what may not be appropriate (NC)

-the validity of the information displayed on the system depends on the validity of the websites used (WH)

-one-line banner is *difficult to parse* (GAWK) ↑ I

+*scattered arrangement* of artifacts across screen accurately reflects the collage metaphor (NC) ↑ I, ↓ C

+*scattered arrangement* of pictures across screen accurately reflects the collage metaphor (WH) ↑ I, ↓ C

-photo artifact *uses a lot of space* (NC)	↑ I
-pictures *uses a lot of space* (WH)	↑ I
+adjustable vertical bar lets *user take control of the space* (NC)	↑ I, ↓ I
-*users define screen space* (NC)	↑ I, ↓ I
+right side allows user to *identify important artifacts* (NC)	↓ I, ↑ C
+small amount of white space *separates individual photos* (PNB)	↓ I, ↑ C
+*"today" day is referred to most* and should be in center and largest (GAWK)	↑ R, ↑ C
+showing work effort according to *group and grade is most meaningful* (GAWK)	↑ C
+dock *clearly displays users* that are badged-in on the right side of the screen (BB)	↑ C
+banner *design optimizes screen* space for photos (PNB)	↑ C
-*space constrained for past items* (GAWK)	↓ C
-*wasted space for future days* (GAWK)	↓ C
-bits of text displayed do not use as much space as they could for *users to be able to read* from greater distances (WH)	↓ C
-full screen forces the user to use a secondary display (NC)	
+main screen area provides enough space for browsing and whiteboard use(BB)	

-full screen use in a screen saver does not allow the user to use their personal computer and monitor the system at the same time on a personal display (WH)	
-the background is a single color that *may blend with pictures* that may use the same color at the edges (WH)	↑ I, ↓ C
+low contrast in all but one item *focuses attention* to the item (transparency)(PNB)	↑ I, ↑ R, ↑ C
+color photos are *appealing to look at* (PNB)	↑ I, ↑ R
+green highlight *stands out* as item being described in banner (GAWK)	↑ I, ↑ R
+Red deadlines *stand out* and *imply importance* (GAWK)	↑ I, ↑ C
-heavy use of red *draws focus away* from past and current screen areas(GAWK)	↑ I, ↓ C
+background of the dock clearly *separates the dock from the rest* of the space (BB)	↓ I
+blue boundaries on quadrants is *pleasing color, produces a calming effect* (PNB)	↓ I
+gray background on banner *separates it from photo area* (PNB)	↓ I, ↑ C
+green highlight is *not overly obtrusive* (GAWK)	↓ I
+blue highlight for *new item flashing retains low obtrusiveness* (GAWK)	↓ I
-green highlight *may not be visible* on white background (GAWK)	↓ I, ↓ R

+lack of other color *reduces visual clutter and avoids confusion* (unintended meaning) (GAWK)	↓ I, ↑ C
+dashed green highlight *associates item versions* (GAWK)	↑ C
-the background of the area on the right of the vertical bar does not *convey the absence* of artifact competition (NC)	↓ C
-the black background of the system does not convey the collage metaphor(WH)	
+lack of a background in the main screen area affords using all the screen space at all times (BB)	
+background affords graffiti-like use(NC)	
-use of a fancier font *decreases clarity* (NC)	↑ I
-*readability* of fonts used will depend on information that is posted by users (BB)	↑ I
-font size is *too small* for a large screen display (WH)	↑ I, ↓ C
-use of a fancier font *decreases clarity* (WH)	↑ I
-font may be *too small to read* from long distances (PNB)	↑ I, ↓ C
+icons *show what work was* a document, chat, or photo without requiring much space (GAWK)	↑ I, ↓ I
-small icons *difficult to distinguish* (from a distance) (GAWK)	↑ I, ↓ C
-smaller font size is *harder to read* on the large screen display (WH)	↑ I, ↓ C

+font size is *readable* at large screen display (NC)	↓ I
-*readability* of fonts used will depend on information that is posted by users(BB)	↓ I
+using sans serif font *facilitates reading* (PNB)	↓ I, ↑ C
+San serif font and large size for *easy reading* from far away (GAWK)	↓ I, ↑ C
+different sized pictures *indicates relative age* of pictures (PNB)	↑ C
+chat icons *show direction of chat* and allow teachers to *infer who's leading efforts* (GAWK)	↑ C
-smallest pictures may not be *recognizable* (on outer edges) (PNB)	↓ C
-size *constrains message length* to 76 characters (GAWK)	↓ C
-font size is too big for a personal display. a smaller size could save screen space (NC)	
-lack of keyboard may lead to writing on the whiteboard that is illegible (BB)	
-rely on visual features totally to *convey presence of new info*, alerts, etc. (GAWK)	↑ I
-relies on visual system for *information changes* (PNB)	↑ I
+lack of audio *prevents distraction/annoying noises* (GAWK)	↓ I
+lack of audio *prevents distraction/annoyance* (PNB)	↓ I

+lack of audio *decreases interruption* and information overload, avoiding sensory overload (NC)	↓ I
-system does not use audio to *alert users* when new information is posted (WH)	↓ I, ↓ R
-system does not use audio when an updated site is displayed in the attract loop to *notify people* of the update (BB)	↓ I, ↓ R
+lack of audio conveys the system's concentration on visual information (BB)	
+lack of audio conveys the system's concentration on visual information (WH)	
-rapid animated updates to artifacts cause the artifact to *move to the front* at constant rate (NC)	↑ I
-transition of the slideshow can *distract users* (NC)	↑ I
+showing movement of pictures when new items arrives *facilitates recognition* of new items (PNB)	↑ I, ↑ R, ↑ C
+highlighting disappears/reappears to allow *quick recognition of change* (PNB)	↑ I, ↑ R
+text changes with highlighting *aids association* of the banner info to the pictures (PNB)	↑ I, ↑ C
-multiple movements (transitioning photos and changed highlighting) *can be confusing* (PNB)	↑ I, ↓ C
-abrupt changes in highlighting *can cause distraction* (PNB)	↑ I

+new item flashing allows *quick recognition* of changes (GAWK)	↑ I, ↑ R
-must watch whole cycle before getting text (banner) info for new item, *causing frustration/confusion/interruption*(GAWK)	↑ I
-flashing with highlighting changes *may be confusing* (GAWK)	↑ I, ↓ C
-highlighting pattern is random which *introduces uncertainty* (PNB)	↑ I, ↓ C
-competition between artifacts result in artifacts *suddenly appearing on top, creating a distraction* (NC)	↑ I
-pictures appearing/disappearing suddenly may *distract users* since there is no apparent use of a fade in/out feature (WH)	↑ I
-slideshow artifact does not have *smooth transitions* (NC)	↑ I
-*difficult to anticipate* highlighting sequence (GAWK)	↑ I, ↓ C
+use of alpha-blend to transition between webpage in the attract loop *decreases interruption* (BB)	↓ I
+lack of animated pictures or video *decreases interruption* produced by the system (WH)	↓ I
+fading banner *minimizes distraction* (GAWK)	↓ I
-flashing duration may be too short and *go unnoticed* (GAWK)	↓ I, ↓ R
+major screen changes occur at the very beginning of a session/day and *aren't interruptive* (GAWK)	↓ I

+movement pattern has *pleasing effect* and is *not interruptive* (PNB)	↓ I
+system uses *alpha-blend to transition* between webpages (BB)	↓ I
+animated banner allows *a lot of info* to be cycled (GAWK)	↑ C
+transition of icon highlight corresponds with banner update to *suggest association* (GAWK)	↑ C
-movement of pictures *prevents tracking* of favorite stories (see where it goes) (PNB)	↓ C
-icons *may be unexplained* until banner cycle updates (GAWK)	↓ C
-once a quadrant is full of pictures, older ones are *moved off and lost* (PNB)	↓ C
-lack of organization *frustrates users* if pictures are covered by others (WH)	↑ I
-lack of organization *frustrates users* when trying to look for an artifact (NC)	↑ I, ↓ C
+showing new items as larger and near center *facilitates recognition* (PNB)	↑ I, ↑ R
-*may not be obvious* which quadrants are associated with which news topics (PNB)	↑ I
-*may not be clear* that larger, center items are new; could be *construed as importance* (PNB)	↑ I, ↓ C
+cyclic banner on top allows people to *notice changed items first* (GAWK)	↑ I, ↑ R
+artifacts placed on the right side of the vertical bar allows *users* to *do their own organization* (NC)	↓ I, ↑ C

+vertical bar *clearly defines where* artifacts must be arranged in order to preserve them (NC)	↓ I, ↑ C
+categorizing stories and arranging them in quadrants allows for *comparison of relative amount of news coverage* for a topic (PNB)	↓ I, ↑ C
-*may not notice* banner information immediately (PNB)	↓ I, ↓ C
+stacking groups and separating with bold line allows *easy comparison* (GAWK)	↓ I, ↑ R, ↑ C
+lack of organization *expresses the variations* in types of pictures the system can display (WH)	↑ C
+badged-in users are *clearly arranged* on the right side (BB)	↑ C
+keeping older photos on edges gives *sense of relative age* of stories (PNB)	↑ C
+showing older photos with new allows for *tracking stories over time* (PNB)	↑ C
+*weeks are distinct* as a thin vertical line (GAWK)	↑ C
-the system does not *group pictures* in the regions according to their source or *type of information* (WH)	↓ C
-*pictures that are covered* do not resurface to the front (WH)	↓ C
-the system does not *arrange information* based on their type. they are all shown in the same screen area (BB)	↓ C
-entire 6th/8th grade performance *difficult to realize* (GAWK)	↓ C

-users do not have a way of moving pictures around to be able to see pictures that may be partially covered (WH)

+lack of organization creates an informal virtual environment for users (NC)

+use of fingers makes the system's interface more intuitive (BB)

-whiteboard causes illegible writing since there isn't any pen (BB)

+icons appear clickable and allow access to work item details (GAWK)

-clickability of deadlines not conveyed (GAWK)

-the fact that you can click on a picture is not immediately clear (PNB)

+showing options menu with arrow symbol implies you can click it to get the options (PNB)

-raising of context menu to initiate direct communication is not intuitive (NC)

-the design of the vertical bar is not intuitive. users may not know they can drag it (NC)

-the validity of the information displayed on the system depends on the validity of the websites used (WH)

-3D day presentation may appear clickable, when it is not (GAWK)

+touching screen while using the

whiteboard draws a line (BB)

+touching and moving an object to the dock is an intuitive way of sending the information to a user (BB)

+right clicking on an artifact raises the context menu to be able to contact the user that posted the artifact (NC)

-the system does not allow any direct interaction since any type of input exits the screensaver (WH)

-trying to click on a picture or text does not take the user to the page it was found at (WH)

-left clicking on a picture does not open the picture in a picture editor (NC)

-left clicking on a webpage artifact does not take the user directly to the page (NC)

+not using a keyboard or mouse simplifies all activities (BB)

+single screen is good for *quickly recognizing* changes or *noticing* changes over time (GAWK) ↓ I, ↑ C

+single screen *facilitates recognition* of changes to information *over time* (easy to *notice a difference*) (PNB) ↑ R, ↑ C

+new icons *appear as work is completed*, showing dynamic *snapshot of progress* (GAWK) ↑ R, ↑ C

-a selected photo *hides the photos* behind it (PNB) ↓ C

+clicking a picture brings it to the center with the story for easy reading/viewing (PNB)

+options appear near option menu (PNB)

+single screen is easy to learn (GAWK)

+single interface promotes learnability (PNB)

-irrelevant or old information can not be changed or removed (WH)

-the system does not allow you to stop a transfer of information done to another user by mistake (BB)

-there is no support for error recovery since the information displayed is driven by the system (WH)

+icon selection is *validated* by banner updating and highlight movement (GAWK) ↑ C

+new icon flashing and presence *provides acknowledgement* of item submission (GAWK) ↑ C

+photo selection is shown by making the photo appear in full color in the center of the screen with news story directly below it (PNB)

+options appear when options button is clicked (PNB)

+e-mails that are sent about information exchanged act as progress receipts (BB)

+social interaction around the display can result in feedback on information (BB)

+the system allows users to *filter out* artifacts that aren't needed (NC)	↑ I, ↓ C
-the system does not allow a user to *limit the number of pictures* viewable at any point in time (WH)	↑ I, ↓ C
-users can not change the *amount of time* a picture is displayed (WH)	↑ I, ↓ C
-users can not configure the *rate at which pictures from a source appear* (WH)	↑ I
-users can not configure *transition* of slideshow (NC)	↑ I
-the system does not allow a user to *limit the number of artifacts viewable* at any point in time (NC)	↑ I, ↓ C
-users do not have the ability to control the *refresh rate of a video* feed another person is posting (NC)	↑ I
+allows people to find the *most distracting settings* (PNB)	↑ I
-too many groups/days makes *interface too cluttered* (GAWK)	↑ I, ↓ C
+customizing the rate at which the video feed updates allows user to *control interruption* (NC)	↑ I, ↓ I
-high speed settings *slows processing* (animation) (PNB)	↑ I, ↓ R, ↓ C
-users can not change the *amount of time* a webpage is displayed in the attract loop (BB)	↓ I, ↑ C
+allows people to find the *least distracting settings* (PNB)	↓ I

-messages added to banner *may not be noticed quickly* (GAWK)	↓ I, ↓ R
+lack of configuration options allows users to *know exactly* how the system will behave at all times since everyone will use the same interface (BB)	↑ C
+accommodates addition of groups and days to allow *increase in project awareness* (GAWK)	↑ C
-users can not resize artifacts to *tailor their use* of space (NC)	↓ C
-users can not configure the interface to accommodate their own needs (BB)	
-users can not request a specific picture again (BB)	
-multiple users may not agree on settings (PNB)	
+selecting speed, animation, and fonts promotes use (PNB)	
+add deadlines or banner messages to promote new activities, actions, plans (GAWK)	

Some of these claims have no classification. This occurs because it is not clear from the wording how the claim would impact any of the three parameters. The unclassified claims are discussed in Section 4.6.3.

F.2 Categorizing Claims

Here we provide the full categorization of the claims into their respective categories. These categories are taken from [77]. Activity is broken down into claims that deal with the ”presence and strength of metaphors” and “supported/unsupported activities” [77]. Information is broken down into “use of screen space”, “object and background color”, “use of fonts”, “use of audio”, “use of animation”, and “grouping of information items” [77]. Interaction is broken into “recognition

of affordances", "behavior of interface controls", "expected state transitions", "error recovery", "feedback on task progress", and "configurability levels and controls" [77].

The following provides the claims and the categories. Italicized words suggest the correct category. IRC ratings are included from the previous description of the classification of claims. Non-italicized claims are to be taken as a whole.

Claims	**IRC**	**SBD Category**
+*Banner* suggests late-breaking changes and adds excitement (GAWK)	↑ I, ↑ R	activity: metaphors
+*pictorial representation* of story will draw interest to the story (PNB)	↑ I, ↑ R	activity: metaphors
-*collage metaphor* may give disorderly haphazard appearance (PNB)	↑ I, ↓ C	activity: metaphors
-lack of organization because of *collage metaphor* can hinder efforts to find an artifact (NC)	↑ I, ↓ C	activity: metaphors
+*collage metaphor* allowed the system to place pictures in an unorganized fashion to use more screen space (WH)	↑ I, ↓ C	activity: metaphors
+*3D rep of "today"* in front helps people understand what's closest (GAWK)	↑ R, ↑ C	activity: metaphors
+use of *timeline* to convey history, present status, upcoming requirements is a strong metaphor (GAWK)	↑ R	activity: metaphors
-*banner* info is often old, not exciting (GAWK)	↓ R, ↑ C	activity: metaphors
+background supports the idea of *graffiti*. ie: put anything you want for everyone to see (NC)	↑ C	activity: metaphors
+*collage metaphor* suggests loose connection between stories (PNB)	↑ C	activity: metaphors
+lack of *metaphor* use allows for less prerequisite knowledge (PNB)	↑ C	activity: metaphors
-*3d Metaphor* may be missed since it	↓ C	activity: metaphors

looks like a clickable object (GAWK)

-*Timeline* constrains lateral description space (GAWK)	↓ C	activity: metaphors
-people may not associate *photos* with stories (PNB)	↓ C	activity: metaphors
+*collage metaphor* allows users to informally post information without any regards to organization (NC)		activity: metaphors
+the *finger painting metaphor* is accurately conveyed in the whiteboard(BB)		activity: metaphors
+seeing photos *triggers curiosity* about topics (PNB)	↑ I, ↑ R	activity: activities
+new items arriving indicates when *news is happening* (PNB)	↑ I, ↑ C	activity: activities
-people may not *recognize photos* but are still interested in the topic (missed information) (PNB)	↑ I, ↓ R, ↓ C	activity: activities
+live video allows a quick and easy way of *showing presence* (NC)	↑ I, ↑ R	activity: activities
+posting of live video, sticky notes, slide shows, etc. allows *information sharing* (NC)	↑ I, ↑ R, ↑ C	activity: activities
-*chatting by using the post-it notes* creates a distraction for others not chatting (NC)	↑ I	activity: activities
+using *pictures as* a single form of *information delivery* reduces the information clutter (WH)	↓ I, ↑ C	activity: activities
+since the system is designed to be opportunistic, users are not forced to regularly *check the pictures* on the system (WH)	↓ I, ↓ R	activity: activities
+filter options to *hide artifacts*	↓ I, ↑ R, ↑ C	activity: activities

can reduce some clutter (NC)		
-lack of filters do not allow users to *stop certain pieces of information* (WH)	↓ I, ↓ C	activity: activities
+the system affords glancing at the display to *retrieve information* opportunistically (WH)	↓ I, ↓ R	activity: activities
+vertical bar allows users to *control what artifacts must remain in clear view* (NC)	↓ I	activity: activities
-users *hiding certain artifacts* may miss important information (NC)	↓ I, ↓ C	activity: activities
+the video feed allows a user to constantly be *aware of people* working in the lab (NC)	↓ I, ↑ R, ↑ C	activity: activities
+showing deadlines helps students *form goals* (GAWK)	↑ R, ↑ C	activity: activities
+showing deadlines helps teachers *focus students on tasks* (GAWK)	↑ R, ↑ C	activity: activities
+showing types of work completed gives a sense of *what contributions are still necessary* (GAWK)	↑ R, ↑ C	activity: activities
+*comparing groups/relative effort* helps a teacher *decide who needs help* (GAWK)	↑ R, ↑ C	activity: activities
+*showing interests* of room occupants triggers conversation among them (PNB)	↑ R, ↑ C	activity: activities
+quadrants allow *awareness of news* categories that people are interested in (PNB)	↑ R, ↑ C	activity: activities
+seeing news summaries allows people to *know the current status of news* areas (PNB)	↑ R, ↑ C	activity: activities
+live video allows a quick and	↑ R, ↑ C	activity: activities

easy way of *showing presence* (NC)

+system allows *rapid exchange of information* by dragging and dropping (BB)	↑ R, ↑ C	activity: activities
+users can be *notified of webpages* tailored to their interests based on their location (BB)	↑ R, ↑ C	activity: activities
+*showing work history* shows group reputation for success (GAWK)	↑ C	activity: activities
-*public comparison of efforts* may be embarrassing for students or grade groups (GAWK)	↑ C	activity: activities
+showing history of photos allows *tracking over time* (PNB)	↑ C	activity: activities
-*showing interests publicly* may be embarrassing or controversial (PNB)	↑ C	activity: activities
+system *displays web pages* tailored to the location in the attract loop (BB)	↑ C	activity: activities
-system does not force a user to badge-out, leaving their account logged in so that others may *pass information* to fill their inbox (BB)	↑ C	activity: activities
+the system affords an environment in which users can be *aware of each other* (NC)	↑ C	activity: activities
+slideshow artifact allows multiple *images to be shown* in limited space (NC)	↑ C	activity: activities
+system affords an environment in which users can be *aware of each other* and their community (WH)	↑ C	activity: activities
-the system did not allow users to *filter certain types of images* (WH)	↓ C	activity: activities

-using an 18 month threshold for webpages allows the system to *show information* that may be up to 18 months old and irrelevant (WH)	↓ C	activity: activities
-the system does not support *receiving a receipt* once an artifact is viewed by intended users (NC)	↓ C	activity: activities
-lack of association of preferences to a person may inhibit *spontaneous communication* (no way to tell who causes preferences highlighting) (PNB)	↓ C	activity: activities
-no immediate indication for when set of *preferences changes* (person leaves or enters room) (PNB)	↓ C	activity: activities
-the system does not provide any feedback on *progress towards personal goals* (WH)	↓ C	activity: activities
-users *hiding certain artifacts* may miss important information (filtering) (NC)	↓ C	activity: activities
-the system does not allow users to *access the page* from which the pictures were taken from (WH)		activity: activities
-system does not allow users to *post information* at their will (WH)		activity: activities
-users do not have a way of going back to *check what was on the system* (WH)		activity: activities
-*live video broadcast* reduces privacy for users (NC)		activity: activities
-lack of option limiting number of artifacts does not allow client to *control clutter* (NC)		activity: activities
+system allows users to *display*		activity: activities

calendars to quickly schedule activities (BB)		
+whiteboard provides a quick space in which users can *write and/or draw* (BB)		activity: activities
+badging-in allows users to *access their own posted information* (BB)		activity: activities
-lack of keyboard does not allow users to *log into site* they may want to see(BB)		activity: activities
-system browser does not *support all standard browsing features* (BB)		activity: activities
-the system does not allow a user to *pass information* they find on the BB to a user who is not badged-in (BB)		activity: activities
+the system allows quick *exchange of information* (BB)		activity: activities
+the back button allows quick *retrieval of previously displayed pages* in the attract loop (BB)		activity: activities
-they system does not allow a user to *see another user's page* unless the other user is actually badged-in (BB)		activity: activities
-the system does not allow *posting information* (WH)		activity: activities
-system does not stop from *posting what may not be appropriate* (NC)		activity: activities
-the *validity of the information* displayed on the system depends on the validity of the websites used (WH)		activity: activities
-*one-line banner* is difficult to parse (GAWK)	↑ I	information: screen space
+scattered arrangement of artifacts	↑ I, ↓ C	information: screen space

across screen accurately reflects the collage metaphor (NC)		
+scattered arrangement of pictures *across screen* accurately reflects the collage metaphor (WH)	↑ I, ↓ C	information: screen space
-photo artifact uses a *lot of space* (NC)	↑ I	information: screen space
-pictures use a *lot of space* (WH)	↑ I	information: screen space
+adjustable vertical bar lets user take *control of the space* (NC)	↑ I, ↓ I	information: screen space
-users define *screen space* (NC)	↑ I, ↓ I	information: screen space
+*right side* allows user to identify important artifacts (NC)	↓ I, ↑ C	information: screen space
+small amount of *white space* separates individual photos (PNB)	↓ I, ↑ C	information: screen space
+"today" day is referred to most and should be in *center and largest* (GAWK)	↑ R, ↑ C	information: screen space
+showing work effort according to *group and grade* is most meaningful (GAWK)	↑ C	information: screen space
+dock clearly displays users that are badged-in on the *right side* of the screen (BB)	↑ C	information: screen space
+banner design optimizes *screen space* for photos (PNB)	↑ C	information: screen space
-*space* constrained for past items (GAWK)	↓ C	information: screen space
-wasted *space* for future days (GAWK)	↓ C	information: screen space
-bits of text displayed do not use as *much space* as they could for users to be able to read from greater distances (WH)	↓ C	information: screen space

-*full screen* forces the user to use a secondary display (NC)		information: screen space
+*main screen area* provides enough *space* for browsing and whiteboard use(BB)		information: screen space
-*full screen* use in a screen saver does not allow the user to use their personal computer and monitor the system at the same time on a personal display (WH)		information: screen space
-the background is a *single color* that may blend with pictures that may use the *same color* at the edges (WH)	↑ I, ↓ C	information: color
+*low contrast* in all but one item focuses attention to the item (transparency)(PNB)	↑ I, ↑ R, ↑ C	information: color
+*color photos* are appealing to look at (PNB)	↑ I, ↑ R	information: color
+*green highlight* stands out as item being described in banner (GAWK)	↑ I, ↑ R	information: color
+*red deadlines* stand out and imply importance (GAWK)	↑ I, ↑ C	information: color
-*heavy use of red* draws focus away from past and current screen areas(GAWK)	↑ I, ↓ C	information: color
+*background* of the dock clearly separates the dock from the rest of the space (BB)	↓ I	information: color
+*blue boundaries* on quadrants is *pleasing color*, produces a calming effect (PNB)	↓ I	information: color
+*gray background* on banner separates it from photo area (PNB)	↓ I, ↑ C	information: color
+*green highlight* is not overly obtrusive (GAWK)	↓ I	information: color

+*blue highlight* for new item flashing retains low obtrusiveness (GAWK)	↓ I	information: color
-*green highlight* may not be visible on *white background* (GAWK)	↓ I, ↓ R	information: color
+lack of other *color* reduces visual clutter and avoids confusion (unintended meaning) (GAWK)	↓ I, ↑ C	information: color
+*dashed green highlight* associates item versions (GAWK)	↑ C	information: color
-the *background* of the area on the right of the vertical bar does not convey the absence of artifact competition (NC)	↓ C	information: color
-the *black background* of the system does not convey the collage metaphor(WH)		information: color
+lack of a *background* in the main screen area affords using all the screen space at all times (BB)		information: color
+*background* affords graffiti-like use(NC)		information: color
-use of a *fancier font* decreases clarity (NC)	↑ I	information: fonts, icons
-*readability of fonts* used will depend on information that is posted by users (BB)	↑ I, ↓ I	information: fonts, icons
-*font size* is too small for a large screen display (WH)	↑ I, ↓ C	information: fonts, icons
-use of a *fancier font* decreases clarity (WH)	↑ I	information: fonts, icons
-*font may be too small* to read from long distances (PNB)	↑ I, ↓ C	information: fonts, icons
+*icons* show what work was a document, chat, or photo without requiring much space (GAWK)	↑ I, ↓ I	information: fonts, icons

-*small icons* difficult to distinguish (from a distance) (GAWK)	↑ I, ↓ C	information: fonts, icons
-*smaller font size* is harder to read on the large screen display (WH)	↑ I, ↓ C	information: fonts, icons
+*font size* is readable at large screen display (NC)	↓ I	information: fonts, icons
+using *sans serif font* facilitates reading (PNB)	↓ I, ↑ C	information: fonts, icons
+*sans serif font* and large size for easy reading from far away (GAWK)	↓ I, ↑ C	information: fonts, icons
+*different sized pictures* indicates relative age of pictures (PNB)	↑ C	information: fonts, icons
+*chat icons* show direction of chat and allow teachers to infer who's leading efforts (GAWK)	↑ C	information: fonts, icons
-smallest *pictures* may not be recognizable (on outer edges) (PNB)	↓ C	information: fonts, icons
-*size* constrains message length to 76 *characters* (GAWK)	↓ C	information: fonts, icons
-*font size* is too big for a personal display. a smaller size could save screen space (NC)		information: fonts, icons
-lack of keyboard may lead to *writing* on the whiteboard that is *illegible* (BB)		information: fonts, icons
-*rely on visual features totally* to convey presence of new info, alerts, etc. (GAWK)	↑ I	information: audio
-*relies on visual system* for information changes (PNB)	↑ I	information: audio
+*lack of audio* prevents distraction/annoying noises (GAWK)	↓ I	information: audio

+*lack of audio* prevents distraction/annoyance (PNB)	↓ I	information: audio
+*lack of audio* decreases interruption and information overload, avoiding sensory overload (NC)	↓ I	information: audio
-system *does not use audio* to alert users when new information is posted (WH)	↓ I, ↓ R	information: audio
-system *does not use audio* when an updated site is displayed in the attract loop to notify people of the update (BB)	↓ I, ↓ R	information: audio
+*lack of audio* conveys the system's concentration on visual information (BB)		information: audio
+*lack of audio* conveys the system's concentration on visual information (WH)		information: audio
-rapid *animated updates* to artifacts cause the artifact to move to the front at constant rate (NC)	↑ I	information: animation
-*transition of the slideshow* can distract users (NC)	↑ I	information: animation
+showing *movement of pictures* when new items arrives facilitates recognition of new items (PNB)	↑ I, ↑ R, ↑ C	information: animation
+*highlighting disappears/reappears* to allow quick recognition of change (PNB)	↑ I, ↑ R	information: animation
+*text changes with highlighting* aids association of the banner info to the pictures (PNB)	↑ I, ↑ C	information: animation
-*multiple movements* (transitioning photos and changed highlighting) can be confusing (PNB)	↑ I, ↓ C	information: animation

-abrupt *changes in highlighting* can cause distraction (PNB)	↑ I	information: animation
+new *item flashing* allows quick recognition of changes (GAWK)	↑ I, ↑ R	information: animation
-must watch *whole cycle* before getting text (banner) info for new item, causing frustration/confusion(GAWK)	↑ I	information: animation
-*flashing with highlighting changes* may be confusing (GAWK)	↑ I, ↓ C	information: animation
-*highlighting pattern* is random which introduces uncertainty (PNB)	↑ I, ↓ C	information: animation
-competition between artifacts result in artifacts *suddenly appearing on top*, creating a distraction (NC)	↑ I	information: animation
-*pictures appearing/disappearing* suddenly may distract users since there is no apparent use of a fade in/out feature (WH)	↑ I	information: animation
-slideshow artifact does not have *smooth transitions* (NC)	↑ I	information: animation
-difficult to anticipate *highlighting sequence* (GAWK)	↑ I, ↓ C	information: animation
+use of *alpha-blend* to transition between webpage in the attract loop decreases interruption (BB)	↓ I	information: animation
+lack of *animated pictures or video* decreases interruption produced by the system (WH)	↓ I	information: animation
+*fading banner* minimizes distraction (GAWK)	↓ I	information: animation
-*flashing duration* may be too short and go unnoticed (GAWK)	↓ I, ↓ R	information: animation
+major *screen changes* occur at the very	↓ I	information: animation

beginning of a session/day and aren't interruptive (GAWK)		
+*movement pattern* has pleasing effect and is not interruptive (PNB)	↓ I	information: animation
+*animated banner* allows a lot of info to be cycled (GAWK)	↑ C	information: animation
+*transition of icon highlight* corresponds with banner update to suggest association (GAWK)	↑ C	information: animation
-*movement of pictures* prevents tracking of favorite stories (see where it goes) (PNB)	↓ C	information: animation
-icons may be unexplained until *banner cycle updates* (GAWK)	↓ C	information: animation
-once a quadrant is full of pictures, older ones are *moved off* and lost (PNB)	↓ C	information: animation
-*lack of organization* frustrates users if pictures are covered by others (WH)	↑ I	information: grouping, layout
-*lack of organization* frustrates users when trying to look for an artifact (NC)	↑ I, ↓ C	information: grouping, layout
+showing new items as *larger and near center* facilitates recognition (PNB)	↑ I, ↑ R	information: grouping, layout
-may not be obvious which *quadrants* are associated with which news topics (PNB)	↑ I	information: grouping, layout
-may not be clear that *larger, center* items are new; could be construed as importance (PNB)	↑ I, ↓ C	information: grouping, layout
+cyclic *banner on top* allows people to notice changed items first (GAWK)	↑ I, ↑ R	information: grouping, layout
+artifacts placed on the *right side* of the vertical bar allows users to do their own organization (NC)	↓ I, ↑ C	information: grouping, layout

+*vertical bar* clearly *defines where* artifacts must be arranged in order to preserve them (NC)	↓ I, ↑ C	information: grouping, layout
+categorizing stories and *arranging them in quadrants* allows for comparison of relative amount of news coverage for a topic (PNB)	↓ I, ↑ C	information: grouping, layout
-may not notice *banner information* immediately (PNB)	↓ I, ↓ C	information: grouping, layout
+*stacking groups* and *separating with bold line* allows easy comparison (GAWK)	↓ I, ↑ R, ↑ C	information: grouping, layout
+*lack of organization* expresses the variations in types of pictures the system can display (WH)	↑ C	information: grouping, layout
+badged-in users are clearly arranged on the *right side* (BB)	↑ C	information: grouping, layout
+keeping older photos on *edges* gives sense of relative age of stories (PNB)	↑ C	information: grouping, layout
+showing *older photos with new* allows for tracking stories over time (PNB)	↑ C	information: grouping, layout
+weeks are distinct as a *thin vertical line* (GAWK)	↑ C	information: grouping, layout
-the system does not *group pictures* in the regions according to their source or type of information (WH)	↓ C	information: grouping, layout
-*pictures that are covered* do not resurface to the front (WH)	↓ C	information: grouping, layout
-the system does not *arrange information* based on their type. they are all shown in the *same screen area* (BB)	↓ C	information: grouping, layout
-*entire 6th/8th grade performance* difficult to realize (GAWK)	↓ C	information: grouping, layout

-users do not have a way of *moving pictures around* to be able to see pictures that may be *partially covered* (WH)	information: grouping, layout
+*lack of organization* creates an informal virtual environment for users (NC)	information: grouping, layout
+use of fingers makes the system's *interface more intuitive* (BB)	interaction: affordances
-whiteboard *causes illegible writing* since there isn't any pen (BB)	interaction: affordances
+icons *appear clickable* and allow access to work item details (GAWK)	interaction: affordances
-*clickability of deadlines* not conveyed (GAWK)	interaction: affordances
-the fact that you *can click* on a picture is *not immediately clear* (PNB)	interaction: affordances
+showing options menu with arrow symbol *implies you can click it* to get the options (PNB)	interaction: affordances
-raising of context menu to initiate direct communication is *not intuitive* (NC)	interaction: affordances
-the design of the vertical bar is *not intuitive*. users may not know they *can drag it* (NC)	interaction: affordances
-3D day presentation may *appear clickable*, when it is not (GAWK)	interaction: affordances
+*touching screen* while using the whiteboard draws a line (BB)	interaction: behavior of controls
+*touching and moving an object* to the dock is an intuitive way of	interaction: behavior of controls

sending the information to a user (BB)		
+*right clicking on an artifact* raises the context menu to be able to contact the user that posted the artifact (NC)		interaction: behavior of controls
-the system does not allow any *direct interaction* since any type of input exits the screensaver (WH)		interaction: behavior of controls
-trying to *click on a picture or text* does not take the user to the page it was found at (WH)		interaction: behavior of controls
-*left clicking on a picture* does not open the picture in a picture editor (NC)		interaction: behavior of controls
-*left clicking on a webpage* artifact does not take the user directly to the page (NC)		interaction: behavior of controls
+*not using a keyboard or mouse* simplifies all activities (BB)		interaction: behavior of controls
+*single screen* is good for quickly recognizing changes or noticing changes over time (GAWK)	↓ I, ↑ C	interaction: state transition
+*single screen* facilitates recognition of changes to information over time (easy to notice a difference) (PNB)	↑ R, ↑ C	interaction: state transition
+new *icons appear as work is completed*, showing dynamic snapshot of progress (GAWK)	↑ R, ↑ C	interaction: state transition
-a *selected photo hides the photos behind it* (PNB)	↓ C	interaction: state transition
+*clicking a picture brings it to the center* with the story for easy reading/viewing (PNB)		interaction: state transition

+*options appear near option menu* (PNB)		interaction: state transition
+*single screen* is easy to learn (GAWK)		interaction: state transition
+*single interface* promotes learnability (PNB)		interaction: state transition
-irrelevant or old information can not be *changed or removed* (WH)		interaction: error/undo
-the system does not allow you to *stop a transfer* of information done to another user *by mistake* (BB)		interaction: error/undo
-there is no support for *error recovery* since the information displayed is driven by the system (WH)		interaction: error/undo
+icon *selection is validated* by banner updating and highlight movement (GAWK)	↑ C	interaction: feedback
+new icon flashing and presence provides *acknowledgement of item* submission (GAWK)	↑ C	interaction: feedback
+photo *selection is shown* by making the photo appear in full color in the center of the screen with news story directly below it (PNB)		interaction: feedback
+options *appear when* options button is *clicked* (PNB)		interaction: feedback
+e-mails that are sent about information exchanged act as *progress receipts* (BB)		interaction: feedback
+social interaction around the display can result in *feedback on information* (BB)		interaction: feedback
+the system *allows users to filter* out artifacts that aren't needed (NC)	↑ I, ↓ C	interaction: configurability

-the system does not *allow a user* to limit the number of pictures viewable at any point in time (WH)	↑ I, ↓ C	interaction: configurability
-*users can not change* the amount of time a picture is displayed (WH)	↑ I, ↓ C	interaction: configurability
-*users can not configure* the rate at which pictures from a source appear (WH)	↑ I	interaction: configurability
-*users can not configure* transition of slideshow (NC)	↑ I	interaction: configurability
-the system *does not allow a user* to limit the number of artifacts viewable at any point in time (NC)	↑ I, ↓ C	interaction: configurability
-*users do not have the ability to* control the refresh rate of a video feed another person is posting (NC)	↑ I	interaction: configurability
+*allows people to find* the most distracting settings (PNB)	↑ I	interaction: configurability
-*too many groups/days* makes interface too cluttered (GAWK)	↑ I, ↓ C	interaction: configurability
+*customizing the rate* at which the video feed updates allows user to control interruption (NC)	↑ I, ↓ I	interaction: configurability
-*high speed settings* slows processing (animation) (PNB)	↑ I, ↓ R, ↓ C	interaction: configurability
-*users can not change* the amount of time a webpage is displayed in the attract loop (BB)	↓ I, ↑ C	interaction: configurability
+*allows people to find* the least distracting settings (PNB)	↓ I	interaction: configurability
-*messages added to banner* may not be noticed quickly (GAWK)	↓ I, ↓ R	interaction: configurability

+*lack of configuration options* allows users to know exactly how the system will behave at all times since everyone will use the same interface (BB)	↑ C	interaction: configurability
+accommodates *addition of groups* and days to allow increase in project awareness (GAWK)	↑ C	interaction: configurability
-*users can not resize* artifacts to tailor their use of space (NC)	↓ C	interaction: configurability
-*users can not configure* the interface to accommodate their own needs (BB)		interaction: configurability
-*users can not request* a specific picture again (BB)		interaction: configurability
-multiple *users may not agree* on settings (PNB)		interaction: configurability
+*selecting* speed, animation, and fonts promotes use (PNB)		interaction: configurability
+*add deadlines or banner messages* to promote new activities, actions, plans (GAWK)		interaction: configurability

F.3 From Claims to Issues

We now provide the detailed analysis of the claims that produced the 22 design issues. These design issues can provide designers with input in early design phases as well as suggest possible design flaws if used in heuristic evaluation (see Section 4.7.2). Some claims did not directly impact the issue formation, others had high impacts on the wordings. Unclassified claims are not included here.

After classification and categorization, we had manageable chunks of claims that could then be probed for underlying design issues. The issues found in this effort are described below. We relied upon the wordings of the claims and in many instances, several claims from different systems had similar wordings or addressed similar issues. Furthermore, some claims seemed obscure or strange and did not necessarily contribute to the formulation of the issues.

Claims	**Resulting Issue**
+*Banner* suggests late-breaking changes	Employ highly recognizable

and adds excitement(GAWK)

+ *pictorial representation* of story will draw interest to the story (PNB)

-*collage metaphor* may give *disorderly haphazard appearance* (PNB)

-*lack of organization* because of *collage metaphor* can *hinder efforts* to find an artifact (NC)

+*collage metaphor* allowed the system to place pictures in an *unorganized fashion* to *use more screen space* (WH)

+*3D rep of "today"* in front helps people *understand what's closest* (GAWK)

+*use of timeline* to convey history, present status, upcoming requirements is a strong metaphor (GAWK)

-*banner info is often old*, not exciting (GAWK)

+background supports the idea of graffiti. ie: put anything you want for everyone to see (NC)

+*collage metaphor* suggests loose connection between stories (PNB)

+lack of metaphor use allows for less prerequisite knowledge (PNB)

-*3d metaphor may be missed* since it looks like a clickable object (GAWK)

-*timeline* constrains lateral description space (GAWK)

-people may not associate photos with stories (PNB)

---

metaphors that use/stress organizational layout.

Avoid metaphors that suggest haphazard or random layouts.

+*seeing photos triggers curiosity* about topics (PNB)

-people *may not recognize photos* but are still interested in the topic

(missed information) (PNB)

+ *live video* allows a quick and easy way of showing presence (NC)

+ *posting of live video*, sticky notes, slide shows, etc. allows information sharing (NC)

-*chatting by using the post-it notes creates a distraction* for others not chatting (NC)

+*using pictures* as a single form of information delivery reduces the information clutter (WH)

+the system affords glancing at the display to *retrieve information opportunistically* (WH)

+vertical bar allows *users to control* what artifacts must remain in clear view (NC)

+the *video feed* allows a user to constantly be aware of people working in the lab (NC)

+ *showing deadlines* helps students form goals (GAWK)

+ *showing deadlines* helps teachers focus students on tasks (GAWK)

+ *showing types of work completed* gives a sense of what contributions are still necessary (GAWK)

Show the presence of information but not the details of the information source.

The magnitude or density of the information dictates visual

representation.

+*showing interests* of room occupants *triggers conversation* among them (PNB)

+*quadrants allow awareness* of news categories that people are interested in (PNB)

+seeing *news summaries* allows people to *know the current status* of news areas (PNB)

+showing *work history* shows group reputation for success (GAWK)

+showing *history of photos* allows tracking over time (PNB)

+*system displays web pages* tailored to the location in the attract loop (BB)

+slideshow artifact allows multiple images to be *shown in limited space* (NC)

+system affords an environment in which users can be *aware of each other* and their community (WH)

-the system did not allow users to *filter certain types of images* (WH)

-the system does not support receiving a receipt once an artifact is viewed by intended users (NC)

-lack of association of preferences to a person may inhibit spontaneous communication (no way to tell who causes preferences highlighting) (PNB)

-no immediate indication for when set of preferences changes (person leaves or enters room) (PNB)

-the system does not provide any

feedback on progress towards personal goals (WH)

-users hiding certain artifacts may miss important information (filtering) (NC)

+ scattered arrangement of artifacts across screen accurately reflects the collage metaphor (NC)

+ scattered arrangement of pictures across screen accurately reflects the collage metaphor (WH)

-photo artifact uses a lot of space (NC)

-pictures uses a lot of space (WH)

+adjustable vertical bar lets user take control of the space (NC)

- users define screen space (NC)

+showing work effort according to group and grade is most meaningful (GAWK)

+banner design optimizes screen space for photos (PNB)

- space constrained for past items (GAWK)

- wasted space for future days (GAWK)

-bits of text displayed do not use as much space as they could for users to be able to read from greater distances (WH)

Appropriate screen space should be delegated according to information importance.

Important areas should be clearly recognizable.

Information grouped by time should be sequential.

Information grouped by type can use random layouts within subgroups.

+*low contrast* in all but one item focuses attention to the item (transparency)(PNB)

+*color photos* are appealing to look at (PNB)

Use cool colors (blues, greens) for borders and backgrounds.

Use warm colors (reds, yellows, oranges) for important information pieces and highlighting.

+*green highlight* stands out as item being described in banner (GAWK)

+*red deadlines* stand out and imply importance (GAWK)

-*heavy use of red* draws focus away from past and current screen areas(GAWK)

+*blue boundaries* on quadrants is pleasing color, produces a *calming effect* (PNB)

+*gray background* on banner separates it from photo area (PNB)

+*green highlight* is *not overly obtrusive* (GAWK)

+*blue highlight* for new item flashing *retains low obtrusiveness* (GAWK)

Avoid heavy use of bright colors.

-use of a *fancier font decreases clarity* (NC)

-*font size is too small* for a large screen display (WH)

-use of a *fancier font decreases clarity* (WH)

-*font may be too small to read* from long distances (PNB)

+*icons show* what work was a document, chat, or photo *without requiring much space* (GAWK)

-*small icons difficult to distinguish* (from a distance) (GAWK)

-*smaller font size is harder to read* on the large screen display (WH)

Use sans serif fonts, in large size to facilitate reading.

Avoid using multiple, small fonts.

Use meaningful icons to show information, avoid text descriptions or delegate them to edges of the display.

Text-based banner information should be located on the top or bottom, depending on focus and use of the information.

+*font size is readable at large screen* display (NC)

+using *sans serif font facilitates* reading (PNB)

+*san serif font and large size* for *easy reading* from far away (GAWK)

+*chat icons* show direction of chat and allow teachers to infer who's leading efforts (GAWK)

-*size constrains message length to 76* characters (GAWK)

-*font size is too big for a personal display*, a smaller size could save screen space (NC)

-*rely on visual features totally* to convey presence of new info, alerts, etc. (GAWK)

-*relies on visual system* for information changes (PNB)

+lack of audio prevents distraction/annoying noises (GAWK)

+lack of audio prevents distraction/annoyance (PNB)

+lack of audio decreases interruption and information overload, avoiding sensory overload (NC)

-system *does not use audio* to alert users when new information is posted (WH)

-system *does not use audio* when an updated site is displayed in the attract loop to notify people of the update (BB)

Avoid the use of audio.

-*rapid animated updates* to artifacts cause the artifact to *move to the front* at constant rate (NC)

- transition of the slideshow can distract users (NC)

+*showing movement of pictures* when new items arrives *facilitates recognition* of new items (PNB)

+*highlighting* disappears/reappears to allow *quick recognition* of change (PNB)

+text changes with highlighting aids association of the banner info to the pictures (PNB)

-multiple movements (transitioning photos and changed highlighting) can be confusing (PNB)

-abrupt changes in highlighting can cause distraction (PNB)

+new item flashing allows quick recognition of changes (GAWK)

-*must watch whole cycle* before getting text (banner) info for new item, *causing frustration/confusion/interruption* (GAWK)

-flashing with highlighting changes may be confusing (GAWK)

-highlighting pattern is *random* which *introduces uncertainty* (PNB)

-competition between artifacts result in artifacts *suddenly appearing on top, creating a distraction* (NC)

-pictures appearing/disappearing suddenly may distract users since there is no

Introduce changes (new items) with slower, smooth transitions.

Highlighting is an effective technique for showing relationships among data.

Use cyclic displays with care, indicate where the display is in the cycle.

Multiple, separate animations should be avoided.

Indicate current and target locations if items move around the display.

apparent use of a fade in/out feature (WH)

-slideshow artifact does not have smooth transitions (NC)

-difficult to anticipate highlighting sequence (GAWK)

+use of alpha-blend to transition between webpage in the attract loop *decreases interruption* (BB)

+lack of animated pictures or video decreases interruption produced by the system (WH)

+fading banner minimizes distraction (GAWK)

-flashing duration may be too short and go unnoticed (GAWK)

+major screen changes occur at the very beginning of a session/day and aren't interruptive (GAWK)

+movement pattern has pleasing effect and is not interruptive (PNB)

+system uses alpha-blend to transition between webpages (BB)

+animated banner allows a lot of info to be cycled (GAWK)

+transition of icon highlight corresponds with banner update to suggest association (GAWK)

-movement of pictures prevents tracking of favorite stories (see where it goes) (PNB)

-icons may be unexplained until banner cycle updates (GAWK)

-once a quadrant is full of pictures,
older ones are moved off and lost (PNB)

+*users can not change* the amount of time a picture is displayed (WH)

+*users can not configure* the rate at which pictures from a source appear (WH)

+*users can not configure* transition of slideshow (NC)

-allows people to find the most distracting settings (PNB)

-too many groups/days makes *interface too cluttered* (GAWK)

+*customizing the rate* at which the video feed updates allows user to *control interruption* (NC)

-*users can not change* the amount of time a webpage is displayed in the attract loop (BB)

+allows people to find the *least distracting settings* (PNB)

+*lack of configuration* options allows users to know exactly how the system will behave at all times since everyone will use the same interface (BB)

-*users can not resize artifacts* to tailor their use of space (NC)

Eliminate or hide configurability controls.

F.4 Issues to Heuristics

To reduce the number of issues, we went through one more level of discussion and mitigation in an attempt to reduce the amount of knowledge into a more manageable set. Again we looked at the wordings of the issues to see if we could extract even higher level design guidance from them. The goal was to find eight to ten heuristics that captured the overall design issues.

We provide the complete listing of the issues and the resulting heuristics from this process. The process is described in Section 4.7.3 and the results are given below.

Issues	**Heuristic**
Use cyclic displays with caution. If used, indicate ”where” the display is in the cycle.	Use cyclic displays with caution if used indicate where the display is in the cycle.
Show the presence of information, but not the details of the information source. The magnitude or density of the information dictates visual representation. Use meaningful icons to show information, avoid text descriptions or delegate them to edges of the display.	Show the presence of information but not the details.
Avoid the use of audio.	Avoid the use of audio.
Introduce changes (new items) with slower, smooth transitions. Highlighting is an effective technique for showing relationships among data. Multiple, separate animations should be avoided. Indicate current and target locations if items move around the display.	Judicious use of animation is necessary for effective design.
Information grouped by time should be sequential. Information grouped by type can use random layouts within sub-groupings.	Layout should reflect the information according to its intended use.

Appropriate screen space should be delegated according to information importance.

Important areas should be clearly recognizable.

Employ highly recognizable metaphors that use/stress organizational layout

Avoid metaphors that suggest haphazard or random layouts.

Use cool colors (blues, greens) for borders and backgrounds.

Use warm colors (reds, yellows, oranges) for important information pieces and highlighting.

Avoid heavy use of bright colors.

Appropriate color schemes should be used for information understanding.

Use sans serif fonts, in large size to facilitate reading.

Text-based banner information should be located on the top or bottom, depending on focus and use of the information.

Avoid using multiple, small fonts.

Use text banners only when necessary.

Eliminate or hide configurability controls.

Eliminate or hide configurability controls.

Appendix G

Questionnaire

Here we have a snapshot of the questionnaire provided to the evaluators in the heuristic comparison study (Chapter 5).

Notification Collage

Using a collage metaphor

+ allows users to informally post information without any regards to organization

+ background supports the idea of graffiti, i.e. you put anything you want for everyone to see

+ lack of organization creates an informal virtual environment for users

+ scattered arrangements of artifacts across the screen reflects the collage aspect

BUT lack of organization can hinder efforts to find an artifact, frustrating users when they are looking for a specific piece of information

Please add any comments (about your ratings or the claim)

Strongly Disagree

Disagree

This claim is appro

1. Appropriate color

2. Layout should refl

3. Judicious use of a

4. Use text banners c

5. Show the presence

6. Using cyclic displ

7. Avoid the use of a

8. Eliminate or hide

Rate the severity th

No problem

Minor

Appendix H

Use Guide

Here we provide a guide for designers, developers, or researchers who are interested in creating heuristics for some system class. Our method supports heuristic creation and the following guide provides a simple explanation with examples at each step to illustrate the process.

Introduction

To understand the creation process described in Chapter 4, we provide a walkthrough of the process using an example system class based on the "secondary display" class of notification systems.

A secondary display focuses on supporting high levels of comprehension of information, supporting high levels of reaction, but also not requiring much attention. A classic example would be a stock ticker. The ticker exists on the screen in a small area, provides information of interest and use, yet does not take anything away from the primary task. The following sections describe a possible approach to creating a set of heuristics targeted to this system class.

Step 1 – Identify Target Systems

The first step in this process requires identification of example systems from the target class. These systems should be typcial of the system class in that they exhibit the desired levels of the critical parameters. Current literature, successful applications, and highly visible interfaces are good candidates for finding example systems.

Good examples of secondary displays would include the aforementioned stock ticker, email biffs, network monitors, and a plethora of others. Choosing three to five example systems should provide adequate coverage of the system class.

Step 2 – Create Scenarios and Claims

After picking the example target system, proceed to write scenarios and perform claims analysis on each system. This process is described in [77]. We have found that anywhere from three to six scenarios are adequate for capturing the typcial usage situations and tasks associated with a given

system. If a system is more robust, more scenarios are needed to cover the typical tasks supported by that system.

Extracting claims from the scenarios should be straigtforward, based on the techniques described in [77]. It is helpful when extracting the claims to keep in mind the user goals associated with the system, and how design choices (colors, animation, etc.) actually impact the user. This ties directly in with the next step.

Step 3 – Categorize and Classify the Claims

As mentioned in step 2, categorization and classifiation of claims is at the heart of this method. 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. For example, if a claim involves the use of animated text (like in a ticker), then a possible impact could be increased interruption. Another example from the same claim would be possible decreased comprehension (due to missed information, or only catching a snippet as it moved off screen). 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.

A useful technique to aid in this process is to create a problem tree that captures the claims and their associated impacts on the critical parameters. What this means is to create a physical depiction of the claims, along with their organization into the categories. This allows the researcher to focus on one group at a time during the next phase, thereby guiding and structuring the work effort.

Step 4 – Extract High Level Design Issues

After grouping the claims by category and assigning appropriate impacts on the critical parameters, the researcher can begin to extract design knowlegde from the problem tree. This process entails inspection of the wordings of the claims to identify the category and classification that led to its inclusion in that particular node of the tree. This wording, coupled with the wordings of the other claims in that node, can lead to one or more statements about the underlying design challenges captured in those claims. These statements about design are considered *high level* design issues and can be useful in their own right as design knowledge.

Determining the wording of these issues is somewhat open to the researchers invovled in the process. It is often useful to describe the type of design artifact and resulting impacts on critical parameters. There should be some amount of specificity to these issues as they are taken from a handful of individual claims. Deriving more generic heuristics occurs in the next step.

Step 5 – Synthesize Heuristics

Synthesizing a set of usable heuristics from the larger set of design issues requires investigation and analysis of the wordings and relationships among the various issues. Identifying similar or

common issues, as well as issues that deal with similar critical parameters suggests grouping and generalizing them 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. The new heuristics should apply to the targeted system class.

Vita

Jacob Somervell

Jacob was born on April 29, 1977 in Hopewell, Va to a chemical plant supervisor and his housewife. He had 3 older brothers and an older sister. Two years later his younger brother was born (six total). He moved from Hopewell, VA to Rich Valley, VA in the spring of 1983.

Jacob completed his undergraduate work at Clinch Valley College of the University of Virginia. He graduated Summa Cum Laude (3.9 or higher GPA) with a Bachelor of Science holding a double major in Mathematics and Computer Science on May 16, 1999. He started Graduate School at Virginia Tech on August 16, 1999. He received his Master's degree in Computer Science and Applications on May 16, 2001. He received his Doctorate in Summer of 2004. He has accepted a tenure-track faculty position with the University of Virginia's College at Wise, in Wise, VA.