Unpacking Critical Parameters for Interface Design: Evaluating Notification Systems with the IRC Framework

C. M. Chewar, D. Scott McCrickard
Center for HCI and Dept. of Computer Science
Virginia Polytechnic Institute and State University
Blacksburg, VA, USA
{cchewar, mccricks}@cs.vt.edu

Alistair G. Sutcliffe
The Centre for HCI Design, Department of Computation
Univ. of Manchester Institute of Science and Technology
Manchester, UK
a.g.sutcliffe@co.umist.ac.uk

Abstract
We elaborate a proposal for capturing, extending, and reusing design knowledge gleaned through usability testing. The proposal is specifically targeted to address interface design for notification systems, but its themes can be generalized to any constrained and well-defined genre of interactive system design. We reiterate arguments for and against using critical parameters to characterize user goals and usability artifacts. Responding to residual arguments, we suggest that clear advantages for research cohesion, design knowledge reuse, and HCI education are possible if several challenges are overcome. As a first step, we recommend a slight variation to the concept of a critical parameter, which would allow both abstract and concrete knowledge representation. With this concept, we demonstrate a feasible approach by introducing equations that elaborate and allow evolution of notification system critical parameters, which is made operational with a variety of usability evaluation instruments. A case study illustrates how one general instrument allowed system designs to be meaningfully compared and resulted in valuable inferences for interface reengineering. Broad implications and conclusions about this approach will be of interest to others concerned with using critical parameters in interface design, development of notification systems interfaces, or approaches to design rationale and knowledge reuse.

Categories & Subject Descriptors: H.1.2 [**Models and Principles**]: User/Machine Systems– *Human Factors.*

General Terms: Design, Experimentation, Human Factors, Measurement.

Keywords: Usability evaluation, peripheral display, design reuse, claims.

INTRODUCTION

Design of interactive systems that are typically used in multitasking or divided attention situations has become an increasingly important topic within human-computer interaction (HCI). Quite often, design challenges in areas such as ubiquitous computing, computer supported cooperative work, and information visualization are resolved with peripheral or ambient information display, multiple coordinated views, and secondary displays techniques. Recent workshops [3][5] and special issues of HCI journals [18] have characterized *notification systems* as interfaces that use these and other techniques to support information delivery during user multitasking. Theories to guide general design thinking have surfaced, such as Horvitz's principles for mixed-initiative notification systems [12] and the attention-utility theme [15]. To increase cohesiveness of research within this emerging design area, we have proposed "critical parameters" to capture user goals related to interruption, reaction, and comprehension (or *IRC*) as a potential solution [15][16][17] (revisited later). This paper elaborates this proposal, with a focus on recognizing and mitigating tradeoffs related to using critical parameters for capturing, extending, and reusing design knowledge gleaned through usability testing.

Newman introduced the concept of *critical parameters* for HCI as a mechanism to enable meaningful modeling and execution of usability evaluations that would allow systems to become progressively better [20]. These figures of merit, when defined and adopted, would help interface designers recognize the broader intentions of the technology, shifting focus away from interface-specific details to qualities that could be directly measured, compared to benchmarks, and reengineered to better serve a user's purpose. Critical parameters have three essential characteristics: their satisfaction is critical to the success of the system, they are persistent across successive systems, and must be manipulable by designers [21]. Newman presents arguments for adapting design practice with critical parameters, which others have extended as an approach for increasing cohesion and relevance within HCI research communities [28].

As these arguments are promising for, and perhaps most adoptable in a newly emerging design research area like notification systems, we have embraced them fully. In recent efforts, we presented an articulation of the notification systems design space, organized by the IRC critical parameters (see Figure 1) [16]. We have provided initial examples of system and design artifact classifications, as well as a demonstration of how IRC parameters could guide a walkthrough of a human information processing model [17]. Exploratory work probed the use of IRC parameters for indexing mechanisms to notification systems design knowledge repositories [23], and identified general challenges with using critical parameters in systems supporting design knowledge reuse [8]. We pursue a long-term vision of enabling integrated *claims reuse* in a software design process, a proposal advocated by Carroll and Sutcliffe as a means of expressing an artifact's psychological consequences (*claims*) in an explicit, accumulable, and generally reusable "designer-digestible packets of HCI knowledge" [6][7][24][25].

DIS2004, August 1–4, 2004, Cambridge, Massachusetts, USA.

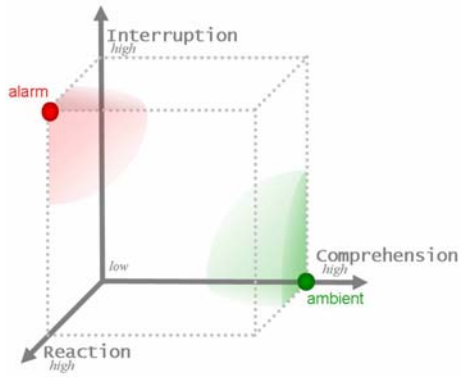

Figure 1. Notification systems design space, defined by IRC critical parameter possibilities. Two classes of systems are depicted, but other combinations suggest other classes [17].

Although we have made progress through understanding and articulating notification systems design challenges in terms of IRC parameters, we feel important counter arguments must be acknowledged and addressed. Our sincere hope is that the analysis and potential approaches we suggest will continue the dialog on methodological and practical aspects of using critical parameters in interactive systems design. We have intentionally developed our proposal to serve as an open, corrigible record of issues and possibilities, rather than a final solution.

CRITICAL PARAMETER ARGUMENTS

To present essential background on both notification systems design challenges and our approach to using critical parameters, we here introduce the key issues that have emerged. Many of these issues have been introduced by anonymous reviewers, workshop attendees, and HCI students reacting to our work. We strive to present all major argument tradeoffs that have come to our attention—mitigating the downside points provides a basis for our continuing proposal and much future work.

Creating Scenario Families

As we seek to define and establish critical parameters for a class of systems, it is important to explore the coverage of systems for various combinations of parameter values. As different systems will be used in similar ways, it is useful to have a mechanism for capturing the similarities.

Critical parameters support the organization of systems by *scenario families*, collections of systems and the context of their use grouped by critical parameter value. Including not only a description of the system but also a description of its use suggests meaningful critical parameters for a design class by shifting focus away from just the technology onto its use. This allows abstraction of the problem space and efficient focus on key design concerns.

However, the use of scenario families risks limiting novel thinking and innovation in the design of new systems, particularly those that use emerging technologies. It may be difficult to generalize lessons across platforms, information types, and other usage situation particulars. By their very nature, scenarios focus a reader on a very specific situation, and great care must be taken in constructing a scenario family to achieve appropriate coverage of the wide range of systems that should be included in it.

Forming a General Design Space

An important step in design and knowledge reuse is the categorization of systems in a domain. Scenario families exemplify key collections of systems, but a definitive design space should position all systems within the space, organizing all existing efforts as a body of examples. In so doing, the space allows recognition of research and innovation opportunity using common critical parameter values. While no design space can capture every possible concern that a designer or user might have, by locating all systems (and their use) within a general design space we subscribe to the belief that some knowledge is better than none at all—a developer can use the space to focus thoughts, guide decisions, and build on the work of predecessors.

However, the difficulty still arises in that we may not have a key, manageable set of critical parameters. It has proven difficult to define commonly used terms in a way agreeable to all even for a new domain like notification systems—for more mature disciplines, it may require an impetus that rarely occurs, such as a dynamic intellectual leader or a large and focused monetary commitment.

Even when a group of researchers agree on critical parameters, there is a need to be able to consistently quantify parameters on a scale. However, it may prove difficult to do so with parameters that are generally considered abstract or nonlinear, such as distraction or privacy. A tradeoff occurs when parameters must be unpacked to the point where the relationship between them is clear—terms are simplified and dependences removed, but the important broader concept can be obfuscated.

Expressing Problems

Designers often face a difficult task in addressing unfamiliar problems that arise in the design process. Expressing new design problems in terms of critical parameter values allows efficient association with theories and guidelines from psychology, sociology, and human factors—information that is otherwise difficult to obtain. Designers are, in effect, using critical parameters as an index into a vast store of knowledge.

However, this process again relies on agreement with and consistency of critical parameters. In their current form, designers must know, understand, and accept the critical parameters of a field to benefit from them. Also, one can argue that this process minimizes the skills of designers, who currently access this information intuitively. For such designers, the formalisms of critical parameters threaten to stifle creativity and waste time, and are therefore viewed as unnecessary overhead.

Assessment through Mediated Evaluation

Mediated evaluation builds a store of knowledge through the design process by creating goals early on, then augmenting or modifying them through the design process to keep work focused on the needs of the user and to understand where the value of the final product resides [7]. Assessment of critical parameter values through mediated evaluation can allow systems to be compared in formative phases with other systems, benchmarks, and standards. As the development process progresses, incremental improvements through *hill climbing* [6] can address the weaknesses of the developing system with respect to the parameters identified as most important, thus lending a systematic structure for knowledge accumulation and reuse.

However, mediated evaluation based on critical parameters relies on standard, unavailable assessment and classification techniques. In addition, the processes related to mediated evaluation are not yet well understood, and the standardized assessment techniques may be limited in generality by platform and usage situation particulars, requiring significant effort in the evaluation phase.

Designers want to evaluate interface features that are important to them, not ones that are important for the research community.

RESIDUAL ISSUES AND PROPOSAL

Having recognized these and other challenges in using critical parameters for design knowledge reuse, this section explores the key outstanding problems. While our approach is not intended to be a final solution, it should evolve thinking and be exemplary of what can be done in the field. Ongoing work in claims reuse suggests parameters for a development environment for the design of notification systems. A system based on this environment shows promise in providing teaching benefits for human computer interaction, and initial tests have demonstrated the ability to achieve consistency in requirements specification for user goals.

Problem Statement

Before critical parameters can be used in notification systems development to capture design knowledge from usability testing, at least two important issues must be resolved. First, it is unclear how an approach for classifying usability artifacts according to critical parameters would proceed. While it may be possible to put forth general artifact characteristics that merit certain ratings and assist classification efforts (i.e., "fast tickering rates have high interruption," or "audio cues provide low comprehension"), this approach would be mired in subjectivity or require an unwieldy set of platform-specific guidelines. Furthermore, it would close dialog that would be useful for conceptual evolution of the critical parameters, their definitions and scales, and measurement techniques. Therefore, a second important issue is determining how classification approaches can encourage critical parameter conceptual evolution.

Conjecture and Argument Structure

- Clearly, if it were possible to express notification system design challenges in terms that anyone could understand—and readily compare—we would gain many advantages. In order to achieve this, we propose that a critical parameter should have two parts (shown in Table 1):

- A sufficiently *abstract term* to allow meaningful generalization and express user goals and situational expectations, and

- *Concrete term(s)* for measurable and manageable psychological effects that can be directly observed or estimated for a given artifact.

Elaborating our previous idea of critical parameters in equation form demonstrates this conjecture, and provides resolution to many of the residual issues inherent in our approach.

Table 1. Proposed critical parameter components.

	Abstract term	Concrete term(s)
General purpose	• Summarizes a user goal • General psychological/ human information processing effect • Meaningful across situations and platforms	• Measurable with an instrument • Manageable through design changes • Characterizes a specific instance in a suitable context
Necessary for…	• Defining design spaces • Requirements engineering • Reusing designs • Comparing interfaces	• Testing artifacts • Explaining effects • Preserving context

Argument in support of this proposal proceeds in the following sequence. First, we show how equations unpack the current critical parameters and provide both abstract and concrete facilities for characterizing notification systems usability concerns. Component variables assist in defining abstract parameters, providing a means for generality and reuse, as well as measurability and manageability. Second, we illustrate how critical parameter equations provide a point of convergence for a variety of usability evaluation methods and assessment instruments. We demonstrate two possible methods (analytical and empirical through controlled lab testing), and provide a case study to detail evaluation results using the analytical instrument on three different notification system interfaces. Results suggest the utility of this approach based on critical parameters, and indicate that we are able to make progress toward using the approach with HCI education efforts. We speculate about other broad implications.

This argument addresses a few of the key concerns raised, but leaves other concerns for future work. In particular, future efforts must address generalizing claims to extend proposals by Carroll and Sutcliffe [6][7][24][25]. Focusing initial efforts toward structuring a design process for the benefit of HCI education diverts immediate need to address points related to designer overhead, but it is our hope that features built into an integrated development environment emerging from ongoing work will mitigate these arguments. Only time, broader dialog, and additional experience will increase or decrease our confidence in critical parameter selection.

PROVIDING ABSTRACT AND CONCRETE TERMS WITH EQUATIONS

In previous work, we have proposed three critical parameters to capture user notification goals related to interruption, reaction, and comprehension (IRC) [15][16][17]. As the design space in Figure 1 illustrates, systems can be thought of as having targeted (design model, as in [22]) and actual (user's model) values for each parameter. For example, a stock ticker notification system may be designed to target low interruption, low reaction, and high comprehension (the ambient class in Figure 1)—but actual system usage may display a complete inversion of these parameters (the alarm class). Understanding targeted goals and user performance characteristics in terms that are comparable to each other and other systems provides opportunity for many benefits, but abstract parameters must be associated with concrete terms that can be assessed in usability evaluations.

Three equations are introduced for notification systems interface evaluation, allowing conversion of measurable, manageable concrete variables (summarized in **Error! Reference source not found.**) to the abstract parameters that relate to general user goals and psychological effects. This is not intended to be a robust, integer-based system. Instead, the equations are intended as a conceptual metaphor, loosely organized as a categorical, interval scale approximation. When considering the validity of the equations, one should think of them as numeric representations of low, somewhat low, moderate, somewhat high, and high parameter categories. The equations are thought to assist in obtaining more consistent selection of these concrete categories while assigning abstract user's model parameter values. Numeric representations are useful in facilitating search/indexing operations. The case study presents an initial testing of this hypothesis.

Table 2a. Concrete terms used in the interruption (I) equation, and usability evaluation assessment techniques for each.

Concrete Term		Assessment Technique	
Symbol	**Description**	**Analytical/Subjective**	**Empirical/Objective**
COI	cost of interruption	*Given the nature and importance of the user's primary task at the receipt of the notification, how **costly** would an interruption be?* {extremely = 1; very = .75; moderately = .5; not very = .25; not at all = 0}	Interruption Workbench [13] output; P(High) is weighed at 1, P(Med) = .5, P(Low)=0
S	primary task sustainment	*Compared to the primary task performance before the notification delivery, how much does the primary task performance **reduce** when the notification is present?* {not at all = 1; less than half = .75; about half = .5; more than half = .25; completely stops = 0}	Ptask performance while multitasking divided by ptask performance as a solo-task [26]

Interruption

The first critical parameter we have identified for notification systems design is *interruption*. There have certainly been many important branches of work in cognitive and experimental psychology to understand the facets of interruption, and recent efforts within the HCI research community have helped deliver findings to system designers and evaluators [2][9][13][19]. Seeking to improve this transfer of research findings, we offer a simplified model of interruption suitable for design and evaluation of notification systems:

$$\mathbf{I} = 1 - s^{3 \cdot COI}$$

where $\boldsymbol{s}$ = sustainment
$\boldsymbol{COI}$ = cost of interruption

In this conception, interruption (I) can be described as the effect of reallocating attention from the primary task to the notification. "I" describes both the appropriateness of an interruption, as well as the actual interruptive effect of the notification artifact (distraction to the primary task). Therefore, "low I" can describe either an artifact that supports attention grading/parallel processing during the performance of an urgent primary task (high sustainment, regardless of COI) or any quality of multitasking performance in a non urgent situation (low COI, regardless of sustainment).

Appropriateness of an interruption is represented by COI (cost of interruption), characterizing the user's willingness to accept an interruption, and thus the urgency of the primary task can be inferred. As established by Horvitz's Interruption Workbench [13], COI describes a total task situation in terms of how much a given user would typically pay in dollars not to be interrupted. The Interruption Workbench records a variety of situation characteristics, such as the specific primary task application, level of ambient noise, recent keystroke and mouse activity, etc) over an extended period of normal user activity. The tool segments the observations into periods in which the task variable combinations are consistent. Users rate each segment, assigning the dollar value they would pay to avoid interruption, allowing Bayesian inference networks to aggregate samples and determine probability distributions for various costs of interruption levels. Alternately, this value can be estimated based on existing empirically determined examples (Table 2a provides a summary).

Actual interruptive effect can be gauged by primary task sustainment—a metric used to quantify the change in the primary task performance from solo-task to dual-task performance. Calculation of primary task sustainment has been demonstrated for notification interfaces [26] and broader psychology efforts [29].

The equation we present is modeled with an exponential COI to reinforce the importance of this factor, but tripled to ensure a fairly wide range of I-values for a given COI and to produce a moderately high I-value (0.65) when both s and COI equal 0.5.

Reaction

The second abstract critical parameter term for notification systems, *reaction*, describes a user goal that can be generalized as an immediate response to a new notification.

$$\mathbf{R} = \frac{(t \cdot h)^{\frac{1}{3 \cdot COI}}}{2} + \frac{h(0.5 + COI)}{3}$$

where $\boldsymbol{t}$ = relative response time
$\boldsymbol{h}$ = hit rate

The reaction (R) equation consists of two parts, each worth up to an R-value component of 0.5. The first term takes two reaction performance metrics—hit rate (h) and relative response time (t)—and lowers the average according to strength of COI. *Hit rate* refers to the concept from signal detection theory [11] where a user correctly detects and responds to a signal (a notification). Relative response time is a ratio between actual and expected response times (see Table 2b for assessment techniques). Certainly, expected response times may be dependant on usage context and information characteristics, and they should be estimated or obtained in requirements gathering. The second term of the equation can add up to half the hit rate to the R-value, depending on the strength of COI. Moderate reaction (R=0.5) is scored when two-thirds of the hit rate and reaction time is achieved with a COI of 0.5. Moderate or high R-values are always obtained when one of the variables is near maximum and the others are at least moderate.

The equation is also designed so that no more than R=0.5 can be achieved if one of the three variables equals zero. In order to understand this rationale, one must consider that R is a characterization of an artifact's effectiveness for supporting

Table 2b. Concrete terms used in the reaction (R) equation, and usability evaluation assessment techniques for each.

Concrete Term		Assessment Technique	
Symbol	**Description**	**Analytical/Subjective**	**Empirical/Objective**
H	hit rate	*How often will users actually notice important changes in the notification, as opposed to not noticing them?* {always = 1; more than half = .75; about half = .5; less than half = .25; never = .0001}	As in Signal Detection Theory, P(H) divided by total signals [8],[28]
T	response time	*In cases where a notification suggests an action for a user to take, how does the user's response time compare to the reasonably desired response time?* {better or as good as expected = 1; slightly slower = .75; about twice as slow as expected = .5; much slower = .25; extremely slow or action never taken; no action ever required = 0}	Determine *actual response time* (*a*) as the difference between signal presentation and signal response; *expected response time* (*e*) provided in system specification; $t = e / a$, when $a > e$ (otherwise $t = 1$)

reaction in a dual-task situation. That is, if the notification system is not attempting to resolve a situation constrained by the tradeoff of limited attention for gain in utility (the attention-utility theme [17]), in which there would generally be at least a moderate value for COI, then the appeal of the artifact for facilitating notification reaction in a dual-task situation is inherently limited and therefore penalized. Both aspects of the reaction performance are also critical—a near-perfect hit rate would not be looked at as effective reaction if the response time were significantly slower than specification. Likewise, an acceptable response time has limited worth in the case that most signals delivered by the notification system are missed. Another feature of the equation is the prominence of the hit rate. Factoring this variable directly into both terms allows quick growth of R-values as hit rate increases, especially when COI is greater than 0.5. This adds a strong characteristic to R of being a measure of response selection probability.

Comprehension

Our abstract parameter of comprehension is based on the concept of situation awareness, in which a user accumulates Perception (of the elements in the system), Comprehension (of the current situation), and then Projection (of future status). Each level is dependent on achieving some part of the preceding level, and represents a progressively higher state of situated awareness [10]. Thinking of notification comprehension as situation awareness brings our efforts in characterizing notification systems in line with a wealth of research in the human factors field, and reinforces our argument that each parameter is a separable dimension. For instance, studies have shown that we can recognize the characteristics of awareness independent of the processes required to maintain it (working and long term memory or attentional state) [1] or the response selections that result from it [28]. Thus, the comprehension critical parameter describes longer-term (not immediate) knowledge gain.

$$\mathbf{C} = f + \frac{(1-f)(p + 2c - cp)}{3}$$

where $\boldsymbol{p}$ = perception rate
$\boldsymbol{c}$ = base comprehension
$\boldsymbol{f}$ = projection (future)

The simplified equation that appears above is difficult to explain, so we revert to the unsimplified version:

$$\mathbf{C} = \frac{p + (1-p)(c + f(1-c))}{3} + \frac{c + f(1-c)}{3} + \frac{f}{3}$$

This equation consists of three terms—one for each level of situation awareness. As each level is maximized, the equation ensures that C=0.33, 0.67, and 1 respectively. If a given level is not maximized, achievements in the higher levels provide credit toward the C-value (see Table 2c for a summary).

Still under review is the issue of whether COI should be an additional factor in the C equation. Some justification for this is present in Endsley's argument that temporal dynamics play an important part in assessing the comprehension and projection levels. Specifically, she mentions that part of projection requires an understanding of the rate at which information is changing. However, by articulating the concrete terms we rely on to form our abstract notion of notification systems comprehension, we open this issue and others for debate within the research community.

Intended Use and Evolution

As stated previously, we present these equations as a conceptual metaphor to connect concrete critical parameter terms with abstract terms that can be generalized to understand design spaces, facilitate requirements engineering, support design knowledge reuse, and compare interfaces within a common design domain. Each variable on the right side of an equation is a concrete term that can be measured in requirements gathering and usability testing with a wide variety of methods, as we demonstrate in the next two sections.

Abstract and concrete terms for critical parameters like these can be introduced for any other class of interactive system to describe user goals and psychological effects of the interface. We hope that our community of researchers will work to evolve these conceptions, adapt them to their own needs, and ultimately improve consensus. Thinking of these terms as "slots" to guide discussion within the research community, we see an important opportunity for mechanisms that elaborate and validate relationships between variables, as well as research that demonstrates extensible, context-specific assessment methods for obtaining concrete variable values.

Table 2c. Concrete terms used in the comprehension (C) equation, and usability evaluation assessment techniques for each.

Concrete Term		Assessment Technique	
Symbol	**Description**	**Analytical/Subjective**	**Empirical/Objective**
P	perception rate	*When considering the total number of times a user interacts with the notification system, what is the **ratio** of the interactions in response to an important notification vs. total interactions (including those when no actual notification was being delivered, i.e., user checking on their own or thinking there was a notification)?* {1 to 1 = 1; 2 to 3= .75; 1 to 2 = .5; 1 to 4 = .25; more than 1 to 4 = 0}	As in Signal Detection Theory, P(H) divided by total responses [8],[28]
C	base comprehension	*How much of the notification content will the user want to remember and be able to remember several minutes after the notification is delivered?* {all content = 1; more than half = .75; about half = .5; less than half = .25; none at all = 0}	Quiz user on a sample of notification content questions to assess correct interpretation, relationship to goals, and storage in long term memory. Use % correct.
F	projection	*Based on the notification content, how successful will the user be in making projections or predictions about future trends or the long-term state of the system being monitored?* {extremely successful = 1; very successful = .75; somewhat successful = .5; not very successful = .25; not a goal for this system = 0}	Quiz user based on a sample of interpretations that can be projected to predict future states or notification patterns. Use % correct.

OBTAINING VARIABLES IN USABILITY EVALUATIONS

If usability evaluation activities were focused on assessing concrete critical parameter terms to yield abstract characterizations, equations like the ones we introduced would provide a point of convergence for a variety of usability evaluation methods and assessment instruments. We certainly feel that a variety of methods and instruments (along with an evaluator's indispensable expert judgment) will always be necessary for the wide ranging and continuously evolving facets typical to usage settings and interface platforms. To clarify, we discuss two possible methods (analytical and empirical through controlled lab testing) for obtaining the concrete terms in our equations.

An Empirical Method

Since the equations are intended to characterize the user's model of the notification system interface, many would argue that data obtained from a user's actual usage experience is of primary value. System event logging, user observation, and user surveys can be tailored to collect data for each of the seven metrics. COI can either be collected by a tool like Horvitz's Interruption Workbench [13] or a survey method with less overhead. Notes for empirically obtaining each of the variables are summarized in **Error! Reference source not found.**. In ongoing work, we are experimenting with an automated notification systems testing platform that allows user event logging of critical actions, such as performance on a primary task with and without the notification system, response accuracy and timeliness to notification signals, and comprehension of important notification information after an extended period of time. Notification systems researchers have used similar testing platforms [2][4][9][15][19], and we are encouraged that data necessary for obtaining the critical parameter terms is often collected by most researchers, implying that existing experimental platforms could be easily modified.

From our reflection on empirical test instruments that help obtain the concrete parameter terms, we also note several points of caution. Since the test protocol relies on a definition of total number of signals present, evaluators should ensure users are only expected to respond to a realistic number of important notifications. This consideration may become important because analysis of signal detection performance may require that system interfaces are tested and compared based on a known, cached set of notification data to allow signal introduction times to be recognized, observed, and automatically processed by a testing platform. Alternately, user performance with actual, real-time data can be measured using screen recording or videotaping of a usability test session.

A final aspect to note about empirically assessing the concrete variables relates to the comprehension and projection terms in the C equation. We suggest data for these variables be collected in post-test surveys that probe recollection of key events, information states, and notification patterns. Alternately (and less desirably), popup windows or brief halts of the interface usage experience by the evaluator can allow comprehension-related questions to be asked throughout the test. A response mechanism that discourages participant guessing or uncertainty, such as open-ended questions or fill-ins, is particularly critical for obtaining these terms.

An Analytical Method

While empirical data may be preferable for characterizing the user's model of an interface design, empirical testing often comes at a much higher cost. To support user lab testing or field studies,

systems must be fairly robust and further along in the design cycle (implying higher cost for large changes and sometimes preventing formative testing). Other drawbacks include overhead involved with system logging or session observation and recording, participant recruitment, lab access, and other factors. For these reasons, and to facilitate formative and mediated usability evaluation, we were eager to develop an analytical testing method that could yield terms for the concrete critical parameter values.

As Table 2 shows, we were able to formulate a survey question and appropriate set of responses to analytically assess each concrete variable present in the equations. Just as with other analytical evaluation methods, we do not intend that a survey composed of these questions be used to collect opinions of general users. Rather, this instrument should be used by interface experts or at least experienced notification systems designers familiar with applicable challenges. While response selections provide feedback in the form of critical parameter values, perhaps of equal or greater value are the specific comments and rationale behind each rating, which can be expressed as claim upsides and downsides. We envision this analytical instrument to be used in a moderated evaluator discussion session that may or may not include the system designer, although each evaluator would provide individual assessments of each question.

The case study presented in the next section was conducted with the analytical instrument. In the case discussion, we provide additional details about the method execution and results analysis, as well as observations related to variations in session moderation techniques. We are generally pleased with the evaluation outcomes provided by this method, and recommend it as a tool for evaluating notification systems, allowing data necessary for the equations to be obtained.

CASE STUDY

We challenged a group of novice designers to improve upon a notification system interface developed by a Microsoft Research group [27]. The Scope (shown in Figure 2) is a small display that resides in the corner of a user's desktop, depicting new and existing notifications in quadrants for email, calendar, task, and alert items. As a circular-shaped interface, the Scope leverages a radar metaphor to convey relative item urgency. In their research, the original design group noted several usability concerns, so we instructed the new teams (15 total) to improve upon these and other issues they discovered through their own requirements gathering efforts. The three-month redesign effort was controlled through class specifications that required a mediated approach to advancing design rationale and making interface improvements.

Motivated by their requirements gathering results rather than any instructions, several of the teams came up with very different display and interaction strategies for the Scope redesign, abandoning the radar metaphor. We wanted to compare redesign options according to impact on notification critical parameters, visualizing each system within our design space. Other objectives of our study were to assess the difference between design model and user's model critical parameters for each system. We hoped that quantifying the conceptual models would help to expose interface features that should be redesigned in subsequent versions, suggest additional requirements gathering steps needed, as well as classify design artifacts for reuse. Note that these objectives are functions of the abstract critical parameter terms, as summarized in Table 1. We hypothesized that our analytical testing tool would be able to test all system designs so they could be meaningfully compared—highlighting differences between systems and between product and designer's intention.

Interfaces

We selected three interface redesigns that exhibited strong differences from the original Scope concept (shown in Figure 2). Although implementations were only in early, unpolished prototype form, we felt that each represented distinct notification strategies that would occupy different portions of the IRC design space (see Figure 1). Like many desktop notification systems and the Scope, the prototypes sought to convert a small portion of screenspace into a glanceable information center for notification awareness. Tooltips often provide brief summaries of notification content, with further details accessible through a mouse click. *Prototype A* was inspired by a bulletin board, introducing notifications as small notes that appear in rows according to category. *Prototype B* is a vertical bar for the side of a desktop that embodies a waterfall metaphor—notification icons fall slowly down the interface as they near their due date and unscheduled items are pooled at the top. Prototype C represents an iconic task list divided into several categories by notification type, which users can reorder and code by urgency. If our usability evaluation goals were met, we would help designers realize inaccurate information and interaction design assumptions and quantify the different psychological effects each option would have on users.

Testing and Analysis Procedure

The first step in our testing procedure was to collect design model intentions in the form of targeted IRC values from each system's design team. This was accomplished with a survey tool that has been validated to produce accurate and consistent design model IRC values [8]. After the designers answer general questions about the dual-task situation requirements assumed for the design constraints, the tool calculates the targeted IRC values.

The second step involved presenting the interface prototypes for analytical evaluation. We recruited 34 experienced notification systems designers to serve as evaluators. Between three and six evaluators were organized into sessions in which one interface was analyzed with the analytic instrument. Although each evaluator provided individual ratings and feedback, sessions were moderated

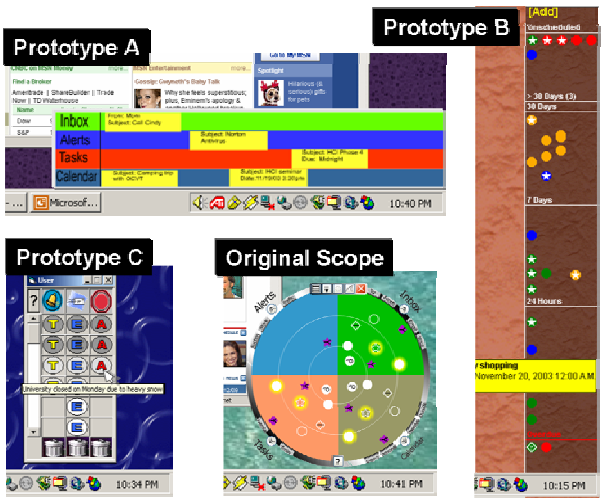

Figure 2. Notification systems interfaces studied in the case study usability evaluation. The three prototypes are redesigns of the original Scope interface, found in [27].

to prompt interactive discussion among evaluators about design decisions. This technique was used to ensure that evaluators were engaged in the process and thoroughly informed about the interface features. All prototypes were sufficiently interactive to demonstrate intended behavior. One session was conducted with a system designer as an assistant moderator—the designer explained intentions and answered evaluator questions related to specific features. However, the evaluator results obtained from this session were no more or less consistent with each other than in all other sessions, implying a negligible effect. All sessions lasted 20-35 minutes. Prototypes B and C were each analyzed by 11 evaluators, and 12 evaluators analyzed Prototype A.

The third step was the data analysis. Evaluator responses to the multiple choice questions were entered into a tool that associated responses with values for the concrete equation terms (included in Table 2), executed the three equations, and returned the user's model IRC values for each evaluator. IRC values for each system were averaged by parameter, and individual response differences from the system average were compared to screen outliers. Two of the Prototype A evaluators and one of the Prototype B evaluators exceeded the threshold ($\sigma = 1.5$), so their IRC values were removed from further analysis.

Next, we checked each system's collection of IRC values to determine whether all evaluators should be grouped together when making inferences, or whether clusters of evaluators should be established. To guide this process, we looked for the same expected rating consistency that is reliably achieved in the design model IRC assessment tool, $^{+}/_{-}$ 0.15 per parameter. Differences between Prototype B evaluators for all three parameters were smaller than this threshold ($I_{diff} = 0.02$, $R_{diff} = 0.11$, $C_{diff} = 0.13$), so all evaluator results were averaged together for inferences about user's model critical parameter values. However, both Prototype A and C had one parameter each that exhibited higher average difference between evaluators. While evaluators of Prototype A were consistent about interruption and reaction ratings ($I_{diff} = 0.10$, $R_{diff} = 0.13$), they differed on comprehension ratings ($C_{diff} = 0.22$). Therefore, evaluator responses were clustered into two groups: those that assessed high and low levels of comprehension. Prototype C evaluators differed on opinions about interruption ($I_{diff} = 0.25$), and the same clustering approach was used. Average differences between evaluators in new clusters for all three parameters fell within threshold consistency.

Finally, we wanted to determine whether the analysis instrument provided significantly more consistent IRC ratings with evaluators assessing the same system, when compared to all evaluators regardless of system. To determine this, we pooled each evaluator's parameter differences from their system's I, R, and C averages and compared that to each evaluator's differences from the overall I, R, and C averages established by all 31 evaluations. We observed a significant difference in support of our hypothesis—the instrument helps evaluators achieve consistency that is meaningful according to system ($F(1, 190) = 3.64$, $p < 0.01$).

Study Results and Implications

Confident that our instrument is sensitive enough to produce evaluator results expressing system nuances, we used the IRC averages (depicted in Figure 3) to make inferences about usability issues and possible redesign directions.

Prototype A

Having collected consistent design model IRCs from the system designers, we recognize that this system was intended to support moderately low interruption (I = .39, on a scale of 0 to 1), moderate reaction (R = .46), and moderately high comprehension (C = .61), which would be an ambient notification system with higher than usual interactivity. Unfortunately, one cluster of evaluators (labeled "UM-1" in Figure 3) thought that both interruption and reaction would be moderately low (I = .36, R= .35) and comprehension would be very low (C = .18). However, the user's model ratings by second cluster agreed much more closely with the design model: I = .35, R = .54, C = .62, implying that the design may meet intentions for some users.

Mitigating the concerns expressed by evaluators in the first cluster would be an important next step for these designers. Background and demographic differences could be studied further to identify distinctions between evaluator groups. Stated comprehension concerns could also be immediately addressed with more sophisticated visualization techniques. For example, one concern involved missing new notifications entirely due to clutter, overlap, and poor scalability—a problem that might be solved with a fisheye technique. Another issue raised was a user's inability to ascertain relative urgency of notifications—a feature apparent in the original Scope that enhances reaction.

Prototype B

Design model IRCs for Prototype B collected from these designers were much less consistent than normal, but averaged out to moderately high values for interruption and comprehension (I = .61, C = .63) and moderate reaction values (R = .57). To probe the inconsistency, we conducted interviews with the two primary designers, which revealed strongly opposed views for the goals of the system. One designer thought a tool that supported very high comprehension and long-term planning would be best, while the other wanted an alarm-like system that would be used to process urgent notifications and forget about long-term action items. While each designer thought they had compromised their goals somewhat, the first designer's model carried through to interface implementation and the user's model IRC. Evaluators consistently rated this system to be an ambient system, with moderately low interruption and reaction (I = .26, R = .27) and moderately high comprehension (C = .66). As expected, both designers were not satisfied with the evaluation result. In this case, the critical parameter models reveal a need for re-negotiation of the requirement assumptions for the basic user goals. This process can be assisted by discussing specific points on the design model survey. However, the system as it is provides a strong artifact example of an ambient user's model.

Prototype C

The design model for the final interface consistently targeted moderate interruption (I = .48) and moderately high reaction and comprehension (R = .71, C = .67). According to both clusters of evaluators, the designers missed their intention. Both clusters agreed that reaction would be moderately low (R = .24 and .21), a major difference from the design model that would be essential to correct. Evaluators were concerned that new notifications would be detected too slowly, since user memory overhead would be too high without any glanceable notification context and the interface's scrolling mechanism would be problematic. One cluster saw these problems as a basis for moderately high interruption (I = .79), while the other cluster felt the interface would simply be ignored and introduce interruption less than

intended (I = .29). Both clusters thought only moderate comprehension gains would be supported by this interface (C = .57 and .45). Faced with these large disparities, the design team may be wise to consider an alternate approach.

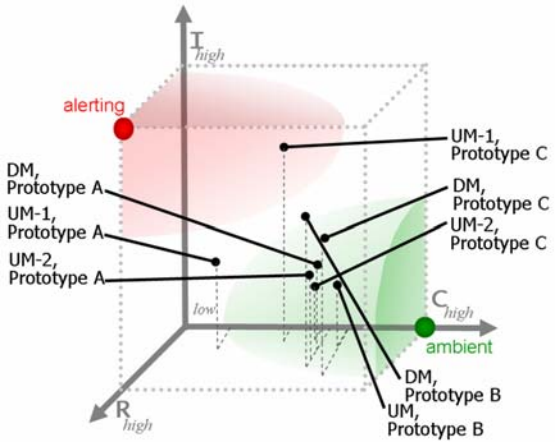

Figure 3. Design model (DM) and user's model (UM) assessments for the prototypes evaluated in the case study.

Broader Implications – Comparison and Reuse

While the IRC parameters were useful in assessing each design individually, the broader benefits of using critical parameters are recognized in activities such as system comparison and design knowledge reuse. For example, if we are looking for a more ambient redesign of the Scope, Prototype B would be the best starting point. However, techniques used in Prototype A may offer relevant inspiration, and it may be wise to conduct an evaluation on the Scope to see whether real critical parameter improvements are even being proposed. As information and interaction design changes are made to any system, a series of IRC evaluations can show progress between versions, as well as specific effects of feature-level artifacts. These psychological effects can be recorded as claims [7], indexed by IRC values [23], and archived in a library for design knowledge reuse [25]. For designers that are faced with brainstorming notification options that match a particular design model (perhaps like the designers of Prototype C), such a library may be an indispensable resource.

CONCLUSIONS

This work provides another step toward a long-term proposal for integrating critical parameters, mediated evaluation, and claims reuse in interactive system design and evaluation activities. Our sincere hope is that the analysis and potential approaches we suggest will continue the dialog on methodological and practical aspects applicable to notification systems. Though rooted in the study of notification systems, we feel that our research approach can generalize to other classes of systems.

We have mentioned directions for future work throughout our proposal. Specific contributions of this work are:

- Summary of arguments for and against using critical parameters to characterize user goals and usability artifacts,
- Variation to the concept of a critical parameter, which would allow benefits related to both abstract and concrete knowledge representation (see Table 2),
- Equations and usability evaluation support to elaborate and allow evolution of notification system critical parameters,
- A case study illustration of how a general (analytical) instrument allowed meaningful comparison of system designs and resulted in valuable inferences for reengineering.

Our proposal makes explicit many aspects of design that researchers are sometimes uncomfortable with. For instance, the notion of setting a user goal and psychological effect like reaction to a linear axis often evokes resistance. However, we suggest that the notion can be embraced as a conceptual metaphor and tool for dialog. We believe that extending the idea of critical parameters [20] and conceptual models [22] from original intentions may inspire improved methods for HCI research.

Primary benefits of this approach may be found in educating students of HCI about design tradeoffs and mediated evaluation. Concepts articulated by equations, tools, and visualizations improve the chance that students will be intrigued by HCI problems. We are also hopeful that a critical parameter approach to interactive design research dialog can improve consensus of key issues, comparison of new efforts to existing efforts, and development of context-specific usability testing methods and instruments. As the community looks for approaches that will increase the likelihood of science of design, or support the practice of usability engineering, these arguments should be of interest, broadening as a topic of continued debate.

ACKNOWLEDGEMENTS

The interface prototypes in our case study were designed by Josh Adell, John Archie, Edwin Bachetti, Niteesh Bharara, Andrew Jackson, Joey Jezioro, Abijeet Jhala, Aaron Kaluszka, Tim Fuller, Theresa Klunk, Jed Lake, and Vinay Lakahani. Special thanks to Edwin Bachetti for assisting the usability evaluation.

REFERENCES

[1] M. J. Adams, Y. K. Tenney, & R. W. Pew. Situation Awareness and the Cognitive Management of Complex Systems, *Human Factors 37*: 87-104, 1995.

[2] B. P. Bailey, J. A. Konstan, & John V. Carlis. The effects of interruptions on task performance, annoyance, and anxiety in the user interface. In *Proceedings of the IFIP TC.13 International Conference on Human-Computer Interaction (INTERACT 2001)*, 593–601, 2001.

[3] L. Bartram & M. Czerwinski. Workshop 9: Design and evaluation of notification interfaces for ubiquitous computing. *Fourth International Conference on Ubiquitous Computing (UbiComp)*, 2002.

[4] L. Bartram, C. Ware, & T. Calvert. Moving icons: Detection and distraction. In *Proceedings of the IFIP TC.13 International Conference on Human-Computer Interaction (INTERACT 2001)*, 157-165, 2001.

[5] J. Cadiz, M. Czerwinski, D. S. McCrickard, & J. Stasko. Workshop: Providing elegant peripheral awareness. In *Conference Extended Abstracts on Human Factors in Computer Systems (CHI '03)*, 1066-1067, 2003.

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

[7] J. M. Carroll, M. K. Singley, M. B. Rosson. Integrating theory development with design evaluation. *Behavior and Information Technology 11*: 247-255, 1992.

[8] C. M. Chewar, E. Bachetti, D. S. McCrickard, J. E. Booker. Automating a design reuse facility with critical parameters: Lessons learned in developing the LINK-UP systems. In *Proceedings of the Conference on Computer-Aided Design of User Interfaces (CADUI)*, January, 2004.

[9] E. Cutrell, M. Czerwinski, & E. Horvitz. Notification, disruption, and memory: Effects of messaging interruptions on memory and performance. In *Proceedings of the IFIP TC.13 International Conference on Human-Computer Interaction (INTERACT 2001)*, 263–269, 2001.

[10] M. R. Endsley, B. Bolte, & D. G. Jones. *Designing for Situation Awareness: An Approach to User-Centered Design*. Taylor & Francis, New York, 2003.

[11] D. M. Green & J. A. Swets. *Signal Detection Theory and Psychophysics*. Wiley, New York, 1966.

[12] E. Horvitz. Principles of mixed-initiative user interfaces. In *Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI '99)*, 159–166, May 1999.

[13] E. Horvitz & J. Apacible. Learning and reasoning about interruption. In *Proceedings of the Fifth International Conference on Multimodal Interfaces*, November 2003, Vancouver, BC, Canada.

[14] E. Horvitz, C. Kadie, T. Paek, & D. Hovel. Models of attention in computing and communication: from principles to applications. *Comm. of the ACM*, 46(3):52–59, 2003.

[15] D. S. McCrickard, R. Catrambone, C. M. Chewar, and J. T. Stasko. Establishing tradeoffs that leverage attention for utility: Empirically evaluating information display in notification systems. *International Journal of Human-Computer Studies 8*(5): 547-582, May, 2003.

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

[17] D. S. McCrickard, C. M. Chewar, J. P. Somervell, & A. Ndiwalana. A model for notification systems evaluation—Assessing user goals for multitasking activity. *ACM Transactions on Computer-Human Interaction (TOCHI) 10*(4): 312-338, December 2003.

[18] D. S. McCrickard, M. Czerwinski, & L. Bartram. Introduction: Design and evaluation of notification user interfaces. *International Journal of Human-Computer Studies 8*(5): 509-514, May 2003.

[19] D. C. McFarlane. Comparison of four primary methods for coordinating the interruption of people in human-computer interaction. *Human Computer Interaction*, 17(3), 2002.

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

[21] W. M. Newman, A. S. Taylor, C. R. Dance, & S. A. Taylor. Performance targets, models and innovation in interactive systems design. In *Proceedings of the Conference on Designing Interactive System (DIS '00)*, 381–387. 2000.

[22] D. A. Norman. Cognitive engineering. In Donald A. Norman and Stephen W. Draper, editors, *User Centered System Design: New Perspectives on Human Computer Interaction*, pp. 31–62. Lawrence Erlbaum Associates, 1986.

[23] C. Payne, C. F. Allgood, C. M. Chewar, C. Holbrook, & D. S. McCrickard. Generalizing interface design knowledge: Lessons learned from developing a claims library. *2003 IEEE International Conference on Information Reuse and Integration (IRI 03)*, October 27-28, 2003.

[24] A. Sutcliffe. *The Domain Theory: Patterns for Knowledge and Software Reuse*. Lawrence Erlbaum Associates, Inc., Mahwah, NJ, 2002.

[25] A. Sutcliffe & J. M. Carroll. Designing claims for reuse in interactive systems design. *International Journal of Human-Computer Studies 50*: 213-241, 1999.

[26] D. Tessendorf, C. M. Chewar, A. Ndiwalana, J. Pryor, D. S. McCrickard, & C. North. An ordering of secondary task display attributes. In *Extended Abstracts on Human Factors in Computing Systems (CHI '02)*, 600–601, 2002.

[27] M. van Dantzich, D. Robbins, E. Horvitz, & M. Czerwinski. Scope: Providing awareness of multiple notifications at a glance. In *Proceedings of the 6th International Working Conference on Advanced Visual Interfaces (AVI '02)*, 2002.

[28] S. Whittaker, L. Terveen, & B. A. Nardi. Let's stop pushing the envelope and start addressing it: A reference task agenda for HCI. *Human-Computer Interaction*, 15:75–106, 2000.

[29] C. D. Wickens & J. G. Hollands. *Engineering Psychology and Human Performance*. Prentice Hall, Upper Saddle River, NJ, third edition, 2000