Creating an Interactive Learning Environment with Reusable HCI Knowledge

Alain Fabian

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

Master of Science
in
Computer Science and Applications

D. Scott McCrickard, Chair
Chris North, Member
Andrea Kavanaugh, Member

May 25, 2006
Blacksburg, Virginia

Keywords: interactive learning environment, information gathering, design knowledge, reuse, interruption, claims, notification systems, human-computer interaction.

Creating an Interactive Learning Environment with Reusable HCI Knowledge

Alain Fabian

Abstract

This thesis proposes creating an interactive learning environment for Human Computer Interaction (HCI) to facilitate access to, and learning of, important design knowledge. By encapsulating HCI knowledge into reusable claims stored in a knowledge repository, or claims library, this learning environment aims at allowing students to effectively explore design features to limit their reliance on intuition to mold their interfaces, help them address proper design concerns, and evaluate alternatives for their designs. This learning approach is based on active learning where students create their own knowledge by gathering information. However, building adequate development records from which students can gather HCI knowledge is critical to support this approach. This thesis explores using effective reusable design components to act as design records to create an interactive learning environment for students learning HCI design.

An initial prototype for the learning environment introduces claims as an encapsulation mechanism for design features from which students can gather HCI knowledge. Pilot testing outlines the accessibility, applicability and reusability problems associated with this approach. To solve theses issues, a taxonomic organization of an improved form of claims (reference claims), is introduced to share core design knowledge among students. A taxonomy is designed as a way to expose students to important design concerns as well as a method to categorize claims. Reference claims are introduced as improved claims inspired by reference tasks to expose students to design alternatives for design concerns. A detailed taxonomy and a set of reference claims for the domain of notification systems demonstrate how existing theories of design can be translated into reference claims to create an interactive learning environment. An experiment illustrates the applicability and reusability of reference claims for various designs within a particular domain. Finally, an evaluation assesses the benefits of this learning environment based on reference claims in terms of improving student designs and increasing the amount of HCI knowledge they reuse. Results show that by exposing students to valuable concerns and alternatives for the design of interactive systems, an interactive learning environment based on reference claims can improve students' understanding of the design scope and lead to an increased use of existing HCI knowledge in their designs.

Acknowledgements

First and foremost, I would like to thank my family and my girlfriend Jean, for their encouragements over these past 2 years. None of this would have been possible without you. I would also like to thank Dr. McCrickard for encouraging me to enter the graduate program and for his guidance in helping me complete this work. I would also like to thank Shahtab Wahid, Saurabh Bhatia and Andrew Sabri for their contributions to this work, Cyril for his support, and the members of my committee, Dr. Kavanaugh and Dr. North.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Human-Computer Interaction (HCI) scientists have conducted much research over the years to improve how designs are created. Much of this research exists in many forms such as theories, guidelines, or observational studies that are taught to students learning interface design. However, students are often lost in this collection of design research and can not easily apply it to their own designs. This thesis proposes creating an interactive learning environment for HCI to facilitate access to, and learning of, important design knowledge. By encapsulating this design knowledge into reusable *claims* stored in a readily accessible repository—the *claims library*—this learning environment has the potential to limit students' reliance on intuition to mold their interfaces, help them address proper design concerns and evaluate alternatives for their designs as well as increase their use of HCI knowledge. This learning approach based on active learning, where students create their own knowledge by gathering information as opposed to passive learning where students absorb knowledge from books and lectures, can improve the learning of HCI design. However, building adequate design records from which students can gather HCI knowledge is critical to support this approach. This thesis explores using reusable design components such as *claims* to act as development records to create an interactive learning environment for students learning HCI design.

1.1 Learning Interface Design

Students are often required to apply various HCI guidelines and theories from many HCI domains during the design of an interactive system. A requirements analysis phase usually guides students to analyze the source of a problem and devise a design that would solve the problem and improve over current existing solutions. However, due to their lack of experience in interface design, students may leave out or misunderstand important design concerns that may negatively impact their design.

Even when design concerns are properly defined, another challenge for students is to understand the scope of the solutions to each one of these design issues. Due to their lack of experience with existing solutions, students often spend a lot of time creating designs that are disconnected from previous efforts. Most design concerns for interactive systems have numerous design solutions, each having advantages and disadvantages to their use. Success can often be measured by how well students understand and apply solutions to each important development concern to create a coherent overall design that solves an existing problem.

These problems may be inherent to traditional learning techniques based on textbooks and lectures usually used in classrooms. *Constructivism* has challenged this approach by stating that knowledge is actively constructed by students as they experience it and combine it with previous knowledge to form a coherent mental model as opposed to being passively transferred from teachers (Ban-Ari, 1998). Parker and Becker applied constructivism in a Computer Science class and found an increase in student performance over traditional learning methods (Parker & Becker 2003). Constructivism in the classroom can be introduced through problem-based learning for HCI design reliant on five stages constructed around problem analysis, information gathering, synthesis, abstraction and reflection (Vat, 2000; Vat, 2001). A key stage in this approach that this thesis tries to support is *information gathering* where students look for valuable references from which they can build experience necessary to construct accurate mental models of design. The main challenge for teachers is to create adequate design references to facilitate student building of the design knowledge that is at the base of a constructivist approach to learning. This thesis proposes using reusable design knowledge encapsulated in *claims* to act as effective references, or design records, from which students can draw design knowledge.

1.2 Knowledge Reuse to Build Design Records for Learning

Information gathering is at the base of a constructivist approach to student learning. Allowing students to effectively search and gather design knowledge can improve the knowledge they will gain from problem-based learning (Ellis et al., 1998). It has often been noted in our experience teaching introductory HCI classes at Virginia Tech that students, due to their lack of experience, often leave out important design concerns which negatively impact their design. A constructivist approach to learning, by enabling them to gather important design knowledge in the form of design records to reuse in their designs and to understand the scope of design problems, can reduce their temptation to often consult radical invention and ignore important knowledge that would improve their understanding of HCI design.

Whittaker, Terveen, and Nardi (2000) addressed a problem of radical invention in HCI and proposed making incremental improvements based on prior designs. They believe designers should consult radical invention only when they have previously referred to prior work and cannot improve upon it. They propose designing around reference tasks, outlining common problems and metrics required for evaluation. The concept of reference tasks can be used to allow students to focus on what requires design, and help them in their information gathering stage of a problem-based learning approach. However, reference tasks do not inform on the mechanism by which knowledge can be encapsulated into design records by teachers to transfer knowledge to students.

Knowledge reuse entails the notion that design processes incorporate previously created artifacts to aid in the creation of new designs. The artifacts can appear in many different forms. The benefits of reusing design patterns in HCI (Borchers, 2000) as well as other forms of design knowledge such as claims or cases (Sutcliffe & Carroll, 1999; Berry, 2004) have been explored. It can often be more difficult to create new design knowledge than to reuse and adapt existing knowledge for new designs. The encapsulation of features, observations, and theories in the form of design knowledge such as *claims* can serve as a delivery method for design alternatives which can be used as references for a problem-based constructivist learning approach. Their use during the design process can certainly prove beneficial to students aiding their decisions by weighing possible design choices and consequences. However, although reusable design components such as claims can capture design knowledge and act as development records for student to use in their problem-based learning experience, their applicability, accessibility, and reusability is critical to their success (Dusink & Van Katwijk, 1995). One of the goals of this thesis is to improve on the applicability, accessibility and reusability of claims as design components to facilitate student gathering of important design knowledge.

1.3 Creating an Interactive Learning Environment Based on Reusable Design Knowledge to Support the Learning of HCI

This thesis aims to assist the learning of HCI design by creating an interactive learning environment based on reusable design knowledge components to support a problem-based constructivist learning approach for HCI. Although there are known advantages to the use of constructivist approaches for scientific domains such as HCI (Parker & Becker, 2003), we have seen that the problem for teachers lies in creating the adequate references from which student can draw knowledge as well as an effective form of encapsulation for this knowledge to increase its applicability, accessibility, and reusability.

This interactive learning environment proposes *claims* as a mechanism to encapsulate design knowledge and make it available for students to explore. Student can reuse claims knowledge to build their own designs and understand the scope of the design domain. Such reuse can focus student efforts on previous knowledge rather than on new, radical designs. Reuse often implies the need for a repository containing design components to make them accessible to designers. By searching or browsing for claims, students can create a collection of components and integrate them to create their design. However, there are problems associated with claims reuse. Claims must be organized in ways relevant to their design domain to maximize their accessibility. They must also be defined such that their applicability and reusability is maximized. The thesis statement is the following:

This thesis assists the learning of HCI design with the creation of an interactive learning environment based on design claims to support a problem-based constructivist learning approach for HCI. To maximize the benefits of this approach, claims are stored in knowledge repository that focuses on applicability, accessibility, and reusability to share design knowledge among students. By allowing students to explore concerns and alternatives for the design of interactive systems, this constructivist learning environment can improve students understanding of the design scope and lead to an increased use of existing HCI knowledge in their designs.

This thesis proposes and evaluates a constructivist learning environment to improve student information gathering through 5 main objectives.

- Introduce claims as applicable references that encapsulate design features from which students can gather design knowledge.
- Improve the applicability, accessibility, and reusability of claims by introducing reference claims to ameliorate student gathering of design knowledge.
- Illustrate how design knowledge can be encapsulated into reference claims to create an applicable, accessible and reusable interactive learning environment.
- Validating the applicability and reusability of reference claims for student designs.
- Evaluate the benefits of this learning environment based on reference claims in terms of improving student designs and increasing the amount of HCI knowledge they reuse.

Chapter 2 provides a summary of related work. The advantages of a constructive, problem-based learning approach is introduced and compared to traditional learning techniques. Reuse theories such as reference tasks and claims are introduced as means to create development records used in problem-based learning. A claims library integrated into a development environment is introduced as a tool to make these development records more accessible and reusable. Finally, the notification system domain is introduced as the application domain that will be used to create a first learning environment based on claims and described in Chapter 3.

Chapter 3 presents the first attempt at encapsulating design features into claims and storing it in a repository from which students can gather design knowledge. The initial library prototype focuses on claim entry forms that would make student claims consistent and reusable. An initial testing outlines the accessibility, applicability, and reusability problems associated with this approach and which are addressed in the subsequent chapter.

Chapter 4 proposes improving the applicability, accessibility, and reusability of the design knowledge repository by introducing a taxonomic organization of reference claims to share core design knowledge among students. A taxonomy was designed as a way to expose students to important design concerns as well as a method to categorize for design components. Reference claims are introduced as a reusable component inspired by reference tasks to expose students to design alternatives for design concerns. The creation of a learning environment based on reference claims organized in a taxonomy is addressed in chapter 5.

Chapter 5 illustrates how design knowledge can be encapsulated into reference claims to create an applicable, accessible, and reusable knowledge repository. A detailed taxonomy and a set of reference claims for the domain of notification systems demonstrate how existing theories of design can be encapsulated into reference claims. The example of the domain of interruption is used to illustrate how a design taxonomy can be improved by expanding its sub-domains with a new set of reference claims.

Chapter 6 proposes an experiment to validate the applicability and reusability of a set of reference claims from Chapter 5. An illustration of how reference claims knowledge can be improved based on the experiment results is also provided.

Finally, Chapter 7 evaluates the benefits of this learning environment proposed in chapter 5 in terms of improving student designs and increasing the amount of HCI knowledge they reuse. Conclusions are made on whether using an interactive environment based on reference claims can improve students' understanding of the design scope and lead to an increased use of existing HCI knowledge in their designs.

Chapter 2

Background and Motivation

The emergence of HCI as a major discipline of Computer Science led to the development of many tools and theories to improve the design of user interfaces. Although initially developed for experienced designers, many of these tools can be used at the base of a constructivist approach for students learning human-computer interaction. Constructivism applied to HCI through problem-based learning requires a set of development records from which students can gather knowledge to build their experience within the design domain. Reuse techniques such as reference tasks and claims can be used as means to create development records that encapsulate design knowledge and expose students to previous design efforts. A claims library integrated into a development environment can be used as a tool to make these development records more accessible and reusable to facilitate student information gathering of design knowledge in a problem-based learning situation. Finally, the notification systems domain can be used as an application domain to evaluate this approach.

2.1 Constructivism at the Base of Student Learning

HCI is usually taught through lectures and textbooks which convey various models and theories of design. However, this traditional approach where students are passively transferred knowledge from teachers has been challenged by constructivism, a novel theory of leaning which appears more successful in teaching scientific materials (Ben-Ari, 1998).

Constructivism states that knowledge is actively constructed by students as they experience it and combine it with previous knowledge to form a coherent mental model (Ban-Ari, 1998). Parker and Becker examined the advantages of using a constructivist approach to teach computer science classes (Parker & Becker 2003). They found that the constructivist approach required more efforts from students but that it also led to an increase of their performance in subsequent classes compared to students that were taught materials through more traditional approaches.

Vat proposes using constructivism in the classroom by introducing a problem-based learning approach for HCI (Vat, 2000; Vat, 2001). Savery and Duffy (1995) also explored this approach. It is based on five stages constructed around problem analysis, information gathering, synthesis, abstraction and reflection.

Berry (2004) detailed how these stages can be used in an HCI class. In the problem analysis stage, students are presented a problem that consists in developing a user interface. Students are asked to examine potential design solutions based on their current knowledge and to plan design activities. In the information gathering stage, students search for related knowledge such as development records that is needed to conduct their design activity. He states that students can learn by example through a set of use cases that illustrate how designs are created. In the synthesis stage, students then combine the knowledge they searched with their previous knowledge to create a model for their design. Through the abstraction stage, they can compare previous efforts to their own effort to better understand the process behind the design of their solution and increase its applicability to other problems. Finally, the reflection stage allows students to reflect on the new knowledge they have gained by reviewing it. One of the key stages in this approach that teachers can support is *information gathering* where students look for valuable references pointed out by teachers and from which they can build experience necessary to construct accurate mental models of design.

2.2 Reusable Knowledge Components at the Base of Constructivism

We have seen that information gathering is an important stage of problem-based learning. For the domain of HCI, teachers need to provide students with a set of references from which they can draw concepts for their designs. Ellis et al. (1998) insist on the importance of these reference materials for problem-based learning. Berry's work focused on teaching students the process behind HCI design, and as such, he used case studies as development records to act as references to illustrate how designs are created (Berry, 2004). However, there are other possible approaches to this. Specifically, a use case approach provides students with examples as to how to design a system, but no concrete examples of what specific features are necessary and how to implement them. This section introduces reusable design knowledge components to provide reusable examples of necessary design features and implementations that can be used at the base of a constructivist approach to teaching HCI design.

The concept of reusing previously created solutions for the design of interactive systems has been a recurrent theme over the past few years. Reuse can ease the development process by reducing the amount of time and effort put into creating solutions that have previously been designed and tested (Dusink & van Katwijk, 1995). But creating successful reusable components is difficult. It requires focusing on the right design issues and making design solutions and concepts reusable for a large scope of possible implementations.

2.2.1 Reference Tasks to Create Reusable Knowledge Components

Whittaker, Terveen, and Nardi (2000) propose that designers base their solutions on reference tasks to provide a shared body of knowledge for HCI. Reference tasks aim to capture core knowledge about common tasks to target existing problems. By concentrating on predetermined tasks, researchers can continue to work on designs for those tasks—a response to the problems associated with radical invention—and gauge their improvement over time using objective metrics. Instead of reinventing designs each time in HCI, the authors believe focusing efforts on common problems will yield long term benefits and measured improvement. The authors describe a set of criteria for determining what qualifies as a reference task. They instruct researchers to understand how frequent, critical, and real, a task is to determine whether it qualifies as a reference task. Care must be given to choosing tasks that will not become obsolete in the future. Examples of reference tasks include information sharing, document processing, and task management (Whittaker, Terveen & Nardi, 2000). It is also proposed that experts within HCI fields gather at conferences and workshops to define and agree on reference tasks and share them. Aspects that should be considered during the creation of reference tasks include shared requirements, accepted task definitions, descriptive vocabulary, task decomposition, and metrics. Reference tasks can therefore help focus on designing consistent reusable components that address important design issues. If reusable design components were specifically addressing important design concerns such as reference tasks, the list of reusable components could focus student attention around what needs to be designed, and help students in their information gathering stage of a problem-based learning approach. However, reference tasks are not sufficient to build development records. The form and mechanism by which design knowledge can be encapsulated into development records remains an important issue that needs to be addressed and which is not covered by Whittaker, Terveen and Nardi.

2.2.2 Claims as Reusable Knowledge Components

The reuse of commonly acknowledged components also has been widely researched over the years. A common body of reusable components can allow researchers to concentrate on problems associated with design alternatives. Reusable components must be found, selected, understood, and, if needed, adapted by designers during design (Dusink & Van Katwijk 1995). These components are typically retrieved from a reuse repository. The success of a user locating a potentially useful component can depend on several factors, including their familiarity with the repository and degree to which characteristics of a component have been specified (Creech, Freeze & Griss 1991).

Sutcliffe and Carroll (1999) propose claims as a means of encapsulating HCI knowledge that is associated with a specific artifact and usage context. Claims describe the upsides and downsides of design decisions and records HCI knowledge related to a specific design. They are designed as a way to store knowledge during the design process. To ensure that the claims knowledge is reusable for various system designs, claims have to be classified and generalized. Sutcliffe and Carroll present methods by which to classify claims. Their schema elaborates how to describe HCI knowledge and create new claims from existing claims or artifacts. Many tools can be developed for claims; claim relationships for example, can be created based on Norman's model of action (1986). Child claims can be derived from parent claims describing different stages of the interaction process. Claims can therefore be used as a mechanism to store design knowledge which can act as a development records that student can use in a constructivist problem-based learning approach HCI.

2.3 Accessibility, Applicability and Reusability of Reusable Knowledge Components

Although claims can effectively capture design knowledge and act as development records for student to use in their problem-based learning experience, the issues of accessibility, applicability and reusability of these claims remains an issue (Dusink & Van Katwijk, 1995). First claims need to be stored to make them easily accessible. Students also need to synthesize this set of components to understand how this collection of claims can be applied to create new systems. Finally, students need to be guided in the process of selecting these components. These steps are essential in supporting the main stages of problem-based learning described earlier.

2.3.1 Claims Library to Store Design Knowledge

Payne et al. (2003) addressed the issue of accessibility of claims and developed a claims library for the development of notification systems. The library contains claims about notification system design aspects and aims to make it easier and quicker for students to build their designs. A claim is represented as a set of attributes to capture the effect a feature will have on a user within a usage scenario. Students are able to search for claims using keywords and receive matching claims as result. When needed, students can create new claims and add them to the library. Figure 2.2 is an example claim from the claims library.

Using MSN style pop-ups to alert *A clickable window about 1/24th the screen size appears in the bottom right hand of the users work area for a limited number of seconds, briefly alerting the user of a certain event, then disappearing. If the user clicks on it, it will give more details about the alert and possible actions to take.* + User gains a quick overview of information which could be important to him or her. - Could appear on top critical information in the primary task application.

Figure 2.1: A claim in the claims library

However, problems inherent to the characteristics of reuse still exist. The organization of the claims in the claims library often inhibits the amount of reuse. Even when reuse does occur, claims may lack quality and not accurately describe the intended designs. The lack of organizational structure also leads to problems in understanding the design domain (Wahid et al., 2004) and selecting claims that adequately cover important design concerns. Understanding the characteristics of the domain is essential to successfully reusing. The claims must also be written to maximize their reusability and must cover important design concerns for a particular domain.

2.3.2 System Image as a Synthesis Mechanism for Claims Collections

We have seen that claims drawn from a knowledge repository can be used by designers to draw ideas and concepts for the design of interactive systems. The set of claims designers select describe the features they intend to implement and therefore represent an image or prototype for their design. Research has been done to evaluate how a collection of claims can be synthesized to describe a system and how an integrated development environment for interface design can guide to the creation of a system image that successfully acts as a design prototype. Our research group developed an integrated and reuse environment for notification systems (Lee et al., 2004). This work is based on Norman’s concept of a system image (1986). The system image is a representation of a design that acts as a bridge between the designer’s conception of a system and the user’s conception of the system’s use. Lee et al. suggest that a better understanding of the system image is essential to the successful evaluation of design prototypes. This design processes makes the system image the central communication point between different stages of design and between different stakeholders. Findings indicate that the effective creation and use of knowledge repositories by HCI designers hinges on the successful application of existing HCI design concepts within a practical integrated design environment.

Chewar and McCrickard (2005) designed an integrated design environment in which Lee developed a system image module. Their system, LINK-UP, proposes an interface for the design process of notification systems. The concept of the system is to provide notification systems designers with a facility for task-based design advice, consistent with the guiding progression throughout an interface design process. Lee's findings defined that a system image representation, expressed in terms of design claims, provides enough information for knowledgeable HCI persons to quickly evaluate prototype designs. A collection of claims can therefore represent a system image that successfully acts as a design prototype that synthesizes the knowledge student gathered from the claims library.

2.3.3 The LINK-UP System to Guide Knowledge Reuse

LINK-UP (Leveraging Integrated Notification Knowledge through Usability Parameters) is a web-based design environment that guides users through the design of notification systems (Chewar et al., 2005). The system is composed of 5 modules corresponding to different stages in the design process (see figure 2.3). The first module is the Requirements Analysis Tool. This module helps designer define the information to be conveyed by the system and the main tasks to be supported. A questionnaire defines critical parameters to achieve during the design (Chewar, McCrickard & Sutcliffe, 2004).

The second module of the LINK-UP system includes the claims library previously discussed (Payne et al., 2003). Designers select claims in the library that they think will achieve the goals defined in the requirements analysis phase. The third module, the Negotiation tool supports an interactive process through which designers exchange ideas, and provide feedback on the design decisions made with the requirements analysis module. Claims extracted from the library are discussed, some are rejected and others added. The final set of claims is the system image of the designed system or *design model* (Lee et al., 2004). The fourth module is an analytical module. Once the system image is entered in the LINK-UP system, analytical experts can review the prototype and provide feedback on the usability aspects of the systems. Specifically, experts will provide feedback on the downsides of the claims selected, and on the critical parameter values that have been reached using this set of claims. Finally, the last module is the empirical testing module. This module allows designers to test a set of claims and to calculate the critical parameter values perceived by users.

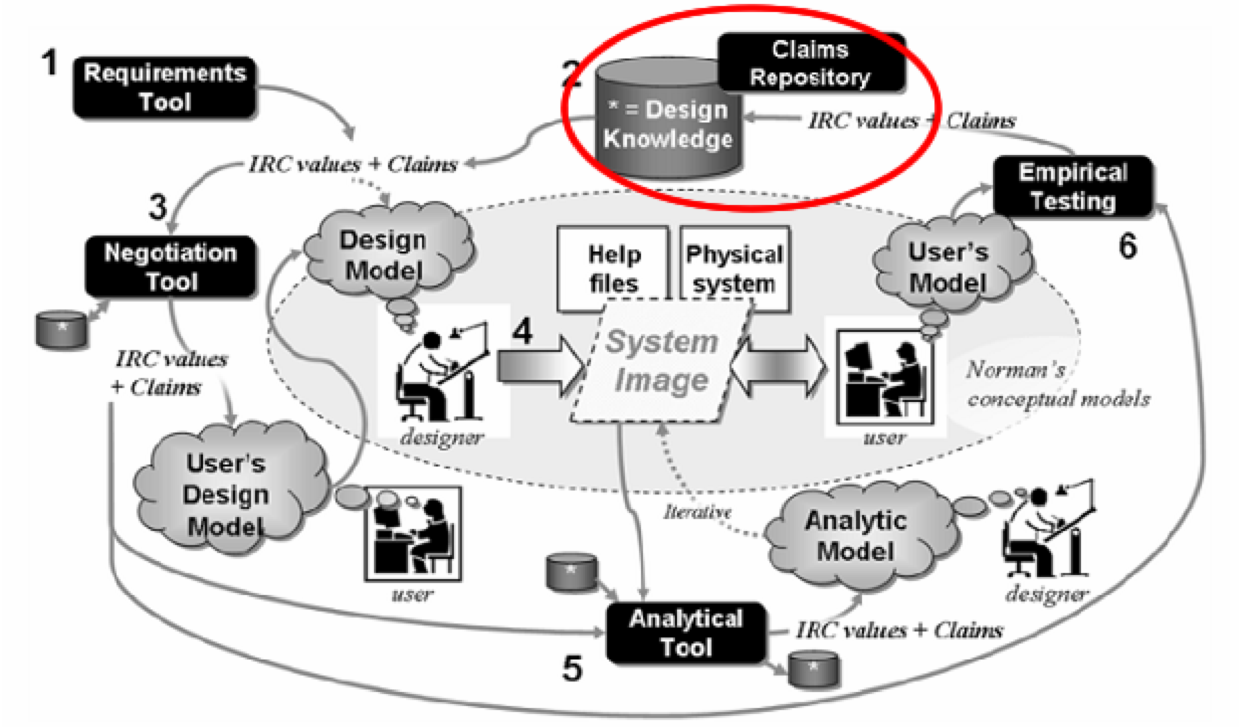


Figure 2.2: LINK-UP system to design interactive systems.

One of the main components of LINK-UP is the claims library that acts as the knowledge repository from which students can draw design features for their designs (circled in red on figure 2.3). We have seen the problems of the current claims library. This work aims at improving the accessibility, applicability and reusability of claims to increase the amount of HCI knowledge students can use to create their designs in a constructivist approach to learning.

2.4 Notification Systems as an Application Domain

To design and evaluate a problem-based learning approach with claims as development records, the design domain has been constrained to the domain of notification systems. Notification systems are systems that are used in a dual task situation. While performing some tasks, such as writing a word document on a computer desktop, notification systems provide interruption to display some monitored information. In other words, these systems provide transition between a primary task (i.e.: writing a document), and a secondary task (i.e.: monitoring weather conditions). Notification systems are invading our everyday lives. GPS systems in our cars guide us while driving; email programs on our computers notify us of incoming emails while working, and ambient displays in our streets notify us of the time or current weather conditions. Because of the variety of domains in which notification systems are needed, it is important to understand the goals of the systems we design in terms of conveying the information in an appropriate manner. The notification systems domain was chosen because it has a well defined framework describing system and user goals (McCrickard, & Chewar, 2003). The framework can be used as a method to gauge the effectiveness of a system's information design options and user goals. It also defines 8 unique notification system types based on differing critical parameters.

2.4.1 Notification Systems Design Framework

McCrickard and Chewar (2003) introduced a design framework for notification systems, interfaces designed to convey information from sources secondary to current activities. Through this framework, they define the tradeoffs of user notifications in terms of the impacts on user attention, reaction to the notification, and comprehension of the information conveyed. They established that the success of a notification system relies on effectively allocating attention between tasks, while allowing access to secondary information. They demonstrated that comprehensible notification which prevents unwanted distraction from a primary task and can still deliver additional information, depends on specific design attributes. Their framework models the tradeoff between utility and attention cost, by associating the design attributes to notification interfaces. These design attributes, defined as three parameters based on interruption, reaction, and comprehension, describe user goals. Because it integrates user goals, system design models, and presentation options, their framework simplifies design decisions for developers and suggests concerns for researchers.

2.4.2 Notification Systems Design Scope

Designers can use the framework to understand the effects of information design options on the achievement of user notification goals. McCrickard and Chewar's study demonstrates the vast advantages of AUIs for meeting these goals with information displays specifically adapted to maintain an effective balance between attention costs and utility benefits. Systems developed for specific characteristics can prevent design problems associated with cognitive differences and interface learning ability. The following representation of the IRC framework illustrates how the analysis of these parameters generates a scope of notification system designs.

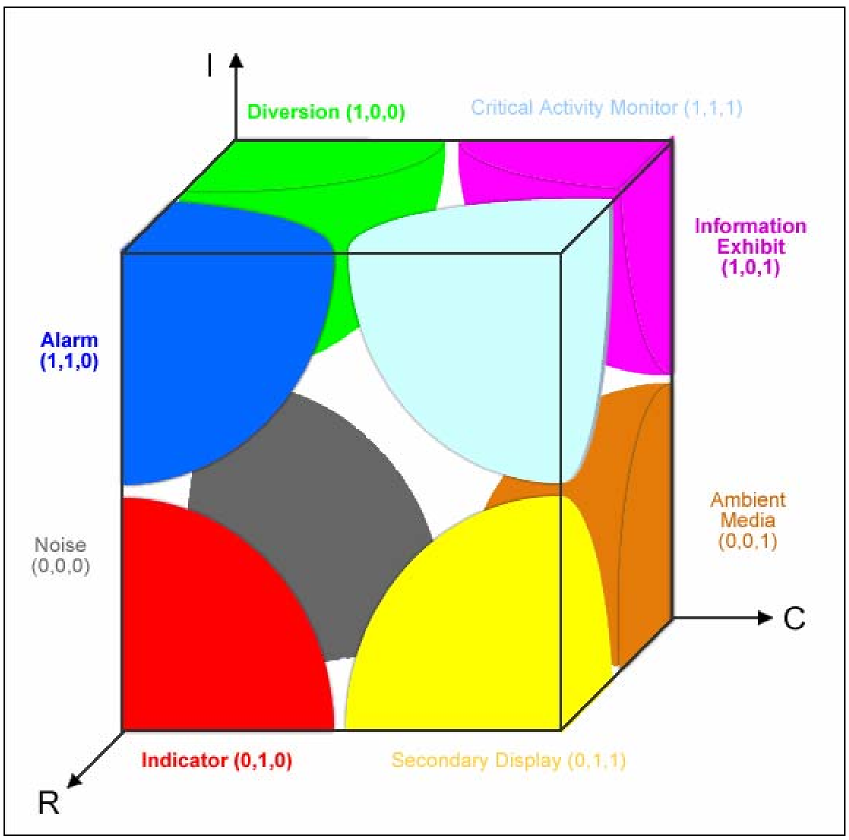


Figure 2.3: Representation of the IRC framework for notification systems.

The 3 critical parameters, I for Interruption from a primary task, R for user Reaction to notifications and C for Comprehension of the information conveyed, have numeric values that vary from 0.0 (low) to 1.0 (high). The various combinations of values for those three parameters lead to different notification system designs that have different design goals. Alarm systems for instance have high interruption, high reaction and low comprehension because it is critical to interrupt users from their primary task and relocate their attention to a secondary task (escaping a building for instance) without needing them to understand why the alarm went on. Other systems such as ambient displays have low interruption, low reaction and high comprehension. These systems are usually situated in the user periphery to monitor information such as weather conditions. It is not critical for them to interrupt users from their primary task or force them to react to their notifications. However, they are designed to convey highly comprehensible information that can be understood by just glancing at the system. Notification systems can therefore be categorized by the different types of design goals determined by the IRC cube. Each corner of the IRC cube can be used as a base to categorize and define reusable components for the domain of notification systems that could help students understand major design concerns and alternatives for the design of these systems.

Chapter 3

A Claims Library to Share Design Knowledge*

We have seen that constructivism can benefit student learning and that a key stage in a constructivist learning approach for HCI design is *information gathering* where students look for valuable references from which they can build experience necessary to construct accurate mental models of design. We have also seen that reusable design components such as claims can be used as references from which students can gather information for HCI and that reuse principles such as reference tasks and design claims can guide in the creation of these components. But the main challenge for teachers is to create effective design components to support student information gathering of HCI design knowledge. This chapter proposes using reusable design knowledge encapsulated in *claims* to act as effective references, or design records, from which students can draw design knowledge. In a first attempt to store design knowledge into claims and make it available for students to reuse, our Notification Systems research group created an initial *claims library* prototype as a knowledge repository. We based our design behind the idea that successfully designing a repository would allow students to effectively retrieve ideas that would broaden their HCI knowledge by exposing them to a range of solutions to solve their design

* Sections of this chapter are covered in Fabian et al., 2003

problems. This chapter presents an initial prototype for *the claims library* as an HCI design knowledge repository.

3.1 Traditional Claims as Reusable Design Knowledge Components

To encapsulate design knowledge, we decided to use Sutcliffe and Carroll's claim concept we previously discussed (1999). A claim is a statement that describes the effect a feature will have on a user within a usage scenario. The ideas presented in a claim allow students to compare them with one another based on design techniques and testing outcomes. Claims make explicit the ideas that are present in designing an interactive system.

Claims consist of a title, description, upsides and downsides, scenarios, theories and artifacts. Critical parameters such as the IRC parameters discussed in chapter 2 for the notification system domain can be assigned to claims to measure the impact of the use of their feature in a particular design. An example claim from our prototype library is presented in Table 3.1.

Our *claims library* is a repository that stores reusable claims for the design of interactive systems. The main goal of this library is to allow students to search for design alternatives in the form of claims, through multiple criteria, and receive matching claims as results. We believe that finding a quality claim and reusing it in a design supports a constructivist approach to learning that allows for rapid prototyping and hinders the need for the students to consult radical invention. Claims allow students to evaluate the tradeoffs of design alternatives and to decide which design choice suits their needs the best. The claims repository grows as students add new claims to the library when they cannot find claims that suit their needs.

To successfully transfer design knowledge to students, claims need to be reusable and accessible. We propose improving the reusability and quality of claims by structuring their entry in the claims library. We also propose a new classification method for claims based on quality ratings, features of usage and critical parameter values (McCrickard & Chewar, 2003) to maximize their accessibility. To illustrate our proposition, we constrained our claims library to the domain of notification systems previously discussed.

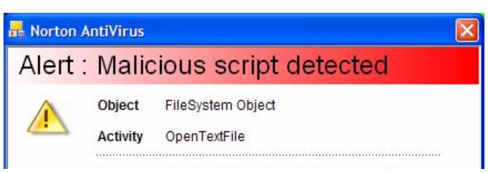
Title	Virus monitoring through informational pop-ups
Description	Virus scanning is done through the background, and when problematic and possibly infected files are found a popup will appear (regardless of what program currently has focus) and will prompt the user to delete, quarantine, or ignore the virus that was detected. The Symantec™ name of the virus is also displayed.
Upsides	+ Quick notification to the user of the virus activity regardless of what program has window focus. + Brief description of the virus (the name) allows user to research on his own on how to react. + Three commonly selected options allow the user to immediately react to possibly save his system.
Downsides	- No information is displayed, leaving virus information gathering up to the user. - Advanced options (such as using specific removal tools) aren't available from the brief popup. - Pop-ups when the virus is easily seen (such as on emails that obviously shouldn't be opened) can become annoying.
Design Issues	Is a more descriptive block of information on the virus warranted or desirable? Would more options on reacting to the virus be detrimental to the simple interface of the popup window? Should a configurable sound alert be added in case the user is away from his computer, or running an application that won't allow the popup to take focus?
Scenarios	Sarah is working on a spreadsheet to keep track of her balances. She has been working for hours, and decides to take a much-needed break to check her email. She only sees one new message from her friend Lisa, containing a brief message with a small executable attachment. Sarah clicks on the executable attachment, watches a brief fireworks animation as a result, then deletes the email and returns to her spreadsheet program. After a few minutes of working on her balance sheet, her virus program pops up a window notifying her of a new Worm virus that was installed on her system. She then presses the quarantine button and returns to her spreadsheet work.
Artifact	Norton Antivirus™ 
IRC	Interruption: 1 Reaction: 1 Comprehension: .5

Table 3.1: Example claim in the library prototype

3.2 Initial Prototyping

Our first goal was to create claims entry and editing forms to ensure the consistency of student claims. These activities focused on six different features to support consistent and high quality claim entry: multiple entry pages, explanation of entry fields, user claim rating, supportive media, and checklists for specifying aspects of feature usage, and critical parameter values. Brainstorming sessions were conducted to discuss the best way to implement each of these features. We wanted to ensure that the interface enabled any HCI student to enter a claim in a consistent manner. We worked under the assumption that students would have no previous knowledge of the claim attributes. The library claim features are described below.

3.2.1 Multiple Entry Pages

To design the claims entry form, we decided to use a multiple page entry format. A progress bar was added to each entry page to inform students of the claim creation status. This solution was designed to avoid students from getting overwhelmed by a single, long entry page. Entry fields were grouped into coherent groups to maintain a ratio of two to four entry requirements per page. A fifth page was added for claim entry confirmation where students could confirm or edit their claim before adding it to the library.

3.2.2 Entry Fields

To ensure students would understand the meaning of the claim fields, we added succinct directions for each claim field (see Figure 3.1). This was designed to help students in their claims entry process by giving them a general sense of what was expected for each claim field. Under each description, we added a link that describes each field and provides concrete examples. This was done to assist students that felt like they needed a more detailed explanation of what type of information was expected. The link opens a small pop-up window showing a sample of what type of information goes in the field. The pop-up window also displays screen shots of existing systems with high, medium, and low critical parameter ratings. Part of our user testing focused on evaluating how effectively the explanations guided the students’ claim entries.

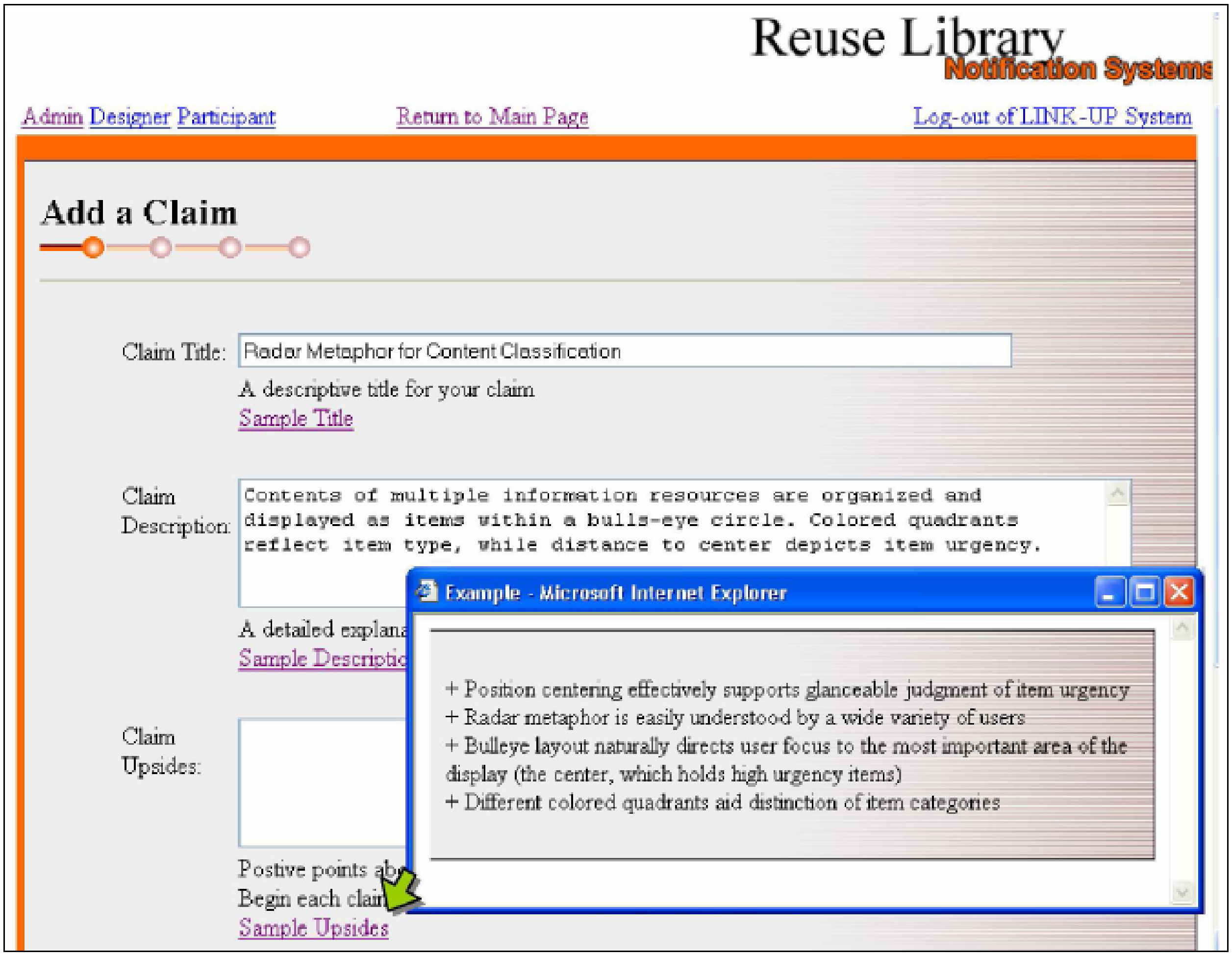


Figure 3.1: Claim entry form.

3.2.3 User Claim Rating

To assist students in selecting quality claims, we wanted to support user feedback on claim effectiveness. We created a star rating system based on a point scale created from six distinct variables. Points for each variable are tallied and the number of stars for each claim is determined based on the total number of points obtained. A claim with zero to nine points is rated with one star, ten to nineteen point claims receive two stars, twenty to twenty nine point claims receives three stars, thirty to thirty nine point claims receive four stars, and forty or more point claims receive the maximum five star rating. There are no fractional star values for simplicity. The variables used to calculate the ratings are the following:

- ***Theory:*** If the claim is based on an HCI theory, it receives between five to ten points depending on the quality of the theory. If the claim is not based on any theory it does not receive any points for the theory variable.
- ***Ratings from other claims by user:*** The average of all of the user's previous claims' ratings divided by the total number of variables (6). For instance a user with an average claim rating of 25 receives four points (rounded) for his new claims.
- ***Author's experience:*** Zero to ten points are assigned based on three levels of expertise with claims writing (Beginner, Novice, Experienced)
- ***Artifact:*** If a claim has an artifact attachment demonstrating its usage in an existing system, it receives an additional five points.
- ***Number of hits:*** Zero to ten points are assigned to a claim based on the number of times the claims has been viewed by students in the claims library.
- ***Other users' rating of the claim:*** The average rating given from other students assessing the claims quality when browsing through the library.

This claims rating was created with the purpose of assisting students into selecting quality claims for their designs.

3.2.4 Supportive Media

We believe that including supportive media will make the claim easier to understand and the artifact mentioned easier to visualize. We included the option to upload pictures or other forms of media that will support the use of the claim in the usage scenario. The media examples that are uploaded appear below the scenario when visualizing the claims.

3.2.5 Checklists for Specifying Feature Usage

We wanted to create an effective way for users to specify the features and usage environment addressed by a claim. We designed a checklist of claim options that describe the claim's Primary Task(s), Notification Task(s) and Design Properties. The rationale behind the checklist was to give designers a list of claim features which could be used to search for claims in addition to the standard keyword search (Sutcliffe, 2002; Payne et al., 2003). By using this list of features to search for claims, we expect to increase the claims' accessibility.

3.2.6 Critical Parameter Values

We anticipated that it may be difficult for students to achieve consistency in selecting the IRC critical parameter values for their claims (McCrickard & Chewar, 2003). We decided to create a web based questionnaire to help students define their claims IRC values. We implemented a "Get Parameters" function that acts as a "wizard" to evaluate the answers to eight multiple-choice questions. The answers to these questions generate the three critical parameter values for their claim.

3.3 Initial Prototype Testing, Results and Analysis

The second phase of our research consisted in validating the claim classification scheme. Multiple rounds of testing were administered with student participants. Between these rounds, changes were made to improve the consistency of the classification format.

3.3.1 Preliminary Testing

Our preliminary testing objective was to see if the initial interface design was effective for entering claims. We were interested in isolating features that needed further development. We tested our system with six students who had successfully completed an HCI course offered at Virginia Tech. These users were given four claim entries that were missing the IRC values, primary tasks, notification tasks, and design concerns. The users were asked to fully review the claims and then decide on its IRC values. They were also asked to complete the primary tasks, notification tasks, and design concerns checklist. The participants used the web interface to enter their results. The help window was available on the interface to guide their decisions.

The initial test on former HCI students did not bring the results we were expecting. The results validated the rating options and the screen flow; however, the IRC ratings, primary tasks, notification tasks, and design concerns were not consistent. Participants were confused by the meanings of some of the terms. In particular, the “Primary Task” term was the most confusing. Most participants did not understand the difference between each primary task option. Towards the end of the test, the participants were becoming frustrated. We believe another factor yielding users' frustration was a heavy reliance on claims extracted from unfamiliar artifacts.

3.3.2 Redesign and Internal Validation

Our preliminary results led us to redesign our classification scheme. We decided to replace the “Primary Task” type with a “Primary Notification Environment” category. We believed this category better exemplified the claims usage. Although the list of generalized tasks we originally used for classification of primary tasks allowed cross domain reuse (Sutcliffe, 2002), we decided to start with a less abstract classification approach. Additionally, we changed the IRC parameters values to high, medium, or low values to replace the decimal values between 0.0 and 1.0. We believed that giving the user three choices for each parameter would increase consistency. Students agreed it would be easier to determine low, medium and high ratings instead of decimal values. We combined several of the notification tasks and updated their definitions. The group thought that by combining similar choices we would increase consistency. We implemented all of these changes in a new prototype.

The second round of testing was internally administered. To achieve accurate results, one group member was chosen to be the independent administrator of the experiment. This created an unbiased testing environment. For this round of testing, we chose claims that used more familiar artifacts, such as a cell phone vibrate feature and the sound notification in AOL Instant Messenger™. Our internal testing validated the effectiveness of most of these changes. As the experiment progressed, it was evident that our group was more decisive due to the familiarity of the artifacts behind the claims. Additionally, a strong majority of the group was correctly able to identify the primary notification environment. The updated notification task titles and definitions proved to be the needed changes to achieve consistency for the claim fields. The values achieved in testing the IRC values were more accurate than in our previous tests but we were not able to achieve the level of consistency that we desired. However, we were unsure of how to reengineer this aspect of the claim classification feature. Uncovering the reasons behind the inconsistency became the specific goal of the third round of experiments.

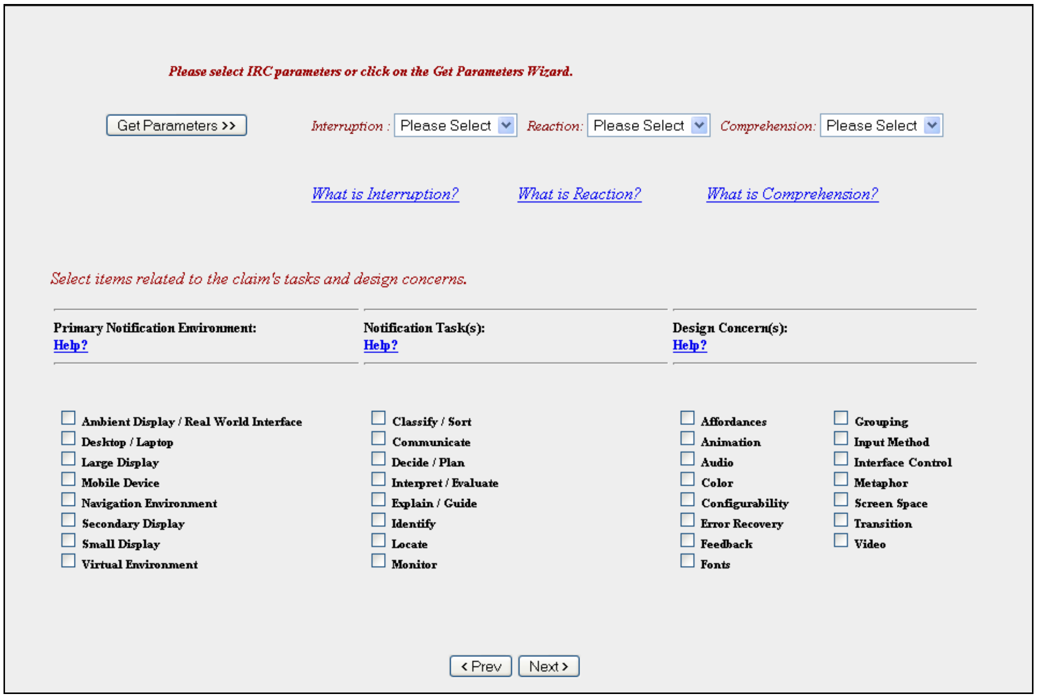


Figure 3.2: Claim feature and critical parameter selection in the revised claims entry form.

3.3.3 Formal Lab Based Testing

To evaluate our new classification scheme, we decided to conduct a third experiment. This experiment involved 11 students who were currently enrolled in an introductory HCI course. The test was conducted in two phases which we called searcher phase and classifier phase. In the first phase, the participants (acting as *searchers*) were presented the claim entry screen, a brief explanation of what a claim is, and a claim summary. The participants were given time to complete two claim searches during this portion of the test. This involved reading a problem and determining the low (0.0), medium (0.5), and high (1.0) IRC values, notification environment, notification tasks, and design concerns in order to retrieve a relevant claim from the library.

The following example is one of the problems that were given to students:

> *"Eric is designing an automobile computer system that will aid a driver in finding her destination using an already working GPS tracking component. Unfortunately, in the car Eric is building his system, he will not have any screen space available for a visual component. However, he wouldn't use one even if he could because he wants his system to be as safe as possible when used on the road. He wants the system to react as the user is driving, constantly updating the driver with information that will guide her to the destination."*

The second phase of the test involved the same participants (acting as *classifiers*) adding claims to the database. They were given existing claims and had to determine the IRC values using the web-based wizard. The distinction in IRC input methods (low, medium, and high vs. wizard) reflects the different levels of familiarity the two roles would be expected to have. A searcher would have a very general idea of appropriate constraints, while a classifier should be able to describe more subtle characteristics of the claim. The classifiers were also asked to identify the notification environment, notification tasks, and design concerns that would be entered into the library as part of the claim. The two phases were reversed for half of our participants to obtain data on searching and classification results for all design problems.

3.3.4 Analysis and Interpretation

We first focused our analysis on understanding the IRC classification results. For the IRC parameter testing, we had two groups of participants alternating as searchers and classifiers for claims on four problems. The manner in which participants specified IRC values depended on the classifier/searcher role. To analyze classifier-to-searcher match tendencies, we calculated the differences between the decimal values obtained by the classifiers and the values the searchers submitted. The general process used for this comparison is depicted in Table 3.2, although the process was repeated for each of the 11 classifiers and searchers. The example data shown in Table 3.2 illustrates a probable claim hit (shaded, Classifier B-Searcher X), as well as how inferences were made on classifiers, searchers, and each of the parameters.

<table>
<tr><td colspan="4" rowspan="2"></td><td colspan="3">Searcher X</td><td colspan="3">Searcher Y</td><td rowspan="3">Classifier Overall</td></tr>
<tr><td>I</td><td>R</td><td>C</td><td>I</td><td>R</td><td>C</td></tr>
<tr><td colspan="4">Classifier</td><td>0</td><td>1</td><td>1</td><td>0</td><td>1</td><td>1</td></tr>
<tr><td rowspan="2">A</td><td>I</td><td>R</td><td>C</td><td colspan="3" rowspan="2">Avg Difference
0.28</td><td colspan="3" rowspan="2">Avg Difference
0.48</td><td rowspan="2">Unacceptable
Avg Difference
0.38</td></tr>
<tr><td>0.25</td><td>0.8</td><td>0.1</td></tr>
<tr><td rowspan="2">B</td><td>I</td><td>R</td><td>C</td><td colspan="3" rowspan="2">Avg Difference
0.1</td><td colspan="3" rowspan="2">Avg Difference
0.27</td><td rowspan="2">Marginal
Avg Difference
0.18</td></tr>
<tr><td>0.1</td><td>0.9</td><td>0.6</td></tr>
<tr><td colspan="4">Searcher Overall</td><td colspan="3">Marginal
Avg Difference
0.19</td><td colspan="3">Unacceptable
Avg Difference
0.39</td><td></td></tr>
</table>

Table 3.2: Process used to analyze classifier-to-searcher match tendencies showing example results for two classifiers and two searchers.

The first result we found is that the overall average differences between searcher and classifier IRC values dropped 0.15 and 0.06 on second rounds for classifiers. This result suggests that, as classifiers became accustomed to the IRC system, their results became more consistent. We noted several examples where classification efforts would have resulted in probable claim finding by a searcher. Recalling that each of the eleven classifiers classified two claims, each of which were searched for by five or six searchers:

- Five classifiers would have had at least one claim found by a searcher
- Three classifiers would have had at least one claim found by two searchers
- One classifier would have had both claims found by a searcher

These results show that the system can be used successfully. However, overall classifier results showed an unacceptably wide range of IRC specification differences (overall std. dev =

0.27). In particular, the Interruption (I) parameter was significantly less consistently matched than the other two ($F(2, 357) = 4.48$, $p < 0.05$), suggesting the most critical need for reengineering.

These results may show differences in human performances and learning effect when classifying claims. When the same kind of experiment was run on expert users with a well-rounded knowledge of IRC parameters, however, results where much more consistent. There are two interpretations to this result. Either classifiers did not have enough understanding of the claim they were classifying or IRC classification requires a more expert understanding of the parameter specification process. Also, searcher performances showed that given a specific problem, participants tended to look for the same IRC values when searching for a claim, especially if “medium” responses were disregarded. This implies that the interface for specifying search parameters should be limited to two value selections— “high” or “low”—with the addition of an option for “uncertain” specification (where that parameter would not be considered in the query).

The second portion of our analysis focuses on the concurrence between claim adding and retrieval, based on the following categories: notification environment, notification tasks and design concerns. Test results strongly favored successful search attempts in the system. Every search resulted in at least two hits in two categories when “OR” comparisons within a category were used to make matches, so this searching technique proved to be successful. Only 6 out of 66 classifications were not strongly matched by searchers (meaning that less than two-thirds matched the classifiers). The average hit rate per classification was 88.3% with a standard deviation of 19%. The notification environment and design choice categories both had very high match averages of 93%. Results for two of the design problems are shown; highest match rates are presented in Table 5a, whereas Table 5b shows the lowest.

a)

Classifier	Environment	Tasks	Design
1	100%	80	100
2	100	80	100
3	40	80	100
4	100	80	100
5	100	80	100
6	100	100	100

Avg: 91% Std Dev: 16%

b)

Classifier	Environment	Tasks	Design
1	83%	0	100
2	83	66	100
3	100	83	100
4	83	100	100
5	83	100	100

Avg: 85% Std Dev: 26%

Table 3.3: Percentage of searchers agreeing with classifier specifications of claims.

3.4 Initial Prototype Conclusions

Our claims library was developed to help students with the design process of interactive systems. The biggest challenge we encountered was implementing an intuitive classification scheme that would help people find claims that correspond to their design concerns. Therefore, achieving consistency between claim adding and claim retrieval was an issue in designing the classification interface. In the system we designed, there are two complementary types of classifications to find claims. The first one is finding a claim in terms of its notification environment, notification task, and design concerns. When using this scheme, our results were consistent and it is very likely that students would be able to find helpful claims. The second classification method we implemented was based on the IRC parameters of a claim. By specifying those parameters, our library finds claims that are closer to users' design models for their notification systems. Our results show specific breakdowns in consistency that can be addressed with interface design or user training. The general results reported showed that IRC values could be consistently specified and used for searching.

However, there are still problems with this claims library approach. Most claims entered by students are unrestricted in content and usually span over many domains within the notification system domain which limited their applicability. Also, most claims were very specific and contained words and concepts that decreased their reusability as they increased their level of detail. Finally, most claims were tied to specific artifacts in their description rather than just as a usage example. All these factors limit the quality of claims in terms of applicability, accessibility, and their potential reuse by other students. These problems are tackled in the next chapter as a new form of claim is presented to increase the use of existing HCI knowledge by students.

Chapter 4

Redesigning the Claims Library with Reference Claims Organized in a Taxonomy*

We have seen that design knowledge can be encapsulated in a claims library and that claim entry forms and classification schemas can allow consistent claim entries that can be found when searched by students gathering HCI knowledge. However, the evaluation of this environment outlined problems associated with the accessibility, applicability and reusability of claims in the library. These problems are critical in that they hinder the ability of students to effectively search for knowledge that is critical in a problem-based constructivist learning approach. This chapter proposes redesigning the claims library by introducing a new form of claims, reference claims, to increase the applicability and reusability of claims, as well as organizing them in a design taxonomy to outline key design concerns and increase their accessibility.

* Sections of this Chapter are covered in Fabian et al. 2006

4.1 Design Taxonomy to Outline Design Concerns

Students are often required to apply various HCI guidelines and theories from many HCI domains during the design of an interactive system. A requirements analysis phase usually guides students to analyze the source of a problem and devise a design that would solve the problem and improve over current existing solutions. However, students, due to their lack of experience in interface design, may leave out important design concerns which may negatively impact their design. A design taxonomy can guide students into addressing the proper design concerns when they design their solutions.

4.1.1 Design Taxonomy Example

Taxonomies provide a decomposition of the domain, outlining topics that combine to define the domain. The notification systems domain was chosen as an example to illustrate this proposition because it has a framework defined by McCrickard and Chewar (2003) that effectively covers the key concepts of the domain. Figure 4.1 present an example taxonomy for the domain of notification systems.

Notification Systems

- **Critical Parameters** (Interruption, Reaction, Comprehension)
- **Notification Methods** (Voice, Visual, Haptic/Tactile, Audio)
- **Notification Systems Types** (Secondary Display, Information Exhibit, Critical Activity Monitor, Noise, Alarms, Diversion, Indicator, Ambient)

Figure 4.1: A notification systems taxonomy.

The notification systems domain focuses on three important aspects: critical parameters, notification system types, and notification systems methods. The three critical parameters of interruption, reaction, and comprehension and the 8 different notification system types are defined by the IRC framework. Much of the notification methods are derived from previous literature and the Virginia Tech Notification System research group expertise. A detailed explanation of the taxonomy is presented in Chapter 5.

The taxonomy was built behind the motivation to provide student designers with a set of important design concerns that define the domain as well as an effective categorization for the reusable design components (reference claims). These components are classified under each element of the taxonomy and represent a set of design alternatives for each design concern.

4.1.2 Benefits of the Design Taxonomy on Claim Accessibility

Students can use the taxonomy to browse through important design concerns they need to consider for their designs. Each design concern (Critical Parameters, Notification Methods and Notification Systems Types) has a set of reference claims classified under each sub-category of concern (Interruption, Reaction and Comprehension for Critical parameters for instance). The reference claims represent a set of design alternatives for each design concern. Students can therefore browse through each design concern and see a set of possible design alternatives to guide them in the creation of their own design. In our previous efforts to create a claims library, claims had a set of attributes and keywords students could search for, but claims had no organizational structure. This factor limited the accessibility of the claims, because students were restricted to “guessing” which claims they were looking for. A taxonomic organization should allow them to easily find claims for each design concern rather than to search for claims that may solve a particular concern. However, even if the taxonomic organization increased the claims accessibility, the knowledge they capture needs to be applicable and reusable in order for students to be able to use them for their designs. For this reason, reference claims are used instead of traditional claims for the reusable design knowledge components within the taxonomy.

4.2 Reference Claims as Reusable Design Components

We previously discussed the benefits of using reference tasks for HCI design. A similar notion can exist within the concept of claims which also can capture and share knowledge. One of the goals of this work is to define a new type of claim, reference claims, inspired by reference tasks and illustrate how student designers can benefit from the reuse of such claims in terms of applicability and reusability. *Reference claims* are a means to encapsulate design knowledge and focus student designers' attention. A collection of reference claims portraying relevant design knowledge allows designers to describe needed design features. The molding of all the reference claims could instantiate new ways for students to design an interactive system.

4.2.1 Characteristics of Reference Claims

In our previous efforts to encapsulate design knowledge and make it available to students, claims were encapsulating any design knowledge about any particular feature students would create for their designs (Payne et al., 2003; Fabian et al., 2004). As each student created a new design s/he was asked to reuse claims for his design and create new ones when the desired feature was not found in a claim. As new claims were created, they were added to our claims library and contributed to a growing set of reusable design components. As a result, the traditional claims in the library were unrestricted and usually spanned over many domains within the notification system domain. Most claims were very specific and contained words and concepts that decreased their reusability as they increased their level of detail. Finally, most claims were tied to specific artifacts that illustrated examples of possible use within certain context. All this factors limited the quality of claims and their potential for reuse among students.

Reference claims are different from traditional claims. To maximize their reusability, reference claims focus on applicability to the design domain, ability to lead to the creation of new designs, language abstraction to increase applicability and artifact independency to avoid restriction of its use to a single context. These four main characteristics that differentiate reference claims from regular claims are described below.

- ***Applicable***: Reference claims, like reference tasks, encapsulate knowledge central to a domain. Reference claims are created by educators who have expertise in a domain, giving them the ability to associate potential reference claims with their places in the domain taxonomy. Educators can maintain quality and coordinate the testing and validation of reference claims, contributing to a growing body of domain knowledge. The list of reference claims can grow as new research yields improving results. Chapter 5 illustrates how reference claims can be tested, validated and improved.
- ***Parent:*** Reference claims can act as parent claims from which traditional claims can be instantiated through specification. The knowledge encapsulated in reference claims allows students to use them by tailoring the knowledge, leading to new instantiations of design features. Each new feature can be a marked proposition for how core domain knowledge can be reused by students for their designs.
- ***Abstract:*** Because the sharing of fundamental knowledge is imperative, reference claims need to be reused. It is often hard to reuse a claim if the language and scope is too specific, making it more difficult to apply to a different situation. The nomenclature can no longer be informal like the traditional claims. They must achieve a level of language abstraction that will allow students to reuse the reference claim in a different context, but within the same domain.
- ***Artifact Independent*****:** The claim's depicted knowledge does not depend on the use of a specific artifact such as a widget or application. Claims should not specifically describe a design solution as an existing application but rather as a possible design alternative to many possible applications.

These four characteristics aim at improving the knowledge sharing capabilities of claims. Table 4.1 outlines the differences between reference claims and traditional claims.

Reference Claim	Traditional Claim
Applicable: Written by educators experienced in describing the domain's core knowledge.	**Unrestricted:** Any student designer may write the claim without restricting it to a single domain, possibly spanning multiple domains.
Parent: Written such that it can be used as a parent from which traditional claims can be instantiated through specification.	**Child:** Can be derived from one or more reference claims.
Generality: The choice of words and concepts should lean toward making the claim more general, increasing its potential for reuse	**Specificity:** May contain words and concepts that decrease its reusability, but increase design details.
Independent: The claim's depicted knowledge will not depend on the use of a specific artifact such as a widget or application.	**Dependent:** The claim's knowledge can be tied to artifacts, showing examples of possible uses within certain contexts.

Table 4.1: Main characteristics that differentiate reference claims from traditional claims

4.2.2 Example Reference Claim

To illustrate our proposition an example reference claim from the notification systems domain is shown in figure 3.2. First, the example claim depicts knowledge pertinent to the notification systems domain—a method for notifying users of some form of monitored information. Second, the claim can be used as inspiration for more specific claims that are based on the knowledge. By instantiating new design features based on this knowledge, student designers can propose new ways of building a system. Third, the style of its language does not restrict the use of the claim. It is not over specified in any way. Fourth, the knowledge is not explicitly tied to any artifact and therefore, can be used to design any notification system.

Use of Visual Encodings for Notifications

Using visual encodings to represent information in a notification delivery event

+ Decreases interruptions because users are not forced to read textual information

+ Allows for quick recognition and interpretation of visual encodings

+ Can benefit in the delivery of quantitative data

- Requires that the user be familiar with the encoding use for the piece of information

- Requires that there be enough distinction between different types of encodings that are being use together to avoid confusion

Figure 4.2: An example reference claim for the notification system domain.

This example reference claims is used in the following section to describe the advantages of using reference claims over regular claims to share design knowledge among student designing interactive system. Chapter 5 describes a set of reference claims for the domain of notification systems, and Chapter 6 demonstrates how reference claims apply to various notification system domains and can be evaluated to improve the knowledge they encapsulate.

4.2.3 Benefits of Reference Claims as Reusable Knowledge Components

We have seen the main characteristics that differentiate reference claims from traditional claims. These characteristics were developed with the idea to improve the reusability and applicability of reference claims to increase the amount of HCI knowledge used by students in their designs.

4.2.3.1 Domain Applicability

A primary reason for connecting reference claims to a specific domain is to increase their applicability. The example reference claim is related to the notification systems domain because it represents a notification method—an important aspect to consider in notification systems design. Key to the notion of domain-relevancy is the need to understand the domain a reference claim belongs to. Using the taxonomy is extremely important to judging whether a claim is relevant to a domain.

Being created and written by experienced educators, their authors can ensure that the features these claims represent seem viable solutions to students. In our evaluation of the library with traditional claims presented in chapter 3, we noticed that many students hesitated selecting many claims because they did not accurately describe the features they represented. Sometimes the features themselves did not seem to be viable design solutions. All these factors contributed to the limited reusability of traditional claims. Reference claims should be more reusable than traditional claims because the knowledge they represent is central to the domain in which a system is being designed.

4.2.3.2 Instantiation of Regular Claims

Reference claims can positively influence student design by providing impetus for their use—possibly in the form of a traditional claim. Giving designers the ability to consider the tradeoffs for a certain feature allows them to decide whether they would like to make use of the knowledge. If they do wish to, the reference claim can be tailored to suit their own needs. The claim about using visual encodings can be used to create a new claim about the use of visual encodings to deliver information related to online auction bids for example. The new claim is a traditional claim, with the reference claim acting as its parent. Thus, the new claim becomes a representative instance of the use of a reference claim. Taking the example reference claim or any other reference claim and applying it to the problem of auction bids allows students to explore how the knowledge can be used to solve a problem within the domain. Eventually, the design solution that proves to yield the best results will point out which reference claims should be used to solve similar problems. This phenomenon provides impetus behind the need to reuse reference claims. We believe it is harder to create a new claim based on a traditional claim than it is to create one based on a reference claim. The creation of a new claim requires the identification of the true contribution made by a claim. It is laborious to extract such a concept from a traditional claim. By taking the contribution of a reference claim, the process of claim creation is simplified.

4.2.3.3 Generality

Reference claims are written with generality in mind. When choosing the concepts and words that will represent the piece of knowledge, one should try not to restrict the use of the claim. For example, a claim about visual encodings could have been about visually encoding data for online auctions and could have contained upsides and downsides related to bids, restricting its use. One can focus on the true contribution of a reference claim by identifying the central theme within the knowledge. This approach makes it potentially easier for students to understand how the claims can be reused for many systems in many different contexts.

4.2.3.4 Artifact Independency

Tying a claim to a specific artifact can also reduce the reusability of the claim. Our definition of artifact implies the use of widgets or applications. Reference claims can instead concentrate on the behavior of artifacts. One can imagine the use of an icon, a common widget, which changes color to deliver certain information. A claim about the use of visual encodings in an icon constricts the use of the claim to icons. The color changing behavior should be independent of the icon itself, allowing the same behavior to be applied to different artifacts.

The characteristics of reference claims make them an improvement over regular claims for sharing important HCI knowledge among students and benefit their learning of HCI design. By using reference claims as reusable components organized in a taxonomy at the base of a learning environment for HCI design, students learning interface design can easily explore design concerns and design alternatives to support the information gathering stage of constructivist approach to learning. One of goals of this thesis is to evaluate the reusability and accessibility of reference claims (Chapter 6) and to try to gauge the impact of their use during the design process (Chapter 7).

Chapter 5

Creating a Learning Environment with Reference Claims

We have seen that a design taxonomy can outline design concerns and categorize design claims. We have also seen that reference claims can improve over traditional design claims because of their applicability, ability to lead to new features, generality, and artifact independency. A challenge that rises is to create an interactive learning environment based on these two principles to evaluate their benefits for student designers. This chapter outlines the creation of a taxonomy for the domain of notification systems as well as a set of reference claims that combine to form a learning environment for these systems. The design concerns for notification systems revolve around the use of critical parameters, notification system types and notification methods. The set of reference claims organized in the notification system taxonomy can be used as the base to a learning interactive environment for students learning the design of notification systems.

5.1 Creating a Notification System Taxonomy and Reference Claims

As we have seen in chapter 4, the notification system's taxonomy's goal is to outline important design concerns and improve accessibility of design claims for students designing these systems. Based on experience with notification system design and previous literature, our research group outlined what we believe are the important design concerns that should combine to form the taxonomy for notification systems.

Using the four characteristics of reference claims, new claims for the notification systems domain were created with researchers from the Virginia Tech Notification System research group. Each claim consisted of a title, a feature describing the claim, and the upsides and downsides of the use of the feature. Each of us created claims for features with which we had extensive experience to ensure accuracy of the upsides and downsides.

Claims were shared among researcher at a later point in time. This process proved extremely beneficial for several different reasons. Sharing the claims repetitively allowed us to iteratively develop each claim by analyzing the contribution, language, and scope of the claim. Revisions were needed when we believed the reusability of the claim was reduced because of the way the claim was written. This method also allowed us to test each reference claim against their four distinct characteristics. Testing each claim enabled us to evaluate how feasible it is to adhere to the characteristics and how easy it would be to contradict them. Claims were all designed such that they would provide at least one design solution to each of the key levels of the taxonomy. Sections 5.1.1 through 5.1.3 explain the levels of the taxonomy and present a set of reference claims to cover each domain.

5.1.1 Critical Parameters

We previously discussed the IRC framework that defines a scope of notification system designs. The framework proposes that the combination of numerical values for the 3 critical parameters, Interruption, Reaction and Comprehension lead to different notification system designs that have different design goals. As critical parameters, Interruption, Reaction and Comprehension therefore represent important design concerns when creating notification systems. Students must understand what design alternatives they have for each of these concerns and the potential results in terms of conveying information in an appropriate manner. We created the reference claims in table 5.1 for the critical parameter section of the taxonomy. The interruption section is expanded in section 5.2.

Interrupting a primary Task *Interrupting a primary task to divert a user to a secondary notification task.* + Can make the user aware of information they wish to monitor - May inappropriately divert the attention of the user from the primary task to the secondary task even when the user may not wish to be interrupted.
Forcing user reaction to the notification *System notification would require the user to interact with the system when the notification occurs.* + Notification task is attempted promptly - Negative impact on primary task performance - Negative impact on notification task performance - Perceived as highly interruptive
Allowing user to delay reaction to the notification *System notification that would allow the user to attend the notification task at any time after the notification occurs.* + Efficiency of primary task + Accuracy of notification task - Completeness of notification task - Promptness of notification task
Accessing more information related to notifications *Allowing a user to access additional information related to a notification to increase comprehension.* + Allows a user to absorb a notification and increase comprehension of the notification + Can aid in the execution of the primary task - Forces the user to stop the primary task and switch to a different primary task
Relating notifications to preexisting user knowledge *The user is able to relate the current notification to preexisting knowledge he/she had, to update the state of his/her understanding of the notified information.* + Allows user to increase their understanding of information related to the notification + Can potentially aid in the performance of the primary task - The user has to be familiar with the information presented in the notification
Showing trends in information Showing the user trends that are formed over time within the notified information. + Enables the user to understand changes in data over a period of time + Does not force the user to interpret raw data + Summarizes information into a representative form - May not provide access to the actual raw data, leaving the user to trust the system

Table 5.1: Reference claims for critical parameters.

5.1.2 Notification System Types

We have seen that the variation of the numerical values for the three critical parameters creates a design scope for the domain of notification systems. Values range from 0 (low), to 1 (high). The IRC cube defines the major notification system categories (see figure 2.1). Notification systems can therefore be categorized by the different types of design goals determined by the IRC cube. The corners of the IRC cube were used to define the main notification system types. Each notification system type has a distinct set of design concerns. Notification systems types can help students understand major design concerns and alternatives for a wide range of notification systems. We created the following reference claims for each corner of the IRC cube (Table 5.2).

Use of indicator to notify *Using an indicator to notify the user of information.* + Has a reduced amount of interruption to the primary task + Effective in conveying single or very limited information, placing the focus on notifying the user of the most important pieces of information - Limited information can decrease the overall amount of comprehension - May require the user to have a preexisting understanding on the notification information
Monitoring critical activities *Using a notification system to monitor mission critical activities.* + Generates high level of interruption to interrupt the user at all cost. + Provides high level of comprehension necessary for the user to react to the notification in a well informed manner. - Failure of the system may have catastrophic consequences on the primary task.
Determining display location based on user *Choosing the location of a public display based on the locations of users.* + Allows for public notifications to be delivered to the correct group of users based on their location + Can be placed in a position that will decrease the interruptions that it can cause - Certain users may receive more notifications than others because of differing proximities between users and the display
Use of noise for notifications *Use of ambient noise to notify the user of information* + allows ambient audio cues + low interruption - users may not be able to perceive the notification

Table 5.2: Reference claims for notification systems types.

Use of alarm to notify *The use of an alarm to deliver a notification to a user* + Can cause a needed interruption to primary tasks - Delivers very little information, decreasing comprehension and requiring the user to know how to interpret the notification - May lead to an unexpected interruption that was not needed at the notification time
Determining display location based on user *Choosing the location of a public display based on the locations of users* + Allows for public notifications to be delivered to the correct group of users based on their location + Can be placed in a position that will decrease the interruptions that it can cause - Certain users may receive more notifications than others because of differing proximities between users and the display
Public display placed in the periphery *A public display is placed in the periphery to deliver information to users* + Provides a public notification system for groups of users + Allows for groups to monitor similar information to enhance collaborative work - The collective information may not apply to all individual user - The displayed information may be private and therefore may be seen by outsiders - Outsiders may not understand the information displayed
Use of information exhibit *Using an information exhibit to notify users of public information* + Takes advantage of group settings, making notifications to groups of users easier + Typically can carry a higher density of information for notification -The information shown may not apply to all the users - May be harder to interact with a public information exhibit
Use of diversion to notify *Diverting the user's attention away from a primary task to a task that does not require immediate reaction.* + Generates high level of interruption which grabs the users attention + User's willingness to accept an interruption that does not require a reaction can reduce user's stress of having to perform the primary task. -Typically does not provide any information that will aid the user's primary task. -Can lead to increased user frustration if overused or incorrectly used.
Use of secondary displays *The use of a secondary display notification system to notify users* + Decrease interruption by placing visual stimulus in the periphery + Has the potential to deliver more information through multiple visual artifacts, increasing comprehension - Has weaker performance in tasks that require immediate and urgent interruptions

Table 5.2 *continued*: Reference claims for notification systems types.

5.1.3 Notification Methods

For the notification methods we decided to provide students with a set of features for conveying notifications to users. Table 5.3 presents a set of reference claims for the three main output methods used in notification system design.

Using visual encodings *Using visual encodings to represent information in a notification.* + Decreases interruptions because users are not forced to read textual information + Allows for quick interpretation of visual encodings + Can benefit in delivery of quantitative data - Requires that the user be familiar with the encoding - Requires that there be enough distinction between different types of encodings that are being used together to avoid confusion
Using textual information for notification *Using plain text to deliver the information in a notification.* + Increases comprehension with the amount of information that can be delivered + Eases interpretation of the notification - Requires the user to spend more time reading the notification
Fading transitions for notifications *Fading is used to transition from one notification to another.* + Decreases interruptions by reducing the amount of sudden changes on a display + Can serve as a link between information pieces, allowing users to infer connected information - The transition time may be too short, making it similar to a sudden change
Sliding transitions for notifications *A sliding effect is used to transition from one notification to another.* + Can divert the attention of the user to the notification if needed + Can serve as a link between information pieces, allowing users to infer relationships between information pieces - Constant use of sliding transitions can interrupt the user often
Using graphical information for notification Using a graphical representation in a notification to deliver information + Can save space in the design of the notification system + Can facilitate quick recognition of information - Can decrease comprehension if the graphical representation is not understood

Table 5.3 Reference claims for notification methods.

Tactile notifications *Using a tactile feedback mechanism to notify users* + Can provide personal notifications to users such that only the user is aware of the notification + Effective for people with disabilities. Does not rely on sense of sight or sound. - Notification device needs to be in physical contact with the user
Voice notifications *Use of voice for notifications* + Pervasive in nature, useful in eyes busy hands busy situations + Can attract visual attention towards the area where it emanates from. + Users do not need to learn the meaning of the notification as the meaning is directly conveyed. - Process of notification delivery takes longer. - Has high interruption and can be potentially disruptive.
Use of audio for notification *Using audio features for notifications* + efficiency of primary task when it focuses on visual tasks + allows a wide variety of notifications - must be adjusted to the desired level of interruption - may be difficult for user to map the audio to the notification task

Table 5.3 *continued*: Reference claims for notification methods.

5.1.4 Other Notification System Reference Claims

The notification system taxonomy exposes students to important design concerns and the reference claims organized under each design concern describe design alternatives for each one of them. However, other reference claims describing the notification system domain may not particularly fit in any design concern category but still represent core notification system knowledge. We subsequently created a set of reference claims that do not belong under any particular design concern but that can still be very useful to increase student understanding of the notification system domain as well as to expand the set of design alternatives for their systems. These reference claims are presented in table 5.4. Figure 5.1 illustrates the organization of these reference claims in the taxonomy.

Integrating Notifications into the Environment *Notifications are delivered to users by the environment in an off-the-desktop situation* + Allows the notification system to exist off the desktop, not requiring the user to use a computer to be notified + Allows for the system to notify users about information that can aid other primary tasks not related to the computer - Users may not be aware of the presence of the notification system
Notifying Groups of Users *A notification is delivered such that a group of users are to be notified* + Enables groups of users to receive updates at the same time + Can aid in the progress of group tasks where all users need the same information- The information may not apply to every person's own tasks
Continuous Notifications *Use of continuous notifications to constantly notify the user of information regardless of whether it has changed* + Allows users to constantly see the most recent up to date information at any point in time + Allows the user to monitor rapidly changing information - May mean that the user could be interrupted often
Sporadic Notifications *Use of sporadic notifications where notifications are only triggered by an event* + Interruptions only occur when a notification is triggered by an event, limiting the number of interruptions - The user may miss the notification and must wait until the next event
Notification of Changing Information *Delivering a notification to signify a change in monitored information* + Provides the user with information that requested through the system + Can potentially aid the primary task performance - May interrupt the primary task - The user must be made familiar with the meaning of the information or value
Personalizing Notifications *Personalizing notifications to suit the needs of an individual user* + Allows the user to determine how often or when a notification should occur + Allows the user to determine how he/she would like to be notified + Can decrease interruption of the primary task - Requires the system to support various notification methods - The user may unknowingly choose options that will cause them to miss notifications
Notification Systems *The use of a notification system as a secondary task information delivery method* + Permit users to maintain dual-tasks, enabling multitasking situations + Alert users of monitored information as they work on their primary task + Can enhance the performance of their primary task through notifications - Can interrupt the user's primary task at a time they do not want to be interrupted - May be lacking in the information it provides, not allowing the user to fully understand the notification

Table 5.4: Other notification system reference claims.

- **Notification Systems**
  - *Integrating Notifications into the Environment*
  - *Notifying Groups of Users*
  - *Continuous Notifications*
  - *Sporadic Notifications*
  - *Notification of Changing Information*
  - *Personalizing Notifications*
  - **Critical Parameters**
    - **Interruption**
      - *Interrupting A Primary Task*
    - **Reaction**
      - *Forcing User Reaction to the Notification*
      - *Allowing Users to Delay Reaction to the Notification*
    - **Comprehension**
      - *Accessing More Information Related to Notification*
      - *Relating Notifications to Preexisting User Knowledge*
      - *Showing Trends in Information*
  - **Notification System Types**
    - **Ambient**
      - *Determining Display Location Based on User*
    - **Secondary Display**
      - *Use of Secondary Displays*
    - **Noise**
      - *Use of Noise for Notifications*
    - **Critical Activity Monitor**
      - *Monitoring Critical Activities*
    - **Indicator**
      - *Use of Indicator to Notify*
    - **Alarms**
      - *Use of Alarm to Notify*
    - **Diversion**
      - *Use of Diversion to notify*
    - **Information Exhibit**
      - *Determining Display Location Based on User*
      - *Public Display Placed in the Periphery*
      - *Use of Information Exhibit*
  - **Notification methods**
    - **Audio**
      - *Use of Audio for Notification*
      - *Voice Notifications*
    - **Visual**
      - *Using Visual Encodings*
      - *Using Textual Information for Notification*
      - *Fading Transitions for Notifications*
      - *Sliding Transitions for Notifications*
      - *Using Graphical Information for Notification*
    - **Haptic/Tactile**
      - *Tactile Notifications*

Figure 5.1: Notification system taxonomy (bold) and reference claims (italic).

5.2 Completing the Taxonomy: Interruption Taxonomy as Example

The notification taxonomy presented above outlines the key design concerns for the design of notification systems. However, this taxonomy is not a complete one. The taxonomy was developed to a degree where it would be sufficient to demonstrate its utility in outlining key design concerns and categorizing design alternatives. Hence, the current taxonomy outlines the upper levels of the taxonomy, but not necessarily the more specific lower levels. The following section illustrates how one of the most important lower levels of the taxonomy, the interruption taxonomy, can be created to lead to the more complete taxonomy described in Figure 5.2. The same process can be repeated for each underlying level of the taxonomy to increase student understanding of the design domain and design alternatives.

5.2.1 Designing to Interrupt

As we have seen, notification systems are systems that are used in a dual task situation. While performing some primary task, such as writing a word document on a computer desktop, notification systems provide interruption to display some monitored information. In other words, these systems provide transition between a primary task and a secondary task.

The properties of interruption produced by a particular notification certainly should be one of the primary concerns of designers designing notification systems. Different notification systems have different interruption properties. An Alarm system for example is meant to be very interruptive, forcing you to abandon your primary task focus and attend some important notification. Other systems such as email programs may not want to completely shift your focus from your primary task each time you receive an email. Notification systems can then be classified according to the properties of the interruption they want their notifications to produce.

We have seen that McCrickard and Chewar (2003) determined that interruption should be a critical parameter for the design of notification systems. However, we have seen from chapter 3 that determining which design choices lead to different interruption levels is not an easy task. If we can determine the properties of interruption associated to different systems and stimuli, we can then build our systems so that they have the effect on a user that we intend them to have. Too many systems today have an important annoyance factor because they interrupt the user unnecessarily. Others do not provide enough interruption and may not be able to inform of critical information. If we could determine adequate interruption properties we could both reduce this annoyance factor and make sure critical information is perceived by the user.

Using reference claims to encapsulate the properties of interruption could guide student designers through a design process and help them achieve the desired interruption properties for their notification systems. There are numerous advantages to this approach compared to a standard design approach followed by empirical testing. If the desired interruption properties can be determined before the design phase, and if the design phase can target desired interruption properties at the beginning of the design, problems associated with interruption can be discovered early in the development process. Exposing student designers to reusable interruption design knowledge will not only help them design faster by informing them to critical interruption properties they need to consider when designing, but it will also save them a lot of time by avoiding many problems related to interruption that may only be discovered through long user studies.

5.2.2 Encapsulating Interruption Knowledge into Reference Claims

The domain of human interruption has been widely studied over the years and many theories, guidelines and properties for human interruption have emerged. Three main design properties with design alternatives that yielded convincing empirical results seem to apply to the design of interactive systems.

5.2.2.1 Coordination of Interruption

McFarlane identified four methods to interrupt a user from a primary task: immediate, negotiated, mediated, and scheduled (McFarlane, 2002). To describe these solutions let’s take the example of a user completing some primary task on a desktop computer which has a software to monitor incoming mail. To inform the user of the content of any incoming mail, the email agent needs to interrupt him from his primary task. The email software would have four distinct coordination options to interrupt the user and show him the new email:

- ***Immediate:*** the full email is immediately displayed as it arrives in the inbox.
- ***Negotiated:*** user is informed of an incoming email but he can access the full content at any given time.
- ***Mediated:*** the full email is displayed when the software estimates it is the best moment to interrupt the user.
- ***Scheduled:*** new emails are displayed at consistent time intervals (i.e. every 20 seconds).

McFarlane tested these coordination methods. Participants were asked to play a simple video game that would serve as the primary task. While they would play the game, interruptions would require them to complete a simple secondary task such as adding two numbers. From the results of his experiment, reference claims were created that capture the tradeoffs of each coordination method. The claims are presented in table 5.5.

Immediate interruption for notifications *Interrupting user from primary task with a notification that would insist that the user immediately attends the notification and performs the notification task.* + Notification task is attended promptly + Effective completion of notification task - Negative impact on primary task performance
Negotiated interruption for notifications *Interrupting user from primary task with a notification that would allow the user to control when to attend the notification task.* + Efficiency on primary task + Efficiency on notification task + Method usually preferred by users + Perceived as low interruption - Negative impact on completeness of notification task - Negative impact on promptness of notification task

Table 5.5: Reference claims for the coordination of interruption.

Mediated interruption for notifications. *Interrupting user from primary task with a notification that would interrupt the user at an estimated least interruptive moment.* + Efficiency on primary task + Completeness of notification task + Method usually preferred by users + Perceived as low interruption - Slight negative impact on promptness of notification task
Scheduled interruption for notifications. *Interrupting user from primary task with a notification that would interrupt the user based on a pre-arranged schedule (i.e. every 30 seconds).* + Predictability of notification task - Negative impact on efficiency of primary task - Negative impact on completeness of notification task - Negative impact on promptness of notification task - Perceived as highly interruptive

Table 5.5 *continued*: Reference claims for the coordination of interruption.

5.2.2.2 Interruption Phase

Franke, Daniels, and McFarlane (2002) studied recovering context after interruption. They state that it is useful to divide user interruption approaches for into three main phases:

- **Pre-interruption phase**: the pre-interruption phase prepares the user for the transition from the primary task to the secondary task.
- **Mid-interruption phase**: the mid-interruption phase generally focuses on the user's transition to the secondary task. It involves user's efforts and ability to maintain awareness of the primary task while working on the secondary task.
- **Post-interruption Phase**: the post-interruption phase sees the user return and refocus to the context of the primary task that was interrupted.

They studied the effects of interrupting during these interruption phases and suggested that we design specific interruptions that would reduce the negative impacts of interrupting users. Cutrell, Czerwinski and Horvitz (2001) also studied the impacts of interruption phases and concluded that reminders of the interrupted task help recover from interruption. From the results of their experiments, the reference claims presented in table 5.6 were created.

Pre-Interruptive interruption *Notification that takes place before the main notification. It prepares the user to transition from the main task to the notification task.* + User can differentiate between the main task and interrupting task contexts + Allows the user to rehearse the context of the primary task before continuing into the notification task - Adds an additional notification for the notification task
Post-Interruptive interruption *Notification that takes place after the main notification. It prepares the user to recover from the notification and focus on the primary task.* + User can differentiate between the main task and interrupting task contexts + Allows the user to regain the context of the primary task - Adds an additional notification for the notification task

Table 5.6: Interruption phase reference claims

5.2.2.3 Mode of Interruption

Arroyo, Selker and Stoufs (2002), studied multi-modal outputs in order to discover which ones where the most disruptive. They tested 5 types of interruption based on the 5 senses:

- ***Smell:*** atomizer directed from behind a wall towards the subjects.
- ***Vibration:*** vibrating device placed in the chair on which the subject sat.
- ***Sound:*** phone ring sound file played by a computer.
- ***Heat:*** room temperature.
- ***Light:*** spotlight controlled by a potentiometer.

The results of their experiment outlined that although personal preferences led to people being affected differently by the various modal outputs, there is nevertheless a general trend in the disruptive properties of these modes of interruption. In this case, the least used modalities in user interfaces such as smell have bigger disruptive effects than more common modalities based on vision such as light. Their results were encapsulated in the reference claims presented in table 5.7.

Visual interruption *Interrupting user from primary task with a visual notification.* + Less disruptive than Audio, Vibration-based, Heat-based and Smell-based interruptions + Visual configuration may lead to a high comprehension of the notification + Usually disrupts only the user - Information overload if primary task involves visual information
Audio interruption *Interrupting user from primary task with an audio notification.* + Audio configuration can lead to a high comprehension of the notification + less disruptive than Vibration-based and Smell-based interruptions - More disruptive than Heat-based and visual interruptions - Information overload if primary task involves audio information - Can disrupt people other than the user
Heat-based interruption *Interrupting user from primary task with a heat-based notification.* + Less disruptive than Audio, Vibration-based and Smell-based interruptions + Disrupts only the user - More disruptive than Visual interruptions - Generally low comprehension of notification
Vibration-based interruption *Interrupting user from primary task with a vibration-based notification.* + Less disruptive than smell, interruptions + Disrupts only the user - More disruptive than Visual, Vibration-based and Smell-based interruptions - Generally low comprehension of notification - Information overload if the primary task contains vibration-based information
Smell-based interruption *Interrupting user from primary task with a smell-based notification.* + Can be used as an additional communication channel when all others are over-loaded - More disruptive than Visual, Vibration-based, Smell-based and audio interruptions - *Generally low comprehension of notification* - *Can disrupt people other than the user*

Table 5.7: Mode of interruption reference claims.

5.2.3 Expanded Taxonomy

By examining and encapsulating existing interruption knowledge we were able to create a more complete notification system taxonomy. Similarly, other domains within the notification system domain can be expanded to improve the coverage of important HCI knowledge. The Notification Methods/Audio/Voice category shown in Figure 5.2 is yet another example of how domains within the taxonomy can be expanded to provide a more complete design scope for the design of notification systems.

- **Notification Systems**
  [..]
  - **Critical Parameters**
    - **Interruption**
      Interrupting A Primary Task
      - **Interruption Coordination**
        Immediate interruption for notifications
        Negotiated interruption for notifications
        Mediated interruption for notifications
        Scheduled interruption for notifications
      - **Interruption Mode**
        Visual Interruption
        Audio Interruption
        Vibration-based Interruption
        Smell-based Interruption
        Heat-based Interruption.
      - **Interruption Phase**
        Pre-Interruptive interruption
        Post-Interruptive interruption
    - **Reaction**
      Forcing user reaction to the notification
      Allowing user to delay reaction to the notification
    - **Comprehension**
      Accessing More Information Related to Notification
      Relating Notifications to Preexisting User Knowledge
      Showing Trends in Information
  - **Notification System Types**
    […]
  - **Notification methods**
    - **Audio**
      Use of audio for notification
      - **Voice**
        Voice Notifications
        - **Voice Notification Enhancement**
          Audio Cues mixed with Voice Notifications
          Prolonged Notification delivery
          Emotional cues present in Voice Notifications
        - **Voice Delivery Methods**
          Recorded Human Voice
          Using a Familiar Voice for Notifications
          Using synthesized voice for Voice Notification
          Mixing synthesized and recorded voice
    - **Visual**
      Using Visual Encodings
      Using Textual Information for Notification
      Fading Transitions for Notifications [5]
      Sliding Transitions for Notifications
      Using Graphical Information for Notification
    - **Haptic/Tactile**
      Tactile Notifications

Figure 5.2: Expanded notification system taxonomy and reference claims.

Chapter 6

Validating the Applicability and Reusability of Reference Claim Knowledge*

We have seen how we could create an interactive learning environment based on a taxonomic organization of reference claims. These reference claims were created such that the knowledge they encapsulate is applicable and reusable for a wide variety of notification systems designed for different usage contexts. This chapter exemplifies how a set of reference claims written with results from a particular context can be reused in a completely different context and still be applicable. This chapter also describes how an evaluation of designs based on reference claims may lead to an improvement of the applicability and reusability of the knowledge encapsulated into reference claims.

6.1 Applying Reference Claims to Different Domains

One of the major characteristics of reference claims is that they must achieve a level of generality that will allow students to reuse them in a different context, but within the same

*Sections of this chapter are covered in Sabri et al., 2006 domain. This section describes how a set of reference claims can be applied to a particular problem within the specific context of large high-resolution displays.

6.1.1 Design Problem

Large, high resolution displays introduce new challenges to the design of interactive systems. As these screens allow users to have an unprecedented amount of detailed information in their field of view, awareness of peripheral information becomes difficult.

The application Wargus demonstrates these challenges. The game is a complete clone of Warcraft II, originally developed by Blizzard. The dynamic geospatial environment that Wargus provides requires the user to focus his attention to all areas of the display in order to win. Figure 6.1 is a snapshot of the game on a standard 1024x768 screen. Important information necessary to perform well such as the map, gold, wood and gas amount are displayed at the top of screen. The size of the screen enables users to easily track changes in the resource levels at any time during the game.

Figure 6.2 is a snapshot of Wargus displayed on a large high resolution display. The game resolution was increased such that users had a larger playing field rather than larger graphics. Each individual screen displays approximately the same amount of information as a single screen would on the game at the standard resolution.

Figure 6.1: The game Wargus on a standard monitor.

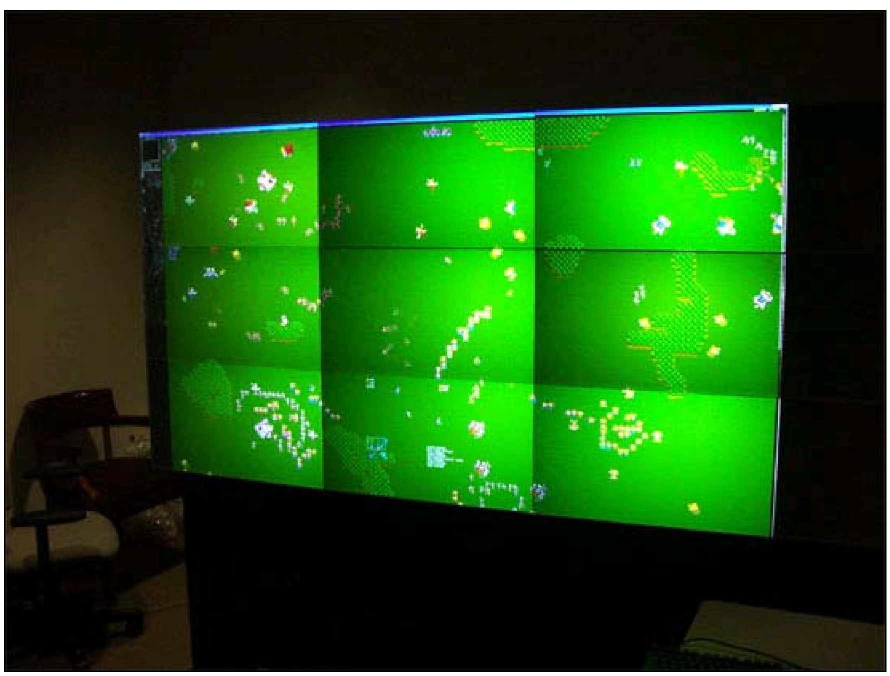


Figure 6.2: Wargus displayed on a large high resolution display.

Previous studies demonstrated that large high resolution displays improve the game task by enabling a larger and more detailed playing field (Sabri et al., 2006). However, they also limited awareness and visibility of important information such as resources placed in the periphery. The main problem lies around maintaining awareness of peripheral information without inhibiting the advantages of large high resolution displays.

6.1.2 Reference Claims to Create New Design Solutions

To solve the problem of peripheral awareness, notifications can be designed to notify users of important information that is usually placed in the periphery. The notification system taxonomy defined in Chapter 5 outlines a set of possible design choices. Rather than redesigning the current peripheral information, focus turned to bringing the current peripheral information in the user's field of focus. The coordination of interruption section of the taxonomy outlines four different methods to interrupt the user with system notifications. Four designs were created based on each one of the coordination of interruption reference claims:

- **Immediate interruption for notifications.**
  Interrupting user from primary task with a notification that would insist that the user immediately attends the notification and performs the notification task.
- **Negotiated interruption for notifications.**
  Interrupting user from primary task with a notification that would allow the user to control when to attend the notification task.
- **Mediated interruption for notifications.**
  Interrupting user from primary task with a notification that would interrupt the user at an estimated least interruptive moment.
- **Scheduled interruption for notifications.**
  Interrupting user from primary task with a notification that would interrupt the user based on a pre-arranged schedule (i.e. every 30 seconds).

6.1.2.1 Immediate Design

Interrupting user from primary task with a notification that would insist that the user immediately attends the notification and performs the notification task.

The primary task in this situation is playing the game. The secondary task is monitoring resources such as gold and wood. Since resources constantly change, an immediate design would insist that the user constantly monitors the resource levels. This solution was implemented in an immediate design for Wargus where peripheral resources are continuously displayed in the user's area of focus. Figure 6.3 shows the resources centered around the cursor. All the peripheral information (e.g. gold, wood, etc.) follows the cursor at all times and is therefore always in the user's area of focus. This design ensures the user's constant awareness of his secondary information.

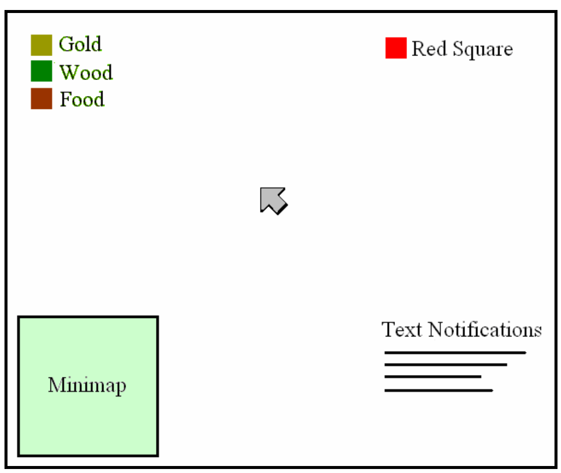


Figure 6.3: Immediate design for Wargus. The resources are centered around the cursor.

6.1.2.2 Negotiated Design

Interrupting user from primary task with a notification that would allow the user to control when to attend the notification task.

In this design, the game information is brought to the cursor on-demand (see Figure 6.4). By default, the information is located at the top-left edge of the screen. However, users can have the game information displayed around the cursor at any time during the game by pressing the mouse's wheel button. The user can then send the information back to the default area of the screen by pressing the mouse-wheel button again.

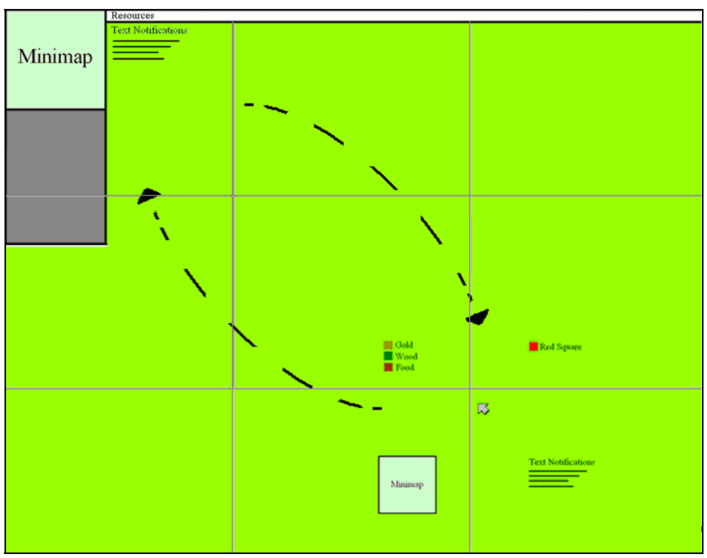


Figure 6.4: A diagram demonstrating how notifications are brought around the cursor.

6.1.2.3 Mediated Design

Interrupting user from primary task with a notification that would interrupt the user at an estimated least interruptive moment.

In this solution, the game information is displayed in the periphery by default. However, a simple computer algorithm constantly evaluates the best moment to interrupt the user based on their primary task status and their resource levels. Information appears in the user's area of focus when the computer determines it is necessary. For instance, gold resources will appear in the user's focus area when the gold level is determined too low.

6.1.2.4 Scheduled Design

Interrupting user from primary task with a notification that would interrupt the user based on a pre-arranged schedule (i.e. every 30 seconds).

In this solution, the game information will be displayed in the periphery by default. However, the game information will appear in the user's area of focus every 30 seconds for 10 seconds, and then move back to the periphery. This allows the user to be interrupted to briefly attend their secondary task status for short periods of time.

6.1.3 Making a Design Choice based on Reference Claim Trade-offs

We have seen how four reference claims for the coordination of interruption can be used to create four new designs in a very specific sub-domain of notification systems. From the set of trade-offs students should be able to select which design options suits their need the best. The question of the portability of the claim trade-offs from one domain to another may rise. Do the tradeoffs of reference claims really work for any notification system design? Do the particularities of certain contexts add new advantages or disadvantages to the use of certain reference claims?

The following section tries to answer these questions by evaluating these four designs to investigate how the tradeoffs of reference claims can apply to different notification system designs used in different contexts.

6.2 Evaluating Reference Claim Designs

To assess the tradeoffs of reference claims, an experiment was set-up to evaluate and compare the four different reference claim designs from section 6.1. From the results of the experiment, conclusions on the upsides and downsides of each design solution can be drawn and compared to the tradeoffs associated with the original reference claims.

6.2.1 Experimental Set-up

This experiment was designed to evaluate user performance on all 5 Wargus designs (4 reference claim designs and control design). Participants were asked to play the game on a large high resolution display against a computer opponent. Each participant played a total of five games using a different notification design for each game. Participants were given a 30 minute practice round where they could learn all five designs and re-familiarize themselves with the game. Each participant played all five designs in differing orders using a Latin square design. A total of 125 games were played.

To more accurately measure user awareness of peripheral information, an additional white square next to the game resources was added. Users were asked to press the space bar each time they saw the square's color change to red. The colored square followed the same behaviors as other resources in all conditions and was just described as an additional resource in the game. In the mediated solution the square would appear next to the game cursor each time it turned red. The red square appeared at random intervals a total of ten times each game. For each design condition, the following in-game data was tracked and stored for analysis (see Table 6.1):

Game performance (score, game time, number of units killed)
Resource monitoring performance (resource level maintenance)
Red square monitoring performance (square detection)

Table 6.1: List of dependent variables for the game study.

Each game lasted between 10 and 20 minutes with the total time of the experiment lasting approximately two hours for each participant. Users were given written instructions detailing common hot-keys and keyboard shortcuts used to build units and give commands. At the end of the experiment participants were asked to respond to a questionnaire. The questionnaire consisted of open-ended questions and seven-point Likert scale question (1 to 7) that they used to rate each design individually.

Twenty-five expert participants with over 100 hours of experience playing Warcraft® 2 or a similar real-time strategy game were recruited for the experiment. There were 24 male participants and one female participant, all with ages between 20 and 23. Most of them had little familiarity with multi-monitor desktops.

The five different versions of Wargus ran on a display which used a 3x3 matrix of high definition rear projectors. The display provides a virtually seamless screen at a resolution of 3840x2160 and runs on a high-performance Dell server configured with five dual-head graphics cards (Figure 6.5).

Figure 6.5: Wargus running on a rear-projection display at a resolution of 3840x2160.

The original version of Wargus with unaltered notifications served as a control to discover whether any of the notification methods improved the original design. The control version was referred to as the *Standard* design. The game ran at a resolution of 3840x2160 using OpenGL to perform rendering across the five graphics cards. As a result, peripheral information such as gold and wood were more distant from the user's focus area and more difficult to access visually than on a standard single screen.

6.2.2 Quantitative Results

This section highlights how the four reference claim designs compared to each other based on secondary and primary task performance. The secondary task was the notification task that consisted in monitoring the change in color of the square resource. The primary task consisted in the user playing the game and trying to beat the computer opponents.

6.2.2.1 Secondary Task Promptness

During the experiment, participants were asked to monitor a white square that would randomly turn red near the other game notifications (Figure 6.3). As such, the average response time for each design was tracked and analyzed. A standard ANOVA test suggests that there is a significant difference in the mean reaction time among the different designs ($F(4,120)=19.41$, $MSE=1.577$, $p<0.0001$). The designs that kept the critical game information in the user's focus at all times had much faster reaction times when detecting changes in the square color.

Specifically, the mean reaction time for the Mediated design ($M=1.68$, $SD=0.74$) was significantly faster than the reaction times for the Control ($t(48)=5.49$, $p<0.0001$), Negotiated ($t(48)=4.64$, $p<0.0001$) and Scheduled ($t(48)=7.5$, $p<0.0001$) designs. Therefore, isolating the critical data and bringing it into the focus greatly helps improve users' reaction time.

The mean reaction time for the Immediate design ($M=1.83$, $SD=1.004$) was also significantly faster than the reaction times for the Control ($M=3.41$, $SD=1.38$) ($t(48)=4.61$, $p<0.0001$), Negotiated ($M=3.14$, $SD=1.38$) ($t(48)=3.83$, $p<0.0004$) and Scheduled ($M=4.3$, $SD=1.57$) ($t(48)=6.6$, $p<0.0001$) designs. Though it is not significantly different from the Mediated design, it shares the same benefits in reaction time, even while presenting other non-critical information in the focus simultaneously.

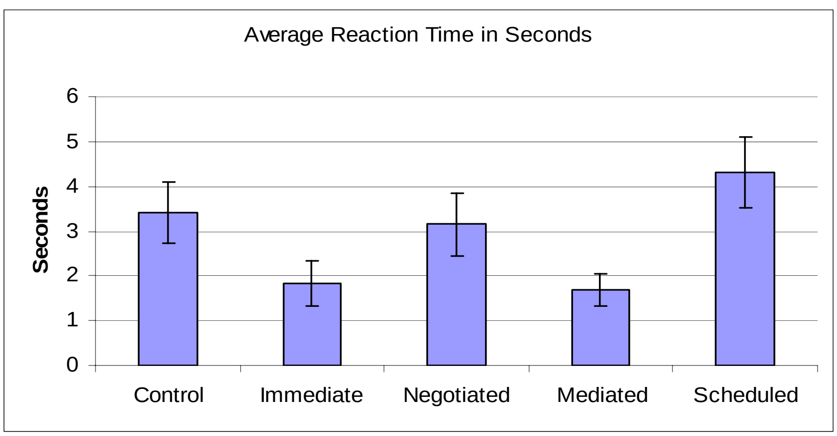


Figure 6.6: Average notification reaction time.

6.2.2.2 Secondary Task Completeness and Efficiency

Each game had ten red squares randomly appear during the game. The game automatically kept track of how often users successfully detected or missed the change in color of the square. Figure 6.7 and 6.7 detail the results. An ANOVA test shows significant differences in the average success rate for detecting the red square ($F(4,120)= 8.23$, $MSE=4.968$, $p<0.0001$).

The mean detection rate for the Mediated design ($M=9.36$, $SD=2.07$) was significantly greater than the detection rates for Standard ($M=6.64$, $SD=2.64$) ($t(48)=-4.04$, $p<0.0002$), Negotiated ($M=7.24$, $SD=2.33$) ($t(48)=-3.39$, $p<0.002$) and Scheduled ($M=7.44$, $SD=1.98$) ($t(48)=-3.34$, $p<0.002$) designs. As with reaction time, the Mediated design allowed participants to successfully detect more squares by constantly keeping the critical data around the mouse than the designs that did not.

Just like the Mediated design, the mean detection rate for the Immediate design ($M=9.4$, $SD=2.04$) was significantly greater than the detection rates for Standard ($M=6.64$, $SD=2.64$) ($t(48)=-4.13$, $p< 0.0002$), Negotiated ($M=7.24$, $SD=2.33$) ($t(48)=-3.48$, $p< 0.002$) and Scheduled ($M=7.44$, $SD=1.98$) ($t(48)=-3.44$, $p< 0.002$) designs. Participants using both the Mediated and Immediate designs missed less than one red square on average. Just as with the reaction times, this shows that critical information can be shown along with passive game information without degradation in awareness.

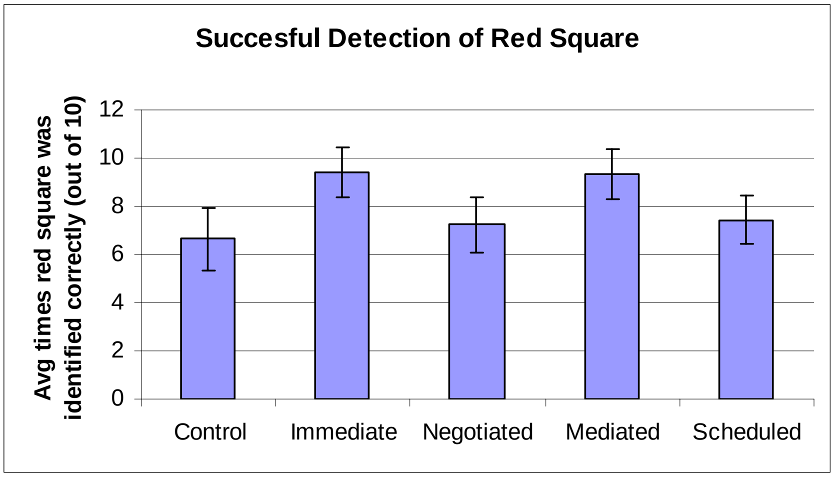


Figure 6.7: Average number of successful notification detections.

The game also tracked the number of times users incorrectly hit the space bar when they thought they were being notified by the red square (see Figure 6.8). ANOVA tests show significant differences in the number of falsely identified red squares ($F(4,120)= 8.23$, MSE=2.19, $p<0.05$). The number of false hits for the Negotiated design (M=1.76, SD=1.76) was significantly greater than the Immediate (M=0.6, SD=1.11) ($t(48)=-2.77$, $p<0.008$) and Mediated designs (M=0.64, SD=1.11)($t(48)=-2.68$, $p<0.01$). This indicates that giving users control of when the notification is put in the focus or the periphery creates an interesting issue. When compared to keeping critical information in the focus at all times, it can significantly cause users to erroneously detect notifications during the game.

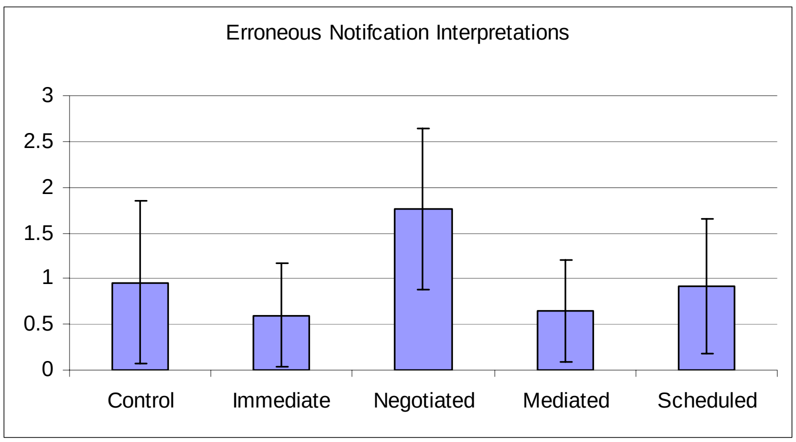


Figure 6.8: Average number of times users inaccurately detected the notification.

6.2.2.3 Primary Task Performance

Additionally, there was no statistical significant difference in regards to whether the four designs affected user score or other performance evaluations. While this does not demonstrate that any of the designs improve performance, it can state that notifying the users near the cursor does not negatively impact game performance with any significance. However, qualitative results show that two of these methods do enhance the user experience on high-resolution displays.

6.2.3 Qualitative Results

This section highlights the participants' reactions to the four designs in comparison to each other and the original game. The results are drawn from the questionnaire they answered at the end of the experiment which consisted of open ended questions and seven point Likert-scale questions.

6.2.3.1 Immediate Design Feedback

All of the participants in the study liked how the information was always readily available without requiring them to divide their attention between different areas of the screen. They claimed that the design was consistent and reliable, so they always knew where to access the information they needed. The participants also claimed that the constant movement with the cursor was initially distracting to the game-play, but this technique helped more than harmed. Users responded to this design in the following way: 93% felt it improved performance compared to the standard design, 20% thought it was the best for detecting the square, and 24% preferred this design over the others.

6.2.3.2 Negotiated Design Feedback

For this design, participants liked how they could have the information readily available on-demand. Since the game information wasn't always around the cursor, it wasn't as distracting as the immediate design. However, since the game data was not always in the user's focus, they commented that they often missed vital information. Users responded to this design in the following way: 72% felt it improved performance compared to standard design, 8% thought it was the best for detecting the square, and 20% preferred this design over the others.

6.2.3.3 Mediated Design Feedback

By only showing pertinent information, users claimed it was easy to discern the information that was critical. The design also kept the information around the mouse uncluttered and it was much less distracting. Most participants also liked that the information would go away without any form of interaction. Users responded to this design in the following way: 76% felt it improved performance compared to standard design, 72% thought it was the best for detecting the square, and 56% preferred this design over the others.

6.2.3.4 Scheduled Design Feedback

The scheduled design divided their attention, and was highly interruptive at inappropriate points during the game. The design also caused them to miss critical information for the periods when it was not mapped to the cursor. Users responded to this design in the following way: 20% felt it improved performance compared to standard design, 0% thought it was the best for detecting the square, and 0% preferred this design over the others.

6.2.3.5 Level of Interruption

With the feedback received from the questionnaire, we were able to successfully evaluate how much each notification design affected the user's experience playing the game using a seven-point Likert scale. We found that there were no significant differences in how much the user perceived to be interrupted from their primary task with the *Standard*, *Immediate*, *Negotiated*, and *Mediated* designs (Figure 6.9).

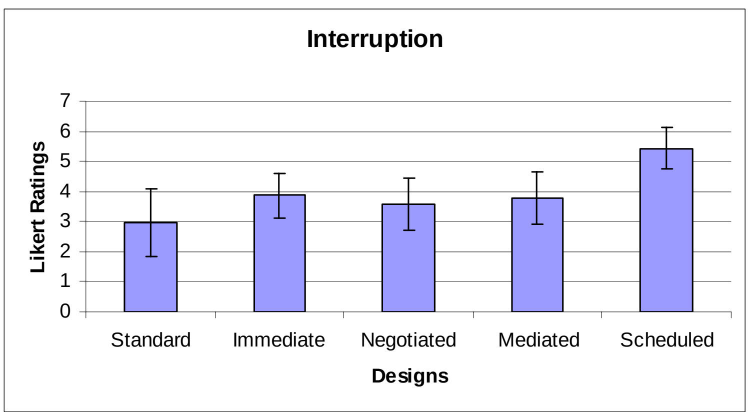


Figure 6.9: User perceived interruption level.

However, ANOVA tests of the user interruption ratings for all designs suggests that there is a significant difference (F(4,120)= 7.07, MSE=2.98, p<0.0001). User's found the Scheduled (M=5.44, SD=1.35) design more interruptive than the Standard (M=2.96, SD=2.2) (t(48)=-4.78, p<0.0001), Immediate (M=3.88, SD=1.48) (t(48)=-3.88, p<0.0004), Negotiated (M=3.56, SD=1.73) (t(48)=-4.26, p<0.0001) and Mediated (M=3.8, SD=1.73) (t(48)=-3.72, p<0.0006) designs. Users found that this design would intrude on their area of focus at inappropriate times and it severely interrupted them from their primary task of playing the game. The other designs were more subtle in their approach to notifying the user, and were not much more interruptive than the *Standard* condition. Due to this low-level of interruption, they were able to play the game without a substantial increase in interruption.

6.2.3.6 User Ratings

Users were also asked to rate each design for their overall preference using the same scale (see Figure 18). A standard ANOVA test suggests significant differences in the users' overall rating of the different designs ($F(4,120)= 18.53$, $MSE=1.82$, $p<0.0001$). The Standard design ($M=3.08$, $SD=1.03$) was rated significantly lower than the Immediate ($M=5.28$, $SD=0.98$) ($t(48) = -7.7$, $p<0.0001$), Negotiated ($M=4.64$, $SD=1.49$) ($t(48)=-4.28$, $p<0.0001$) and Mediated designs ($M=5.12$, $SD=1.83$) ($t(48)=-4.84$, $p<0.0001$). With too little interruption in the Standard design, the users were unable to maintain comprehension of the peripheral information and it forced them to divide their attention away from the game.

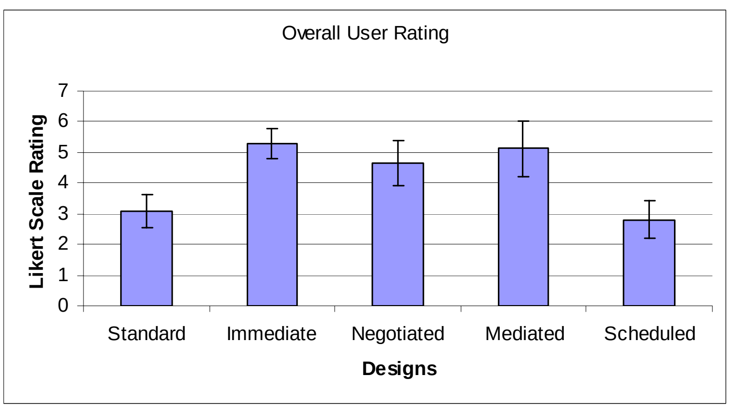


Figure 6.10: User design preference based on ratings.

The Scheduled design ($M=2.8$, $SD=1.22$) was also rated significantly lower than the Immediate ($t(48)=-7.9$, $p<0.0001$), Negotiated ($t(48)=-4.75$, $p<0.0001$) and Mediated ($t(48)=-5.26$, $p<0.0001$) designs. With too much interruption in the Scheduled design, users were distracted away from their primary task. Thus, by bringing the game information to the mouse without being too interruptive, we were able to increase the user's preference of this design compared to the Standard design.

6.3 Adjusting Reference Claim Knowledge

We have seen that the evaluation of reference claim designs led to the discovery of numerous tradeoffs associated with their use. This section investigates how the HCI knowledge encapsulated into the interruption reference claims also applies to the specific sub-domain of the experiment in this chapter. These additional results can also lead to an adjustment of reference claim knowledge to ameliorate their applicability.

6.3.1 Differences in Design Contexts

There are differences between the context of the experiment from which the coordination of interruption reference claims were drawn and the context of the experiment described in this chapter. These differences may lead to uncertainty of the accuracy of the tradeoffs associated with the reuse of these reference claims.

Reference claims were drawn from McFarlane's experiment on the coordination of interruption (2002). In this experiment, users were playing a computer video game which required attention, coordination and reflexes to complete the game task. Punctual notifications required them to complete a secondary task consisting of matching shapes and evaluating the results of an addition.

The first main difference with our current experiment is that although the primary task can be considered similar, the secondary task has important different characteristics. First of all, the system notifications are continuous. The resources change constantly, and the secondary task consists in monitoring their levels. However, in McFarlane's design, users alternated phases where they either attended the primary task or the secondary task. The continuous nature of the notifications in our design, forces the user to constantly monitor the resource levels as s/he is playing the game and as such, the primary task and secondary task are synchronous rather than asynchronous.

The other main difference is the system's display. In McFarlane's experiment, users were performing their activities on a standard single-screen, monitor, limiting their primary task field of view and need for peripheral awareness. Our experiment uses a large high resolution display that requires the user to focus on multiple areas of the screen as well as maintain distant peripheral awareness. The large display reinforces the need for interruption from the primary task to adequately perform the secondary task.

These differences may lead in tradeoff differences associated with the use of each design solution. However, as we shall see, most tradeoffs remain applicable throughout the context differences.

6.3.2 Immediate Interruption Reference Claim Revisited

Results from section 6.2.2.1 and 6.2.2.2 reinforced the positive impact of the immediate interruption solution on notification task promptness and completion. These upsides from the original reference claim therefore hold in the new context.

Participants also claimed that the constant movement with the cursor was initially distracting to the game-play, but that this technique helped more than harmed. Therefore the original downside of this claim which stated the negative impact of this interruption method on the primary task, only partially holds in the new context. Table 6.2 illustrates the adjustment of this reference claim to take into account these new results.

Immediate interruption for notifications		
Upside/Downside	Holds?	Adjustment
+Notification task is attended promptly	Yes	None
+ Effective completion of notification task	Yes	None
- Negative impact on primary task performance	Partially	- Slight negative impact for monitoring secondary tasks, higher negative impact otherwise.

Table 6.2: Immediate interruption reference claim adjustment.

6.3.3 Negotiated Interruption Reference Claim Revisited

Results from section 6.2.2.1 and 6.2.2.2 reinforced the negative impact of the negotiated interruption solution on notification task promptness and completion compared to the immediate and mediated solutions. These downsides from the original reference claim therefore hold in the new context.

Participants found that since the game information wasn't always around the cursor, it wasn't as distracting as other designs and helped them focus on their primary task. Therefore the tradeoffs associated with primary task efficiency and interruption perception hold in the new context.

However, secondary task efficiency varies. As continuous notification monitoring requires users to constantly evaluate the notifications, a negotiated approach that tends to hide the notifications for long periods of time where the user wants to focus on his primary task, hinders performance on the secondary task. Table 6.3 illustrates the adjustment of this reference claim to take into account these new results.

Negotiated interruption for notifications		
Upside/Downside	Holds?	Adjustment
+ Efficiency of primary task	Yes	None
+ Efficiency of secondary task	No	+ Efficiency of secondary tasks that do not consist in monitoring continuous information - Negative impact on efficiency of monitoring-type secondary tasks.
+ Method usually preferred by users	Yes	None
+ Perceived as low interruption	Yes	None
- Negative impact on completeness of notification task	Yes	None
- Negative impact on promptness of notification task	Yes	None

Table 6.3: Negotiated interruption reference claim adjustment.

6.3.4 Mediated Interruption Reference Claim Revisited

Results from section 6.2.2.2 also reinforce the positive impact of the mediated interruption solution on notification task efficiency. This upside from the original reference claim therefore holds in the new context.

Conclusions on primary task efficiency could not be made (see section 6.2.2.3), however, user feed back showed that this method was not considered very interruptive and therefore benefited their primary task performance. More than half the users also mentioned this was their favorite design. Therefore, the upsides related to primary task efficiency, preference, and low interruption hold in the new context.

However, secondary task promptness impacts vary. As our experiment required users to constantly monitor information at relatively short intervals whereas McFarlane's experiment required users to attend a secondary task after much larger time intervals, a mediated approach seems best suited to continuous notifications in terms of notification task promptness. Table 6.4 illustrates the adjustment of this reference claim to take into account these new results.

Mediated interruption for notifications		
Upside/Downside	Holds?	Adjustment
+ Efficiency on primary task	Yes	None
+ Completeness of notification task	Yes	None
+ Method usually preferred by users	Yes	None
+ Perceived as low interruption	Yes	None
- Slight negative impact on promptness of notification task.	Partially	- Slight negative impact on promptness of notification task other than monitoring continuous notifications. + Promptness of monitoring secondary tasks.

Table 6.4: Mediated interruption reference claim adjustment.

6.3.5 Scheduled Interruption Reference Claim Revisited

Predictability of the notification task in this design was not discovered, and therefore nothing enables us to reject the claims upside on notification task predictability.

Results from section 6.2.2.1 and 6.2.2.2 reinforced the negative impact of the scheduled interruption solution on notification task promptness and completion compared to the immediate and mediated solutions. These downsides from the original reference claim therefore hold in the new context.

The perceived high interruption was also found in the results but additional results from section 6.2.3.6 provide an additional drawback of this solution in terms of user preference. Table 6.5 illustrates the adjustment of this reference claim to take into account these new results.

Scheduled interruption for notifications		
Upside/Downside	Holds?	Adjustment
+ Predictability of notification task	Yes	None
- Negative impact on efficiency of primary task	Yes	None
- Negative impact on completeness of notification task	Yes	None
- Negative impact on promptness of notification task	Yes	None
- Perceived as highly interruptive	Yes	- Perceived as highly interruptive - Method least preferred by users

Table 6.5: Scheduled interruption reference claim adjustment.

6.4 Reusability of Reference Claim Knowledge

The results of the experiment demonstrate that when properly implemented, the four characteristics defined in chapter 4 make reference claims easily applicable to a variety of contexts within a domain. Testing and adjustment of reference claims can also contribute to a growing body of more accurate domain knowledge.

Traditional claims seem applicable to only certain contexts which add uncertainties to the impacts of their use in other situations. Their dependency on the context from which they were created often limits their reuse for other environments as well as students' understanding of the HCI concepts behind them. On the contrary, the applicability of reference claims, and the accuracy of their tradeoffs throughout different contexts should improve students' reuse and understanding of the HCI knowledge that is captured in their tradeoffs. Reference claims therefore seem better suited to support the information gathering stage of a constructivist learning approach for students designing interactive systems. Chapter 7 investigates the potential benefits of the use of reference claims at the base of an interactive learning environment for students learning HCI design.

Chapter 7

Evaluating a Learning Environment Based on a Taxonomic Organization of Reference Claims*

We have seen that an evaluation of reference claims shows their potential in terms of applicability and reusability and can lead to an adjustment of reference claim knowledge. This chapter evaluates the benefits of a learning environment based on reference claims organized in a taxonomy through three main points of interest. First, evaluate how well the claims organization in a taxonomy can lead to more complete student designs that cover important design concerns. Second, gauge the ability of reference claims to increase the amount of reuse and therefore the amount of HCI knowledge transferred to students. And third, measure the ability of reference claims to describe the intended designs.

7.1 Reference Claims in the Claims Library

The notification systems taxonomy presented in chapter 5 was developed as a new organizational method in the claims library of the LINK-UP tool described in chapter 2. All the reference claims presented in chapter 5 were placed within a category in the taxonomy following

* Sections of this chapter are extract from Fabian et al. 2006

the structure of Figures 5.1 and 5.2. With this addition, students were able to browse the library by navigating within the taxonomy and viewing claims associated with each category. In total, 50 new reference claims were added to the claims library. Each category of the taxonomy had at least one reference claim. None of the traditional claims in the library were placed within the categories. Figure 7.1 shows a claim in the claims library.

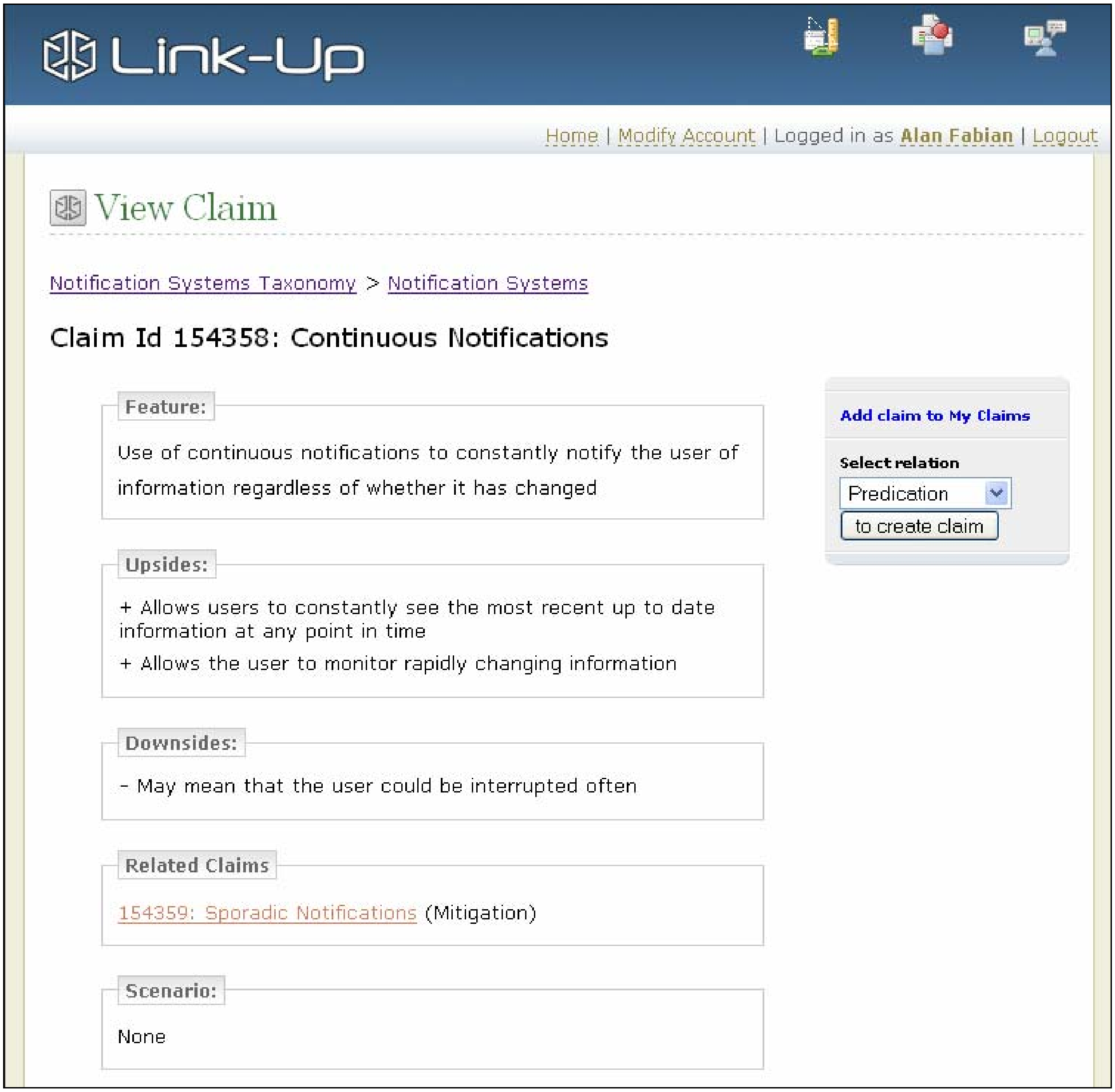


Figure 7.1: A reference claim in the new claims library.

7.2 Experimental Design

The experimental design revolved around the investigation of three main points of interest. First, measure how well the claims organization in a taxonomy could lead to a more complete design that covered important design concerns. Second, gauge the ability of reference claims to increase the amount of reuse and therefore the amount of HCI knowledge transferred to students. And third, evaluate the descriptive capabilities of reference claims. The claims library was prepared for a study that would allow students to collect claims and create a conceptual design for a notification system.

Twenty-four HCI students participated in the study. All the students were familiar with the use of claims, but their familiarity with the claims library varied. A few had used the library before. Most of the others had either heard of it or had no knowledge of it. None of the students were familiar with the concept of reference claims. Almost all the participants were familiar with the notification systems domain although this varied to different degrees. Each student was paired to create a total of 12 groups for the study. Instructions presented each group with a scenario depicting the need for a notification system in a train station and asked them to work together to design a notification system to solve the problem.

To create the system, they were instructed to access the claims library and search for claims they believed would apply to their design. For every claim they wished to reuse, they were told to write down the unique ID number of the claim on a form we provided. If they could not find a claim they needed, the option to create claims was given. The students were free to reuse and create as many claims as they wished. We did not place a time limit and asked the participants to end their design process when they felt their system was sufficiently described by their claims.

We altered two different variables in two different ways to create four conditions for our study. The first variable was the type of claim designers would be given access to. In the Search and Browse (SB) condition groups could view traditional claims by using a search engine or browse through the library linearly to view all the claims. In the Reference Claims and Taxonomy (RT) condition, reference claims could be found by browsing through the taxonomy and viewing the claims associated with each category. At no point were users able to view both types of claims. This alteration in the study allowed the investigation of the reusability and descriptive ability of reference claims. Figure 7.2 presents the SB condition and figures 7.3 and 7.5 present the RT condition to find claims.

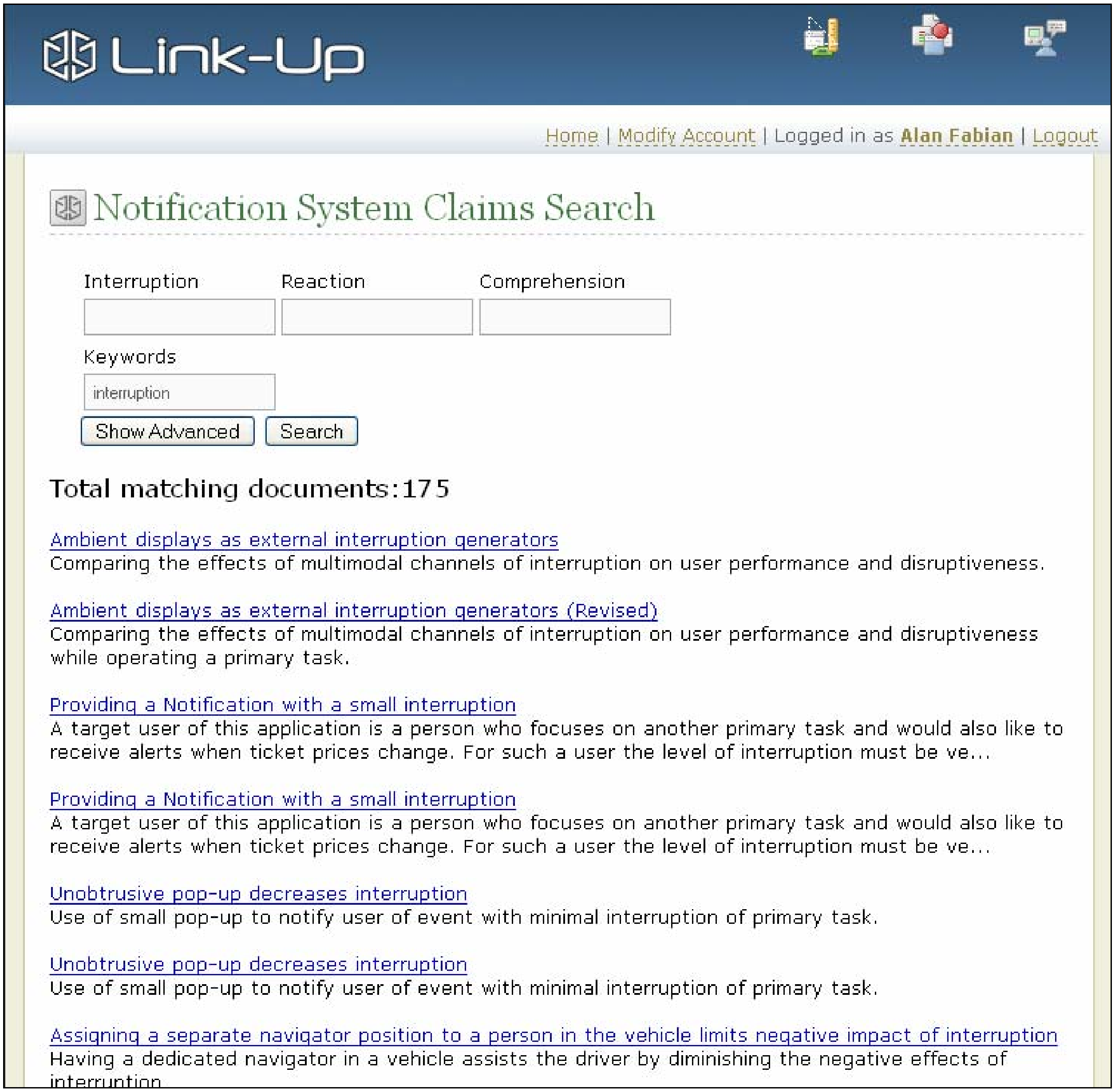


Figure 7.2: Claim search and browse in claims library.

Figure 7.3: Notification systems taxonomy and reference claims in the claims library.

Figure 7.4: Lower-level of the notification system taxonomy in claims library.

The second variable changed the instructions that were given to the participants regarding the design of the notification system. The Guided (G) condition provided half the groups with extra guiding instructions pointing out design aspects to consider while designing notification systems. They were instructed to consider the three critical parameters, the type of notification system they wished to design, and the methods used to notify users of the system. The Unguided (UG) condition did not received these instructions in any way. This variation in the study allowed considering the impact of the taxonomy on the designed systems. All the groups were placed in one of the conditions with a total of 3 in each condition.

	Search and Browse (SB)	**Reference Claims and Taxonomy (RT)**
Unguided (UG)	• Access to traditional claims • Uses search engine or linear browse • Not given notification system design aspects	• Accesses reference claims • Uses the taxonomy • Not given notification system design aspects
Guided (G)	• Access to traditional claims • Uses search engine or linear browse • Given notification system design aspects	• Accesses reference claims • Uses the taxonomy • Given notification system design aspects

Table 7.1: The four conditions in the reference claim experimental design.

At the end of the experiments all students were asked to individually fill out a post-questionnaire. The questionnaire consisted of open ended questions asking to describe their design and questions based on a seven-point Likert scale.

There were three hypotheses associated with this study. First, students were expected to create “better” notification systems. “Better” refers to students creating designs that take into account the three design aspects given to some groups in their instructions. The reference claim condition with extra guiding instructions was seen as the condition which should lead to the highest amount of reuse. The extra instructions should force the groups to look for more claims to make sure they covered all the design aspects. Also, although approximating how many more claims would be reused is difficult, the reference claim groups were expected to reuse more claims than the other groups. Finally, the sets of reference claims should better describe the intended designs than the sets of traditional claims.

7.3 Results

The first point of interest was to gauge the environment's ability to lead to "better" designs. The assumption is that for a design to achieve a greater degree of success, it should incorporate the concerns addressed by the three identified design guidelines. Design scores were assigned to each group based on how well they accounted for the three guidelines: critical parameters (interruption, reaction, and comprehension), notification system types and notification methods. The list of claims was analyzed and decisions were made based on whether the claims specifically addressed any of the design concerns. One point was given for each design concern successfully addressed by the set of claims. Thus, the maximum score possible was 5, with 3 points for critical parameters and 1 each for notification system type and notification method. Three experienced notification systems researchers worked independently to rate the designs from each group to minimize bias, yielding three different scores for each design. The scores were then averaged and used in the analysis (see table 7.2).

	SB	**RT**	**SB and RT**
UG	Mean = 1.77 SD = 0.50	Mean = 3.88 SD = 0.76	Mean = 2.83 SD = 1.29
G	Mean = 2.33 SD = 0.66	Mean = 3.44 SD = 0.50	Mean = 2.88 SD = 0.80
G and UG	Mean = 2.05 SD = 0.61	Mean = 3.66 SD = 0.63	

Table 7.2: Mean and standard deviation for the design scores

The analysis of the design scores proved to be very encouraging. There was a significant difference in design scores between the SB and RT conditions (t(10)=-4.48, p=0.001). This overall result was bolstered by differences between the individual conditions (see Table 7.2, color coded results are significant). The UG SB and the UG RT conditions had a significant difference (t(4)=-3.96, p=0.016) and the G SB and G RT condition was close to having a significant different (t(4)=-2.29, p= 0.08). The unguided RT groups actually performed better than the guided RT groups hindering the need for guidelines in the RT condition.

There was an uneven distribution in the scores between the design categories. The average score for the critical parameters aspect was the smallest (0.43) showing this was the least considered aspect. Notification system type had an average of 0.58. Most of the claims in groups were about notification methods, yielding notification method mean score of 0.97.

For this part of the analysis focus was also turned to questions asked in the post-questionnaire to further gauge the impact of the taxonomy. Participants were asked to rate their familiarity with the notification systems domain before and after the experiment. A seven-point Likert scale was used for this question.

The students subjected to the RT condition were found to show a significant improvement in their perceived familiarity with the domain (t(22)=-2.56, p=0.017). We could not find any significant improvement in the SB condition students (t(22)=-1.32, p=0.2). The results of the design scores and the improvements seen in the designers' domain familiarity indicate that the use of the taxonomy can lead to more complete designs that cover more important design concerns.

The second interest was to analyze the amount of reuse that occurred in each condition. We can refer to the number of claims indicated as reused by each group (see figure 7.5). When comparing each condition individually, there were no significant differences. This is mainly due to the fact that each condition only had 3 data points. Averages suggest the most reuse occurred in the G and RT condition.

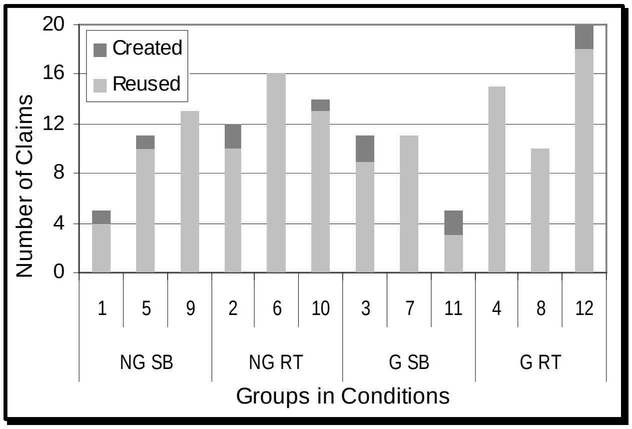


Figure 7.5: Number of reused and created claims for each group in each condition

Comparing all the groups in the SB and RT conditions yielded appealing results (see table 7.3, color coded results are significant). A T-test suggests the number of claims reused in the RT condition was significantly greater than in the SB condition ($t(10)=-2.53$, $p=0.029$). There was no significant difference in reuse between the G and UG conditions, but this was expected. There was no significant difference in the number of claims created.

	SB	**RT**	**SB and RT**
UG	Mean = 9 SD = 4.5	Mean = 13 SD = 3	Mean= 11 SD = 4.09
G	Mean = 7.66 SD = 4.16	Mean = 14.33 SD = 4.04	Mean = 11 SD = 5.17
G and UG	Mean = 8.33 SD = 3.98	Mean = 13.66 SD = 3.26	

Table 7.3: The means and standard deviations for the number of reused claims

These results indicate that reference claims, combined together with the taxonomy, do have the ability to increase the amount of reuse during design.

For the final part of the analysis we can again turn to the post-questionnaire. To understand how well reference claims could describe the intended design, users were asked to rate how well they believe their claims represent their design using a seven-point Likert scale.

The answers to this question turned out to be diverse. The analysis suggested the reference claims were less representative with a mean of 4.16. The traditional claims had a mean rating of 5.13. The difference, however, was not found to be significant (p=0.093).

As a follow up to the above question users were also asked to answer in an open ended manner as to why they created the claims they did. Of all groups that used the reference claims, one group specifically said that the claims were too general to apply to their system. Two other groups said that they wanted to add more features to their system that the reference claims did not cover. Three groups chose not to answer this question as they did not create any claims. For the traditional claims condition, most groups said that they created claims because they couldn’t find any claim that described the features they envisioned. Two groups in this condition did not answer this question as they did not create any claims. These observations lead to reject the last hypothesis. Reference claims are not necessarily better at describing the intended designs.

7.4 Discussion

The experiment results provide insight into the three hypotheses: the increase in reuse, positive impact of the taxonomy, and better descriptive ability of reference claims. Many of the conclusions from the experiment are encouraging in several ways while others provide insight into possible reasons for why unexpected results were observed.

7.4.1 Design Impact of the Taxonomy

The first hypothesis led to investigate whether the taxonomy could encourage students to consider all the important design concerns. The results of our design score analysis show that the RT condition bested the SB condition when participants were guided and not guided by extra design guidelines.

These results can be attributed to the structure of the taxonomy. The top-level structure of the taxonomy defined the overall domain in terms of the three design aspects that everyone was expected to consider. If participants wanted to consider a critical parameter such as interruption, they could easily do so by entering that part of the taxonomy. This process was not possible in the SB condition. Most participants in this condition used claims about notification methods since they were the easiest to find. Taking such critical aspects into account can potentially ensure a more complete design.

This inherent inhibiting factor is further exemplified when we consider all four experimental conditions. When the participants are told what design aspects to consider, we can see a considerable difference, although not statistically significant, between the SB and RT conditions. When they are not told, we see a solid significant difference. This leads us to conclude that even when the participants were told what to consider, they were not able to do so. The lack of organization and poor writing of the traditional claims contributed to this factor. When participants in the RT condition were told what characteristics to incorporate in their design, they were able to do so relatively successfully. When they were not told what to consider, they still implicitly considered many of these aspects.

The results also point out that even the users of the taxonomy did not always consider every single design aspect. A possible explanation to this is the fact that not every category in the taxonomy had a large selection of reference claims. The participants wanted to consider a particular aspect and looked for claims in the taxonomy, but did not find a claim that was to their liking, forcing them to ignore that aspect of their design. Although participants had the option to create claims, none of the groups in the taxonomy condition created a claim to specifically consider a design aspect they could not address by reusing a reference claim from the library. We can explain this occurrence to be a result of participants already having many reference claims for their design. They did not feel the need to create more claims. Despite these observations, the taxonomy is a step in the right direction. Its use encourages students to create more complete designs. Through the expansion and refinement of the taxonomy and the placement of more reference claims, we can anticipate students will be more likely to consider all design aspects. Overall, these results demonstrate that organizing claims in a taxonomy can lead students to create more complete designs. By using this taxonomy as a reference for what to consider, students can therefore quickly understand and define they important design concerns they need to address for their systems.

Certainly the current state of the taxonomy also proved to be beneficial to students in another way. Results show the taxonomy increased the participants' familiarity with the notification systems domain over the course of the experiment. This is another factor contributing to the increase in consideration of the design aspects—a potentially helpful tool for student designers. The results demonstrate the taxonomy's ability to focus students' design efforts around key design concerns.

7.4.2 Reusability of Reference Claims

The second hypothesis was that using reference claims and a notification system taxonomy would increase reuse. A significant difference in the amount of reuse between all the groups in the SB and RT conditions demonstrates that the use of reference claims can increase reuse. We could have anticipated these results because of the generalized reference claim language and improved browsing capabilities based on organization.

The most important reason for the increase in reuse is the way the claims are written. Participants found that most of the reference claims could be used in a different context. This can be observed in the different types of notification systems that were designed. The same claims were applied in completely different ways for different design features. This is an indication of them being able to easily apply the high-level concepts to their designs. The claims were not tied to any artifacts, making them easier to reuse. Most traditional claims were very specific in many cases. Participants found traditional claims that only partly defined the description of the design feature they wanted to implement. This was evident when comparing their claims to the textual description of their notification system. Many students reused traditional claims with the intention of only using part of the concepts within the claims. Some groups decided to create more general claims because they could not find any appropriate claims.

The classification of claims by categories in the RT condition may have improved browsing capabilities by allowing them to discover claims rather than requiring them to formulate queries. The query formulation process usually requires the user to anticipate the contents of the library and to know the important design concerns that need to be covered. Reuse may have been inhibited by the lack of familiarity with the library in the SB condition. By grouping claims into clusters in the RT condition, reference claims were made more accessible, increasing the probability of being found and reused.

In some cases the RT condition allowed students to realize they needed more claims. The taxonomy made them notice there were aspects of the notification systems domain that they may not have been considering. Therefore, they were more prone to look for more claims covering specific aspects of a design, subsequently increasing the number of reused claims. Although not significant, this was observed in the responses among several participants to a question asking for the driving factor (availability of claims or predetermined design vision) behind looking for claims. This was not always possible in the SB condition. Without a taxonomy, participants only looked for claims if they had a predetermined design aspect they wanted to consider.

These results therefore strongly support that the use reference claims coupled with a taxonomy to organize them, increases reuse. This is further reinforced by the fact that there were over 200 claims in the SB condition compared to only 50 in the RT condition to cover all the important design concerns. Reference claims therefore position themselves as an effective means to teach students HCI knowledge encapsulated in design alternatives for their systems, and to limit the amount of time they spend trying to recreate design features that already exist, and have identified tradeoffs associated with them.

7.4.3 Descriptive Capabilities of Reference Claims

The third hypothesis proposed that reference claims would be better at depicting the overall system due to their broader nature. The results showed that the reference claims did not necessarily fare well in describing the intended system although no significant differences were found. Reference claims seemed too abstract to capture the design accurately. The opposite was said about the traditional claims—participants found them to be too narrow in scope, capturing additional undesired characteristics.

Although the observation force the rejection of the third hypothesis, the results should not be surprising. Previous literature has mentioned the need for both general and specific reusable components. Kruger (1992) defines four aspects that are inherent to reuse: abstraction, selection, specification, and integration. The notion of specification addresses the need for tailoring a general component such that it better addresses the needs of an intended design. This notion is also valid in the case of claims. Sutcliffe (2002) states, "Their advantage may lie in the combination of the specific with the general".

The combination of reference claims and traditional claims can be a perfect solution to addressing this problem. A collection of claims created to specify a system should account for gradually changing levels of scope. This motivates the use of both reference claims and traditional claims during the design process. Reference claims can describe high-level goals of a system, but without traditional claims, students cannot narrow down how to implement a system. One student did mention that there was a need to add more specific features not mentioned by reference claims, which supports this assessment. This vision is consistent with the suggestion that the knowledge contained within reference claims should cause the creation of new design features in the form of traditional claims, leading to incremental improvements in the overall design.

Chapter 8

Conclusion and Future Work

This thesis focused on creating an interactive learning environment based on reference claims to aid students in gathering important HCI knowledge necessary in a problem-based constructivist learning approach which can improve the learning of HCI design compared to more traditional, passive, learning techniques. An initial prototype for the learning environment introduced a claims library to encapsulate design features and from which students can gather HCI knowledge. Pilot testing outlined the accessibility, applicability, and reusability problems associated with this initial claims library. Reference claims organized in a taxonomy were introduced as an improved form of claims to solve these issues. The taxonomic organization was designed as a way to expose students to important design concerns as well as a classification schema for claims to increase their accessibility. Inspired by reference tasks, reference claims were introduced as an improved, more applicable, and more reusable encapsulation form of HCI knowledge to expose students to design alternatives for design concerns. A detailed learning environment with a taxonomy and a set of reference claims for the domain of notification systems demonstrated how existing theories of design can be encapsulated into reference claims. The potential applicability and reusability of reference claims for various designs within a particular domain was demonstrated in an experiment. Finally, an evaluation assessed the benefits of this learning environment based on reference claims in terms of improving student designs and increasing the amount of HCI knowledge they reuse.

8.1 Major Findings and Contributions

This thesis aimed at supporting a constructivist approach of HCI learning by creating an interactive learning environment to improve student gathering of important HCI knowledge. The major findings are the following:

- **Reference claim knowledge is more applicable and reusable than traditional claim knowledge.** Results in chapter 6 demonstrate that when properly implemented, the four reference claim characteristics make them applicable and reusable for a variety of contexts within a domain. Reference claim knowledge can also be refined through testing to increase their applicability, reusability and knowledge transfer capabilities.

- **An interactive learning environment based on reference claims organized in a taxonomy improves students' designs by enabling them to effectively browse through important design concerns and the corresponding design alternatives.** Results from Chapter 7 demonstrate that organizing reference claims in a taxonomy can lead students to create more complete designs. By using this taxonomy as a reference for what to consider, and reference claims as a set of alternatives, students can therefore quickly understand, define, and address the important design concerns for their systems.

- **This learning environment also leads to a significant increase in reuse of HCI knowledge during design.** Results from Chapter 7 strongly support that the use reference claims coupled with a taxonomy to organize them, increases reuse. Reference claims therefore position themselves as an effective means to teach students HCI knowledge encapsulated in design alternatives for their systems, and to limit the amount of time they spend trying to recreate design features that already exist, and have identified tradeoffs associated with them.

- **However, the environment may also need more specific claims with design knowledge that more accurately describes system features.** Results showed that reference claims did not necessarily fare well in describing the students' intended systems. Reference claims seem too abstract to capture implementation details for students.

The major contributions are the following

- **A library of claims that acts as applicable references that encapsulate design features from which students can draw design knowledge.** Chapter 3 introduced an initial claims library prototype that encapsulated design features that students could use to browse for design knowledge.
- **Reference claims as an improved form of claims that increase the accessibility, applicability, and reusability of claims.** Chapter 4 presented a set of characteristics for reference claims based on applicability, ability to lead to new features, generality, and artifact independency, aimed at improving the encapsulation schema of design knowledge into claims.
- **An interactive learning environment based on a taxonomic organization of reference claims to create an effective knowledge repository from which students can gather important HCI design knowledge necessary in a constructivist learning approach to HCI.** Chapter 5 described the creation of a taxonomy and reference claims for the domain of notification systems that combine to create an interactive learning environment for the design of theses systems.
- **Proof of the potential applicability, reusability, and accessibility of this learning environment which suggests its potential to support an information gathering stage of a constructivist approach to HCI.** The results of the experiments in Chapters 6 and 7 support the potential of reference claims in terms of applicability, reusability and accessibility which are critical characteristics to share design knowledge among students.

These conclusions confirm that a constructivist leaning environment based on reference claims can benefit students learning HCI. The taxonomy exposes students to important design concerns and improves their understanding of the design domain. Reference claims convey the knowledge gathered from previous designs to students and their organization in the taxonomy teaches them a range of design solutions along with the tradeoffs associated with their use. However, although reference claims do provide the advantage of being more reusable than traditional claims, they also lack the specificity that would enable them to more accurately define the intended systems.

8.2 Future Work

Traditional claims are still important to students designing interactive systems because they can allow them to narrow down how to accurately implement a system within a specific context. This learning environment can be improved by giving students the ability to access both reference claims in a taxonomy and more specific traditional claims. In order to not hinder the benefits of the taxonomy and reference claims, traditional claims should be linked to reference claims through the use of relationships. This addition could allow students to navigate from reference claims to traditional claims using claim relationships (Wahid et al., 2004). Students will be able to more accurately define their designs using linked traditional claims without hindering the positive aspects of reference claims.

We have also seen that many categories within the taxonomy did not provide enough reference claims to address design concerns. More reference claims must be created to cover lower levels of the taxonomy. Chapter 5 illustrated how new reference claim to cover sub-domains of a taxonomy can be created. Increasing the number of reference claims in the library should increase the amount of HCI knowledge transferred to students. Furthermore, Chapter 6 illustrated that the knowledge captured in reference claims can be furthered as new experiments yield new results. Most reference claims were written by experienced designers anticipating the tradeoffs associated with their use. New experiments should be conducted to verify and adjust the knowledge that is encapsulated in reference claims in order to ensure the accuracy of the knowledge they transfer to students.

Finally, this learning environment only supports the information gathering stage of a problem-based constructivist approach to HCI. Other tools that support problem analysis, synthesis, abstraction, and reflection could be built to create a development environment that would support all stages of a constructivist approach to the learning of HCI.

Instructors could use this interactive learning environment in introductory HCI classes to complement traditional HCI lectures and textbooks. Its interactivity and organization can improve over traditional HCI material used in class by making it easier for students to access a wide range of HCI knowledge and helping them understand its impact by applying it to a concrete design. This constructivist approach to the teaching of HCI design has the potential to enable students to better comprehend how to use core HCI knowledge to address important design concerns when they design interactive systems.

References

Arroyo, E., Selker, T., Stouffs, A. (2002). Interruptions as Multimodal Outputs: Which are the Less Disruptive? In *Proceedings of the fourth IEEE International Conference on Multimodal Interfaces*, 479-483.

Borchers, J. O. (2000). A pattern approach to interaction design. In *Proceedings of Designing Interactive Systems*, 369-378.

Carroll, J. M. (2000). *Making Use: Scenario-Based Design of Human-Computer Interactions.* Cambridge, MA: MIT Press.

Carroll, J. M. and Rosson, M. B. (1992). Getting Around the Task-Artifact Cycle: How to Make Claims and Design by Scenario. *ACM Transactions on Information System*, 10, 2, 181-212.

Chewar, C.M, Bachetti, E., McCrickard, D.S., Booker, J. (2004). Automating a Design Reuse Facility with Critical Parameters: Lessons Learned in Developing the LINK-UP System. In *Proceedings of the 2004 International Conference on Computer-Aided Design of User Interfaces*, 236-247.

Chewar, C.M and McCrickard, D.S. (2005). Links for a Human-Centered Science of Design: Integrated Design Knowledge Environments for a Software Development Process. In *Proceedings of the 2005 Hawaii's International Conference on System Sciences*, 10 pgs (CD-ROM).

Creech, M. L., Freeze, D. F., and Griss, M. L. (1991). Using Hypertext in Selecting Reusable Software Components. In *Proceedings of ACM Hypertext and Hypermedia*, 25-38.

Cutrell, E., Czerwinski, M., Horvitz, E. (2001). Notification, Disruption, and Memory: Effects of Messaging Interruptions on Memory and Performance. In *Proceedings of Interact 2001*, 263-269.

Dusink, L., and van Katwijk, J. (1995). Reuse Dimensions. In *Proceedings of Symposium on Software Reusability*, 137-149.

Ellis, A., Carswell, L., Bernat, A., Deveaux, D., Frison, P., Meisalo, V., Meyer, J., Nulden, U., Rugelj, J., and Tarhio, J. (1998). Resources, tools, and techniques for problem based learning in computing. *Working Group reports of the 3rd annual SIGCSE/SIGCUE ITiCSE conference on Integrating technology into computer science education*, 41–56.

Fabian, A., Felton, D., Grant, M., Montabert, C., Pious, K., Rashidi, N., Tarpley III, A. R., Taylor, N., Chewar, C. M. and McCrickard, D. S. (2004). Designing the Claims Reuse Library: Validating Classification Methods for Notification Systems. In *Proceedings of ACM Southeast Conference*, 357-362.

Fabian, A., Wahid, S., Bhatia, S., McCrickard, D.S. (2006). Creating an Interactive Learning Environment with Reusable HCI Knowledge. *World Conference on Educational Multimedia/Hypermedia and Educational Telecommunications*, to appear.

Franke J. L., Daniels J. J. & McFarlane D. C. (2002) Recovering context after interruption, in: W. Gray & C. Schunn (Eds.), In *Proceedings of 24th Annual Meeting of the Cognitive Science Society*, 310-315.

Gamma, E., Helm, R., Johnson, R., and Vlissides, J. (1995). *Design Patterns: elements of reusable object-oriented software*. Boston: Addison-Wesley Longman.

Greenberg, S. and Rounding, M. (2001). The Notification Collage: Posting Information to Public and Personal Displays. In *Proceedings of Conference on Human factors in Computing Systems*, 524-521.

Horvitz, E., Apacible, J. (2003). Learning and reasoning about interruption. In *Proceedings of the Fifth International Conference on Multimodal Interfaces*, 20-27.

Krueger, C. W. (1992). Software Reuse. *ACM Computing Surveys*, 24, 2, 131-183.

Landay, J. A., and Borriello, G. (2003). Design Patterns for Ubiquitous Computing. *IEEE Computer*, 36, 8, 93-95.

Lee, J.C, Chewar, C.M, and McCrickard, D.S. (2004). Image is Everything: Advancing HCI Knowledge and Interface Design Using the System Image. In *Proceedings of the ACM Southeast Conference*, 2-376 - 2-381.

Lee, J.C, Lin, S, Chewar, C.M, McCrickard, D.S, Fabian, A., Jackson, A.(2004). From Chaos to Cooperation: Teaching Analytic Evaluation with LINK-UP. *Proceedings of the World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education*, 2755-2762.

Lee, J. C., Wahid, S., Chewar, C. M., Congleton, B., and McCrickard, D. S. (2005). Spiraling Toward Usability: An Integrated Design Environment and Management System. *Technical Report TR-05-15* Computer Science, Virginia Tech.

Long, L. and Lai J. (2001). Shall we mix synthetic speech and human speech?: impact on users' performance, perception and attitude. In *Proceedings of Conference on Human factors in Computing Systems*, 158-165.

McCrickard, D.S. and Chewar, C.M. (2003). Attuning Notification Design to User Goals and Attention Costs. *Communications of the ACM* , 46 (3), 67-72.

McCrickard, D. S., Chewar, C. M., Somervell, J. P., and Ndiwalana, A. (2003). A Model for Notification Systems Evaluation—Assessing User Goals for Multitasking Activity. *ACM Transactions on Computer-Human Interaction*, 10, 4, 312-338.

McFarlane, D.C. (2002). Comparison of Four Primary Methods for Coordinating the Interruption of People in Human-Computer Interaction. *Human-Computer Interaction*, 17, 1, 63-139.

McFarlane, D. C. and Latorella, K. A. (2002). The scope and importance of human interruption in human-computer interaction design. *Human-Computer Interaction*, 17, 1, 1-61.

Mordechai B. (1998). Constructivism in computer science education. In *Proceedings of the twenty-ninth SIGCSE technical symposium on Computer science education*, 257–261.

Nass, C. and Gong, L. (2000). Speech Interfaces from an Evolutionary Perspective. *Communications of the ACM*, 43, 9, 36-43.

Newman, W. M. (1997). Better or just different? On the benefits of designing interactive systems in terms of critical parameters. In *Proceedings of Designing Interactive Systems*, 239-245.

Norman, D.A. (1986). Cognitive engineering. In *D. A. Norman & S. W. Draper (Eds.) User Centered Systems Design: New Perspectives on Human-Computer Interaction*, 31-61.

Parker, J. R. and Becker, K. (2003). Measuring effectiveness of constructivist and behaviorist assignments in cs102. In *Proceedings of the 8th annual conference on Innovation and technology in computer science education*, 40–44.

Payne, C., Allgood, C. F., Chewar, C. M., Holbrook, C., and McCrickard, D. S. (2003). Generalizing Interface Design Knowledge: Lessons Learned from Developing a Claims Library. In *Proceedings of Information Reuse and Integration*, 362-369.

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

Sabri, A., Ball, R., Fabian, A., Bhatia, S., and North, C. (2006). High-Resolution Gaming: "Interfaces, Notifications and the User Experience". Submitted to Special Issue of *Interacting with Computers* on "HCI Issues in Computer Games".

Savery, J.R. and Duffy, T.M. (1995). Problem based learning: An instructional model and its constructivist framework. In *Educational Technology,* 35:31–38.

Sawhney, N., and Schmandt, C. (1999). Nomadic Radio: Scaleable and Contextual Notification for Wearable Audio Messaging. In *Proceedings of Conference on Human factors in Computing Systems*, 96-103.

Sutcliffe, A. G. (2000). On the Effective Use and Reuse of HCI Knowledge. *ACM Transactions on Computer-Human Interaction*, 7, 2, 197-221

Sutcliffe, A. (2002). The Domain Theory: Patterns for Knowledge and Software Reuse. Mahwah, NJ: Lawrence Erlbaum Associates.

References

Sutcliffe, A. G. and Carroll, J. M. (1999). Designing Claims for Reuse in Interactive Systems Design. *International Journal of Human-Computer Studies*, 50, 3, 213-241.

Vat, K.H. (2000). Teaching software psychology: expanding the perspective. In *Proceedings of the thirty-first SIGCSE technical symposium on Computer science education*, 392–396.

Vat, K.H. (2001). Teaching hci with scenario-based design: the constructivist's synthesis. In *Proceedings of the 6th annual conference on Innovation and technology in computer science education*, 9–12.

Wahid, S. Allgood, C. F., Chewar, C. M., and McCrickard, D. S. (2004). Entering the Heart of Design: Relationships for Tracing Claims Evolution. In *Proceedings of Software Engineering and Knowledge Engineering*, 167-172.

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

Appendix A: Continuous Notifications on Large High Resolution Displays

IRB # 06-204

A.1 Experiment Briefing

Briefing

For this experiment, you will be competing in five (5) separate games using against a computer opponent on a nine (9) screen projection display. You will be allowed a break between each game if you desire. Each game you play will have the same map and scenario, but the user interface will change. Each game should approximately 10-20 minutes to complete. There have been some modifications to the game (see below) so you will be given 10 minutes to practice to become accustomed to them and the new display. Please ask questions during this practice session to ensure you understand the mechanics of the game. At the end of the experiment you will be asked to fill out a simple and confidential questionnaire.

Secondary Task:

> While your primary goal is to kill as many computer units as fast as possible, you will also have a secondary task. **WHENEVER YOU SEE A RED SQUARE, PRESS THE SPACE BAR.**

Limitations:

- You can only build Peasants, Footmen, Archers, Demolition Squads and their respective buildings.
- There are **no** unit upgrades.
- Sound has been disabled.
- The resources are fairly low.

Winning/Losing:

If you are out of resources **AND** have no more units I will declare you as dead and you will have to surrender. You will win if the computer opponent is in this situation.

User Interface:

Each user interface has been changed slightly from the original presented in the game. Before each game, the UI will be demonstrated to you until you understand it. Some of the user interfaces require that you press the **MIDDLE MOUSE BUTTON / SCROLL WHEEL** to activate a special part.

SPECIAL

If Red Square	**Space Bar**
Change User Interface	**Click Mouse Wheel**

BUILDINGS

Town Hall	**B H**
Keep (Upgrade)	**K**
Farm	**B F**
Barracks	**B B**
Blacksmith	**B S**
Lumberyard	**B L**
Gnomish Inventor	**V I**

UNITS

Peasant	**P**	**400g**
Footman	**F**	**600g**
Archer	**A**	**500g 50w**
Demo Squad	**D**	**700g 250w**

COMMANDS

Attack	**A**
Patrol	**P**
Stand Ground	**T**
Stop	**S**
***Demolish!**	**D**
Move	**M**

For *Non*-Peasants, Right-Click will Both Attack and Move

A.2 Experiment Questionnaire

Warcraft Experiment

Please enter your name and PID:

How familiar are you with video games?

- 1 (Not Familiar)
- 2
- 3
- 4
- 5
- 6
- 7 (Very Familiar)

How familiar are you with Warcraft (any version) ?

- 1 (Not Familiar)
- 2
- 3
- 4
- 5
- 6
- 7 (Very Familiar)

How familiar are you with using large displays (more than one monitor)?

- 1 (Not Familiar)
- 2
- 3
- 4
- 5
- 6
- 7 (Very Familiar)

The following questions refer to the Standard design (resources on the side only)

What did you like about this design ?

What did you not like about this design?

How interruptive did you find this design?

- 1 (Not Interruptive)
- 2
- 3
- 4
- 5
- 6
- 7 (Very Interruptive)

How would you rate this design overall?

- 1 (Bad)
- 2
- 3
- 4
- 5
- 6
- 7 (Good)

The following questions refer to the Immediate design (resources always on cursor)

What did you like about this design?

What did you not like about this design?

How interruptive did you find this design?

- 1 (Not Interruptive)
- 2
- 3
- 4
- 5
- 6
- 7 (Very Interruptive)

How would you rate this design overall?

- 1 (Bad)
- 2
- 3
- 4
- 5
- 6
- 7 (Good)

Do you think this design improved your performance compared to the standard design?

- Yes
- No
- Not Sure

The following questions refer to the Negotiated design (resources on demand)

What did you like about this design?

What did you not like about this design?

How interruptive did you find this design?

- 1 (Not Interruptive)
- 2
- 3
- 4
- 5
- 6
- 7 (Very Interruptive)

How would you rate this design overall?

- 1 (Bad)
- 2
- 3
- 4
- 5
- 6
- 7 (Good)

Do you think this design improved your performance compared to the standard design?

- Yes
- No
- Not Sure

The following questions refer to the Mediated design (resources on cursor on computer decision)

What did you like about this design?

What did you not like about this design?

How interruptive did you find this design?

- 1 (Not Interruptive)
- 2
- 3
- 4
- 5
- 6
- 7 (Very Interruptive)

How would you rate this design overall?

- 1 (Bad)
- 2
- 3
- 4
- 5
- 6
- 7 (Good)

Do you think this design improved your performance compared to the standard design?

- Yes
- No
- Not Sure

The following questions refer to the Scheduled design (resources on cursor every x seconds)

What did you like about this design?

What did you not like about this design?

How interruptive did you find this design?

- 1 (Not Interruptive)
- 2
- 3
- 4
- 5
- 6
- 7 (Very Interruptive)

How would you rate this design overall?

- 1 (Bad)
- 2
- 3
- 4
- 5
- 6
- 7 (Good)

Do you think this design improved your performance compared to the standard design?

- Yes
- No
- Not Sure

Comparison of all 5 designs

Which design did you feel was the most interruptive?

- Immediate (resources always visible)
- Negotiated (resources on demand)
- Scheduled (resources show every x seconds)
- Mediated (resources show on computer decision)

Which design did you feel was the least interruptive?

- Immediate (resources always visible)
- Negotiated (resources on demand)
- Scheduled (resources show every x seconds)
- Mediated (resources show on computer decision)

Which design did you feel was the best for detecting the red square?

- Standard (the normal warcraft)
- Immediate (resources always visible)
- Negotiated (resources on demand)
- Scheduled (resources show every x seconds)
- Mediated (resources show on computer decision)

Which design did you feel was the worst for detecting the red square?

- Standard (the normal warcraft)
- Immediate (resources always visible)
- Negotiated (resources on demand)
- Scheduled (resources show every x seconds)
- Mediated (resources show on computer decision)

Which design did you prefer?

- Standard (the normal warcraft)
- Immediate (resources always visible)
- Negotiated (resources on demand)
- Scheduled (resources show every x seconds)
- Mediated (resources show on computer decision)

Why did you prefer that design?

Which design did you believe improved the most over the standard design?

- Immediate (resources always visible)
- Negotiated (resources on demand)
- Scheduled (resources show every x seconds)
- Mediated (resources show on computer decision)

Why did you believe that design improved the most over the standard design?

Submit

Appendix B: Reference Claim Study

B.1 Claims Library Activity

IRB # 05-142

Group ID:
Group Member ID:

Instructions:
The LINK-UP system is a design environment that facilitates the use and reuse of claims for notification systems design. This activity will give you the opportunity to create a notification system by reusing claims in the LINK-UP Claims Library. A problem scenario describing the need for a notification system has been provided. Based on this scenario, collect a set of claims you wish to reuse to define the prototype of your envisioned system.

Problem Scenario:
Many occurrences throughout the day at a train station cause the train schedule to constantly change. Commuters need to know the most recent information regarding arrivals, departures, and delays. Currently, the only way to get the most up to date arrival and departure times is to ask a station employee at an information desk. The station manager receives complaints from many commuters regarding how the time spent in line for the information desk could be better spent doing other activities such as reading, eating, and working on their laptops. The station manager wishes there was a way to inform the commuters of train schedules and updates while allowing them to continue performing their other tasks. Design a notification system to meet the station manager's needs.

Using the LINK-UP system
Go to http://ticker.cs.vt.ed:8080/Linkup
Login –
Password –
After successfully logging in you will see the LINK-UP main user portal.
Click on "Browse claims category based on hierarchy". (YOU MAY NOT USE ANY OTHER LINK)
This will bring you to the Notification Systems Taxonomy page.
You can browse through the taxonomy by clicking on the subcategories present at each level.
Claims associated with each subcategory (if any) are displayed below the category listings.
Clicking on the claim title will allow you to view the claim in detail. You can view a claim and look at its features, upsides, and downsides. When viewing claims use the back button to go back to the taxonomy.
If you find a claim that represents your system and want to use it in your design, write down the claim ID number and title in the Claim list form provided.

If you can not find an appropriate claim that represents a feature of your system, you may create the claim on the claim creation form.

Using the LINK-UP system (second version)
Go to http://ticker.cs.vt.ed:8080/Linkup
Login –
Password –
After successfully logging in you will see the LINK-UP main user portal.
Use the following two links to search for claims (YOU MAY NOT USE ANY OTHER LINK)
"Browse claims library"
"Search for claims using Lucene"
"Browse claims library"
Using this option you can sequentially browse through all the claims stored in the library.
You can browse through claims using the previous/next buttons or entering the claim id number in the textbox.
"Search for claims using Lucene"
Using this option you perform a keyword search on all the claims in the library.
You can type in the keyword in the keyword textbox and hit search. This will display a list of claims that matched the keyword query.
Clicking on the claim title will allow you to view the claim in detail. You can view a claim and look at its features and upsides and downsides.
If you find a claim that represents your system and want to use it in your design, write down the claim ID number and title in the Claim list form provided.
Continue to explore through the repository using the search/browse methods till you feel you have identified enough claims to design you system.
If you can not find an appropriate claim that represents a feature of your system, you may create the claim on the claim creation form.

When designing your system you should consider the following design aspects:

The notification systems critical parameters: interruption, reaction, and comprehension.
The method you use to deliver your notification to the users.
The type of notification system you are designing.

B.2 Claim Collection Form

Group members:

Claims list form: use this form to list the claims you are using for your system. For claims that you created, enter the claim number marked on the claims creation form instead of the claim ID number required for claims from the library.

Claim ID	Claim name	Reused/created
1284568	Example claim name	reused

Appendix B: Reference Claim Study

B.3 Claim Creation Form

Group members:

Claims creation form: If you can't find claims for your system in the claims library, you can use this form to add claims that describe features of your system that you feel are important.

Claim 1

Title:

Feature:

Upsides:

+

+

+

Appendix B: Reference Claim Study

+

+

Downsides:

-

-

-

-

-

Claim 2

Title:

Feature:

Upsides:

+

+

+

+

+

Downsides:

-

-

-

B.4 Reference Claim Questionnaire

Claims Library Activity

Please enter your Group ID

Please enter your Group Member ID:

Describe the system you wanted to design for the station manager:

To what degree did you find the claims to be reusable?

- ☐ 0 - Not reusable at all
- ☐ 1
- ☐ 2
- ☐ 3
- ☐ 4
- ☐ 5
- ☐ 6
- ☐ 7 - Completely Reusable

What made you arrive at this conclusion?

How useful were the upsides and downsides of the claims you used for your design?

- ☐ 0 - Not useful
- ☐ 1
- ☐ 2

- 3
- 4
- 5
- 6
- 7 - Very useful

How clear were the claims?

- 0 - Not clear
- 1
- 2
- 3
- 4
- 5
- 6
- 7 - Very Clear

How well did the claims you reused represent your design?

- 0 - Not Representative
- 1
- 2
- 3
- 4
- 5
- 6

- [ ] 7 - Accurately Representative

Why did you create the claims you created?

To what degree do you find the method you used to find claims helpful?

- [ ] 0 - Not helpful at all
- [ ] 1
- [ ] 2
- [ ] 3
- [ ] 4
- [ ] 5
- [ ] 6
- [ ] 7 - Completely Helpful

Did the claims in the library give you ideas of what to design or did you follow your predetermined design vision and search for and/or create the required claims?

- [ ] 0 - Design Vision Driven
- [ ] 1
- [ ] 2
- [ ] 3
- [ ] 4
- [ ] 5
- [ ] 6
- [ ] 7 - Claims Availability Driven

Explain more about how you went about designing your system:

Overall, how useful was the claims library?

- [ ] 0 - Not useful at all
- [ ] 1
- [ ] 2
- [ ] 3
- [ ] 4
- [ ] 5
- [ ] 6
- [ ] 7 - Very useful

To what degree do you feel the organization of the claims in the library is helpful?

- [ ] 0 - Not helpful at all
- [ ] 1
- [ ] 2
- [ ] 3
- [ ] 4
- [ ] 5
- [ ] 6
- [ ] 7 - Completely helpful

How familiar were you with the Notification Systems domain BEFORE the activity?

- [ ] 0 - Not familiar

- 1
- 2
- 3
- 4
- 5
- 6
- 7 - Completely Familiar

How familiar were you with the Notification Systems domain AFTER the activity?

- 0 - Not familiar
- 1
- 2
- 3
- 4
- 5
- 6
- 7 - Completely Familiar

Vita

Alain Fabian

<u>Education:</u>

M.S Computer Science and Applications, Spring 2006
Virginia Polytechnic Institute & State University, Blacksburg, VA.
GPA: 3.72/4.00

B.S Computer Science, *Summa Cum Laude*, Spring 2004
M*inor:* Mathematics
Virginia Polytechnic Institute & State University (Virginia Tech),
Blacksburg, VA.
GPA : 3.94/4.00, Major Rank: 6 of 207

Physics, Technology, Engineering Sciences 1998-2000
Advanced Mathematics Institute Les Eucalyptus, Nice, France.

French Baccalaureate, Spring 1998.

<u>Publications:</u>

Fabian, A., Felton, D., Grant, M., Montabert, C., Pious, K., Rashidi, N., Tarpley III, A. R., Taylor, N., Chewar, C. M. and McCrickard, D. S. (2004). Designing the Claims Reuse Library: Validating Classification Methods for Notification Systems. *Proceedings of ACM Southeast Conference*, 357-362.

Lee, J.C, Lin, S, Chewar, C.M, McCrickard, D.S, Fabian, A., Jackson, A.(2004). From Chaos to Cooperation: Teaching Analytic Evaluation with LINK-UP. *Proceedings of the World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education*, 2755-2762.

Fabian, A., Wahid, S., Bhatia, S., McCrickard, D.S. (2006). Creating an Interactive Learning Environment with Reusable HCI Knowledge. *World Conference on Educational Multimedia/Hypermedia and Educational Telecommunications*, to appear.