A Critique of Design Approaches for Notification Systems

Goldie B. Terrell

Independent Study Report

Virginia Tech Computer Science Department

December 2004

Dr. D. Scott McCrickard, advisor

1. INTRODUCTION

The Fall 2004 Virginia Tech Undergraduate Research in Computer Science (VTURCS) class, taught by Dr. McCrickard, covered the following topics: problem, activity and information/interaction phases of design; scenario based design; interruption, reaction, and comprehension (IRC) values; stages of action; ubiquitous computing evaluation areas (UEAs) and participatory negotiation. This critique is my assessment and observations of how these design approaches worked for our project.

The class focused on improving the Buddy Bugs Phidgets project done by Susannah McPhail (now Tawse) at the University of Calgary. Links for downloading both her video and paper, as well as another web page are contained in [3]. Phidgets (physical widgets) [4], [6], [9] and [11] provide software and circuit boards to allow off-screen physical devices to be developed relatively easily by computer scientists. The Buddy Bugs project used glass "bugs" on a paper-maché "leaf", which by turning, lighting up and sensing pressure, provided a physical substitute for a person's buddy list on MSN Instant Messenger.

Examples of other Phidget projects may be found in [3] and [5]. A real-world Phidgets application, involving HVAC may be found at [7]. Alternative computer controlled physical device systems are detailed at [12] and [13].

In designing our Phidget interface, we used the general HCI approach and vocabulary found in Rosson and Carroll's textbook [15]. This includes scenario-based design, claims, phases (stages) of design, and stages of action. In addition, we employed other design techniques, including participatory negotiation [16] and ubiquitous computing evaluation areas (UEAs) [1]. For a description of the project and an overview of our general design process and decisions, see [17].

Notification Systems

Notification systems are those applications that are not the primary task of the user. They provide reaction to and comprehension of valued information in an efficient and effective manner without introducing unwanted interruption to a primary task.

I would argue that there are two classes of notification systems, which should be looked at differently: alarms, and continuous (or ambient) displays. In alarms, the user virtually ignores the application until it puts up an alarm of some sort. A continuous display, such as a stock ticker, shows information of possible interest as its values change and the user would be paying some occasional or peripheral attention to the application. There can also be applications that combine those two. The fact there are dual combinations argues for treating these all as one class of applications, but I think time could well be spent in investigating these categories separately.

My Background

My biases are formed in part by my experiences, which include an undergraduate degree in mechanical engineering and graduate classes in computer science taken between 1986 and 1989 at Virginia Tech. My knowledge of real-life design processes and negotiations

comes from my community work over a period of 16 years.

When I returned to school in 2003, I took Usability Engineering from Dr. Hartson. He does not use the Rosson and Carroll book; although his approaches are similar, the vocabulary and categorization is somewhat different. His class notes can be found at [18].

2. PHASES OF DESIGN: PROBLEM, ACTIVITY AND INTERACTION/INFORMATION

The phases of design listed in Rosson and Carroll [15] include problem, activity, information, interaction, prototyping, evaluation and documentation. (Rosson and Carroll call them stages, but I will refer to them as phases to avoid confusion with stages of action.) While our class went through most of the phases while doing the project, our concentration was on the first four.

The problem phase involves exploring the problem that you hope to solve with your project. This is done by investigating the current way that the user does the activity, pointing up the shortcomings that you hope to better and taking note of any desirable features that you should include. Ideally, this phase involves considerable user input, as what the user wants may not be what you suppose.

In the activity phase, the overall metaphor and broad ideas of the project are devised. This is not the detail phase, and ideally, little time is spent thinking about how the project could be implemented. Thought is given to how the user should model the system and whether the metaphor aids that modeling.

The details are worked out in the information and interaction phases, information concerning itself with how the information is communicated to the user and the interaction handling the way the user can communicate with the program.

Ideally, at each point in the design, the users should be consulted again to verify that the design meets the users most important (but sometimes unarticulated) desires for the project. This user input may necessitate rethinking the design.

Overlaps Between the Three Phases

We spent the most time on the problem phase, as we were learning the various systems that we would use for the other phases too. We also held a participatory negotiation session to get user input. Preparing for that and analyzing the results took considerable time. Spending time on the problem phase, in my opinion, is time well spent, as knowing what the users want is an essential part of the project.

I feel that the problem phase is fairly well defined. There is some overlap into the activity phase, especially if you start to think about the possibilities of satisfying the users wants, rather than just gathering information and analyzing it to find salient features. But overall, the idea of gathering information about what is currently being used and what the users want instead is fairly well defined.

In the activity phase, it is easy to start also thinking about the details of transferring information and how interaction might take place for a given metaphor. So there is not necessarily a clean line between activity and information/interaction. Indeed, a number of the claims that we had for the activity phase could be classed easily as "high level detail". We also had trouble with the idea of a metaphor. It was easy to keep referring back to Susannah's "bugs on a leaf", and the "room map" metaphor that we settled on is clear. What is not so clear is what the metaphors are for other projects. I'm not sure that I can pick out the metaphor for say, MSN Instant Messenger.

Rosson and Carroll separate the information and interaction phases and we lumped them together. By the point at which we were considering those phases, we were running out of time. Even without a time crunch, it is natural to run together thinking about the details of the information display and what detailed actions the user would take to use the system. It would be hard to strictly separate these. You should start with the information display, as that comes from the metaphor, and without the display, it is hard anticipate how the user could be guided to use the system. But after the initial information details are worked out, the interaction details become intermingled.

The Phases Concept Overall

The Rosson and Carroll model matches fairly closely with what I learned from Dr. Hartson, although Rosson and Carroll fail to mention actual construction of the project! Dr. Hartson charted the phases as systems analysis, problem domain design, software design, software implementation, and software testing. In practice, this translates to problem, activity, information/interaction with prototypes, implementation, and evaluation, so the two systems worked out very similarly. Dr. Hartson's phases may be found in section 2 (02-LC.pdf) of his class notes [18].

I feel that the phases in either manifestation are a good way to break the design into parts. They made sense to me during my projects in Dr. Hartson's class and it was rather easy to follow the phases for this project, with the exception of the blur from activity to information/interaction. Each of the phases flows fairly well into the next phase, providing the information needed to start the next phase.

The claims that we generated in each of the phases did not have much to do with each other. It seemed that at each phase, there were new things to consider (more detailed) that did not necessarily follow from the previous claims. Perhaps this was difficult for me because I was unaccustomed to working with claims, as Dr. Hartson did not frame things in this way.

Although we gave scant notice of the prototype phase, I did draw a true-to-size paper diagram, which we spent a class period discussing and altering. I would have liked to have feedback from users about the prototype. We should at least have talked to some of the lab personnel who were around during our discussion of the prototype. In the press of time, we neglected this important part of the prototype phase.

Our evaluation was very casual, but it seemed that the project was a positive addition to the lab. The documentation, in the form of the paper about the project [17], is fairly complete.

3. SCENARIO-BASED DESIGN AND CLAIMS

Scenario-based design is carried out by creating scenarios at each design phase that cover likely uses of the project. The scenarios should cover the use by different classes of users, such as casual users, intensive users, and information administrators. Claims and their corresponding pros and cons are developed for each scenario.

The scenarios help the designer to think about the likely way in which the project will be used, and should include information about the environment that the user will work in and the people the user will interact with in completion of a task.

A claim is a possible feature of the project and can be an artifact, such as the color of a button, or an activity, such as pushing a button. Scenarios, user classes, stages of action, UEA measures or other means should be used to ensure that all the pros and cons of each claim are itemized.

Scenarios Work Well

In my opinion, scenarios are good design tools for all of the reasons that Rosson and Carroll list on pp. 20-24. For me, the obvious strength is in keeping the designers focused on the user's experience. Coming up with a set of scenarios at each design phase forces the designer to consider how someone else will view the project.

Rosson and Carroll highlight how scenarios give designers and stakeholders a common language. The concreteness of the scenario allows much more specific input from the stakeholders than might usually be given when faced with a set of objectives written from the designers point of view and using the designers language.

I was surprised at how well the scenario of a given phase provided a springboard to the next phase. In Dr. Hartson's course, we had not handed off the scenarios between the phases; we had designed new scenarios for testing of our prototypes and I did not directly connect those to the problem scenarios.

Claims Are Documentation

In my opinion, claims for the scenarios are most useful as a way of documenting the discussions that go on in design sessions about tradeoffs. Since they capture various features that might be implemented, features that ultimately were rejected still have their place in the paper trail. These may be useful if a redesign is necessary. The associated pros and cons provide the reasons that design choices were made.

As with much documentation, if left to our own devices, we probably would not have used written claims. Because we were required to make claims as homework before meeting and discussing them, they provided a means of capturing individual thinking to be discussed at the joint meeting.

In addition, the claims generated could be very useful when the next expansion of the system is designed. We considered many things that ultimately were not applicable to our final design, but are applicable to the design of an expansion. The scenarios also continue to serve as guides for development.

Notification Systems
Since notification systems are a subset of computer interfaces, I feel that scenario based design and claims serve well for notification system design. In particular, the claims can be used rather effectively to designate the different levels of interruption that might be desired. In the event that the desired level of interruption changes, it is easy to alter the claims slightly as to pros and cons and the weight given to each.

4. INTERRUPTION, REACTION, AND COMPREHENSION

Interruption, Reaction, and Comprehension (IRC) values are sets of values from 0 to 1 for each of the three components. In design of notification systems, the target IRC values for the project are determined from user input and other considerations in the problem phase, and are modified throughout the design process as more information and specifics are available. Problem claims are given IRC values also, so that an attempt to match features to the desired IRC levels can be made. As design proceeds through other phases, claims in those phases are given IRC values so that the features will support the target values.

IRC values are intended for notification systems, mainly because of taking interruption into account, which you would not do for a primary task. Since it was, at times, hard for me to determine if we actually were designing a notification system, rather than a primary task, there were times when I questioned whether IRC was appropriate to use for our project.

The target IRC values will usually not be (1, 1, 1). In many notification systems, a low I may easily be desirable. It makes sense to me to combine IRC values with stages of action, as a transition from a peripheral task to a primary one takes place with many notification systems. Different IRC values are often desired between a peripheral task and a primary one. It could be argued that once the task becomes primary, IRC should not be used.

My Difficulties with IRC Implementation
In spite of the time spent in this class on IRC values, I continue to be puzzled by them. I understand in theory why you would want to pay attention to interruption, reaction and comprehension in notification systems and I think I understand how IRC values are intended to help. Perhaps our project was not enough of a notification system for IRC to make sense.

I had problems assigning IRC values. The fact that in many cases the class participants chose vastly different values indicates that it is a highly subjective endeavor. However, since I was usually at odds with my two classmates, who often used similar values, a case could be made that I do not understand the system properly.

For me, one of the main problems was the lack of standards for the values. Our attempt to make standards seemed to come to a consensus after a while, but I found even then that it was hard to apply those standards in many cases. The rationale for the values kept eluding me.

I had trouble just because there are so many details to be considered, with each changing

the answer. Even two scenarios that feature the same person and differ only in the primary task might prompt different values. How interested the user is in the information given by the notification system at a given time could also generate different values.

After using IRC values in the problem phase, even though Dr. McCrickard urged us to use them in subsequent phases, somehow we never got around to it until we were forced to for the final paper. I think that they were seen as having little utility. This may be because what we designed was not strictly a notification system. In reviewing, I continue to think that their utility, even for an expansion, would be low.

Improvements to IRC

When you arrive at the information/interaction phase, where there are detailed claims, you could have a library of IRC values for them, e.g. a red display is ten percent more noticeable than a green one. However, I would imagine that the values in problem claims would not fit into a library. LINK-UP, being developed at Virginia Tech describes a claims library [8] [9],but I have no direct experience with it.

Where IRC values became most difficult for me in practice was the vast difference between user levels of attention when considering an alarm system. When an alarm sounds, the way in which the application is viewed changes, sometimes dramatically. The application may become the primary task, a situation not accounted for in our definition of a notification system. Even a continuous display has separate values when the user is absorbed in his primary task and when he notices the information, however briefly.

Integration with stages of action might provide a solution to this, as it allows for the difference in levels of attention. In practice, when I tried this, it seemed to help me, but it was a slow process because of the level of detail. Even then, the values that I produced as a result did not agree with the other students' values.

There are important considerations in notification systems that are not covered by IRC. The chief one of these is aesthetics. This indicates to me that another system, such as UEA measures, should be used in addition to IRC values.

Using IRC as a Basis for Rating Claims Pros and Cons

In forming the pros and cons for a claim, it was sometimes difficult to determine whether the point (the pro or con itself) was positive or negative. For some stated goals it might be positive, but for others, negative. For instance, a claim might draw your attention to the spatial aspects of the display- whether that is good or bad depends on the emphasis desired.

I had wanted a sliding scale for the pros and cons instead of just positives and negatives. With a sliding scale, not only is there more information about how strongly positive or negative a point is, but if the situation changes, adjustments are easy. The scale can be slid so that the values remain accurate for the new situation or the values can be interpreted against a new target. This is what IRC does. The values are absolute on the scale, but whether or not they are pro or con depends on what the target value is.

I had attempted to merge IRC with pros and cons in the problem phase. As with the merge with stages of action, there was too much detail and I felt that others saw little value in the combination.

5. STAGES OF ACTION

Carroll and Rosson divide one application interaction step into six stages: perception, interpretation, making sense, system goal, action plan, and execution. The first three form the Gulf of Evaluation, which is concerned with how the user views the system, and the second three form the Gulf of Execution, or how the user plans and executes his reaction.

As the user goes through the stages, he is ready for different pieces of the application. First, the user must perceive the discrete items on the screen, then interpret what kind of items they are, such as a table. In making sense, the user relates the information to his particular task. Deciding what the system needs to do to carry out his wishes, the user must look at a system goal. From this he forms an action plan and then does the physical actions needed to execute the plan.

Dr. Hartson teaches a version of the stages of action, but he does not use the same terms, nor talk about Gulfs. Hence, since the class assumed that we knew this material and only the names were mentioned without any definitions, it was quite a while until I realized what the subject was.

By looking at all the ways in which the project will be viewed by the user, the designer can more completely anticipate what things the user will need. I found it helpful in devising the pros and cons of the claims.

Stages of action have particular advantages when devising IRC values for an alarm application. The whole purpose of the application may change when the alarm sounds and the stages of action help to define this transition.

I would argue that in this case, there is a seventh stage worth taking into account. When an alarm sounds, the cycle starts with perception of the alarm. So what comes before that, when the application is being "ignored"? Its characteristics during this time comprise a dormant stage.

While the design of many types of projects can be served well by the stages of action, the continuous display application is not, as the user is not interacting with the application.

6. UBIQUITOUS COMPUTING EVALUATION AREAS

Ubiquitous Computing Evaluation Areas (UEAs)[1] form a framework that delineates items to be considered in evaluation of any computer application, but especially applicable to off-the-desktop (ubiquitous) applications. Since these measures are designed for off-the-desktop applications, they apply particularly well to our project.

The items are grouped into several areas which each have several metrics. For instance, the area of appeal has fun, aesthetics and status metrics. Each metric has several

associated measures, which provide more concrete ways to measure the metrics, e.g. typical time spent setting up the technology.

Some of the measures are fairly specific, using tangible quantities. Other measures deal with intangibles that are hard to place numbers on. This is regrettable, as quantification aids analysis, but trying to quantify something such as social acceptance is hard.

The interaction, attention, and application robustness areas provide good measures, but these can be found in many other systems. The areas of appeal, impact and side effects, and adoption provide good measures and these areas are not often thought of in other systems. The conceptual models and invisibility areas had particularly weak measures that added little advantage. I would like to see the framework expanded.

The examples in Scholtz and Consolvo's paper are all written as if they had gathered comments from users about projects, then had assigned metrics to them. This may be how they arrived at their framework, but it is backwards from the way that UEAs should be applied. The measures help design what information should be gathered, which in turn produces evaluations of the project.

Scholtz and Consolvo tout their framework as something to use in the evaluation phase. I think that one of the strong points of the system is its ability to generate metrics for the design. This should happen in the problem phase and be carried forward throughout the phases, as the evaluations within each phase are performed. The particular project metrics drive the design, not merely appear at the end of the design process.

In writing scenarios, consideration of the measures would lead to more complete descriptions of the environment and interaction with the system. This in turn would lead to generation of more pros and cons of claims. Having this itemization of measures is a useful tool that encourages more complete design considerations.

To me, interruption, reaction and comprehension would be metrics if integrated into the UEA framework and it would be helpful if there were measures associated with IRC. The values of IRC are somewhat analogous to the measures, as they attempt to quantify characteristics. Perhaps I would have a better grasp of IRC values if I had associated measures to consider.

One might think that the UEAs of interaction and invisibility would speak to reaction and interruption. However, none of the measures and metrics in any category include interruption, reaction and comprehension considerations. I had thought that the UEA framework was fairly complete, but this omission only points up that completeness is ellusive.

Both stages of action and UEA measures help to produce claims and their pros and cons, but they seem to be rather disparate. One might use both systems to formulate claims pros and cons, but I would not be able to combine these two systems into one system. The measures seem especially suited to the activity phase of design and the stages of action would seem to help more with the details at the interaction/information phase.

Having these measures in hand would have helped guide the evaluation phase of the project that my team designed in Dr. Hartson's course. Although we did not formally evaluate our project in Dr. McCrickard's class, had we done a formal evaluation, I would have made use of the measures in deciding what information to gather. The proper use would be to guide the claims pros and cons in the previous phases, which in turn would be used to develop the metrics by which the project was evaluated.

7. PARTICIPATORY NEGOTIATION SESSION

The participatory negotiation session was performed in order to gather information about what the stakeholders desired. Ali Ndiwalana, who is researching Particiatory Negotiation Sessions, helped us prepare for the session. We refined our root concept, problem scenarios and problem claims to present to the participants for their comment.

During the Session

Ali presented the root concept and problem scenarios and discussion was encouraged while the class took notes. Some questions were directed to the class members, which they answered. After discussion, the participants filled out a questionnaire (see appendix). It asked for comments on the completeness and accuracy of the root concept, the salient features and accuracy of the problem scenarios, rankings of the pros and cons in each problem claim and any desired additional or changed pros and cons. Wanting to give participants time to fill out the questionnaire while staying to their time commitment of one hour, Ali chose to omit discussion of the problem claims. We were rewarded by more written comments than I would have anticipated.

The dual approach of discussion and questionnaire helped to gather more information than we would have gotten through either method alone. Since some people are not comfortable expressing themselves in a meeting, those people had the option to express themselves on paper. The discussion provided the ability for questions from the participants, which gave us insight into what assumptions they made and where their concerns lay. In turn, our questions to the participants helped to clarify their ideas.

I was particularly struck by the fact that although there were about fourteen people in the room, most did not speak at all, and the majority of the talking was done by four people. I was prepared for the fact that most people do not like to fill out questionnaires, and expected at least some to give the questionnaire only minimal answers. However, when the questionnaires were viewed, I was pleasantly surprised to find how completely almost all of the participants had filled them out. I took this as an indication that we had made them sympathetic to our goals and communicated our esteem of their input. An alternative interpretation is that Dr. McCrickard made their participation a recognized part of their duties.

I feel that the session went well. I saw little wasted time- all time was spent in communicating our ideas or in receiving theirs. Distributing the material ahead of time certainly helped spend a minimum of time explaining our ideas, although not everyone will read material ahead of time, either because of time constraints or because they prefer to hear the material. Some of the comments we received during the discussion were repetitive, but that served to reinforce how important the issue was to the participant.

Shortcomings

Dr. McCrickard invited people to participate who were involved with the lab, either as a regular worker, an occasional worker, or a professor. We had hoped to cover all the stakeholders that we had identified: the regular user, professors in the lab, people looking for regular users, and administrators. Unfortunately, obtaining participants from the latter two groups was impracticable. Dr. McCrickard attended, but did not participate and we had no other professors attend. This means that all of our responses came from regular or occasional lab users.

We used the results of the session to alter our problem scenarios and claims to their final state, prior to moving to the activity phase. I found it awkward and confusing to try to use these results, but also allow for Dr. McCrickard's stated goals for the project. If he or another professor had participated, I think it would have been easier to follow the expected course of action. This may mirror a real-world situation, as information may not be available from all stakeholders.

We did our best to put together appropriate claims for the participatory negotiation session, but the amount of time we could reasonably ask the participants to give meant the number was much smaller than I would have liked to have feedback on.

The information collected in the questionnaire about the ranking of the pros and cons was of dubious merit. I feel it would have been far more instructive to ask for a rating for each pro or con. That is, asking the participants to rank the importance of each pro or con on a scale of 1-5. This would have given us a sounder basis for drawing conclusions about the importance of different features.

The analysis (see appendix) of the questionnaires, prepared by Ali, was easier to interpret when the summary information was put on a questionnaire. Previously, lacking the questions along with the answers, interpretation of what was being measured was difficult.

It would have been nice to have a discussion of the claims. This would have involved extra time, so might not be feasible. I would like to try having a session for an hour and a half next time- an hour for discussion and 30 minutes for the questionnaire. It would be interesting to see how the extra half hour time commitment affects participation. I think it would be a poor idea to ask the participants to fill out the questionnaire later, as I feel many people would not do it in that case.

REFERENCES

1. Scholtz, J., and Consolvo, S. Toward a Framework for Evaluating Ubiquitous Computing Applications. IEEE Pervasive Computing, volume 3, number 2, April-June 2004, 82-88.
2. McPhail, S. (2002) Buddy Bugs: A Physical User Interface for Windows® Instant Messenger. Western Computer Graphics Symposium (Skigraph'02), March.
3. http://grouplab.cpsc.ucalgary.ca/phidgets/gallery/index.html
4. http://grouplab.cpsc.ucalgary.ca/phidgets/

5. Greenberg, S. (2004) Collaborative Physical User Interfaces. Report 2004-740-05, Department of Computer Science, University of Calgary, Calgary, Alberta, Canada T2N 1N4, March.
6. Saul Greenberg and Chester Fitchett (2001). Phidgets: Easy development of physical interfaces through physical widgets. In the Proceedings of the 14th Annual ACM Symposium on User Interface Software and Technology, Orlando, FL. pp. 209-218. Available at http://www.cpsc.ucalgary.ca/grouplab/papers/
7. http://diy-zoning.sourceforge.net/
8. Catherine Payne, C. F. Allgood, C. M. Chewar, Chuck Holbrook, and D. Scott McCrickard. "Generalizing Interface Design Knowledge: Lessons Learned from Developing a Claims Library." In *Proceedings of the 2003 IEEE International Conference on Information Reuse and Integration (IRI '03)*, Las Vegas NV, October 2003, pp 362-369.
9. Greenberg, S. and Fitchett, C. (2001) **Phidgets: Incorporating Physical Devices into the Interface.** In M. Newman, K. Edwards and J. Sedivy (Eds) *Proceedings of the Workshop* on Building the Ubiquitous Computing User Experience. (Held at ACM CHI'01, Seattle). http://grouplab.cpsc.ucalgary.ca/papers/
10. C. M. Chewar and D. Scott McCrickard. "Claims Reuse for Notification Systems Design: LINK-UP Vision and IRC Equations." Technical Report TR-03-29, Computer Science, Virginia Tech, 2003.
11. Greenberg, S. and Fitchett, C. (2001) **Phidgets: Easy Development of Physical Interfaces through Physical Widgets.** *Proceedings of the UIST 2001 14th Annual ACM Symposium on User Interface Software and Technology,* November 11-14, Orlando, Florida, p209-218, ACM Press. http://grouplab.cpsc.ucalgary.ca/papers/
12. Sorenson, Franklin E. PODs: Physical Object Devices, PhD thesis at Brigham Young University, Department of Computer Science, February 2004
13. D. Scott McCrickard, Dillon Bussert, and David Wrighton. "A Toolkit for the Construction of Real World Interfaces." In *Proceedings of the ACM Southeast Conference (ACMSE '03)*, Savannah GA, March 2003, pp. 118-123.
14. Matthews, T., Dey, A., Mankoff, J., Carter, S., Rattenbury, T. A Toolkit for Managing User Attention in Peripheral Displays. *Proceedings of the ACM Symposium on User Interface Software and Technology (UIST 2004),* November 2004, 247-256.
15. Rosson, M. and Carroll, J.M. *Usability Engineering: Scenario-Based Development of Human-Computer Interaction*. Morgan-Kaufman, San Francisco, USA, 2002.
16. Ndiwalana, A., Kampanya, N., McEwan, I., Chewar, C.M., McCrickard, D.S., and Pious, K. A Tool for Participatory Negotiation: LINKing-UP Participatory Design and Design Knowledge Reuse. *Proceedings of the Eighth Biennial Participatory Design Conference (PDC '04),* volume 2, July 2004, 118-122.
17. Heir, Manveer , Harish Hoon, D. Scott McCrickard, and Goldie Terrell. *Online Enlightenment: A Phidget Notification System for Online Status* Virginia Tech Department of Computer Science Technical Report TR-04-22, December 2004.
18. http://courses.cs.vt.edu/~cs5714/spring2004/calendar.html

APPENDIX

1. The Peripheral Display Toolkit

The Peripheral Display Toolkit (PTK) [14] was considered for use in our project. There are three policies in effect for each project implemented with this system: abstraction, notification, and transitions. The abstraction policy uses supplied functions to convert input data to objects such as numbers, switches, images, etc. depending on the desired output. The notification policy sets thresholds for the data, e.g. the proper number of lights to display for a given input level. The transition policy coordinates the transitions from one display setting to another.

The attractive thing about the PTK was that it allows for modular change of the system. If you decided to use a slider that would go to the appropriate position instead of a row of lights, you could easily change only the input policy.

We had hoped that the toolkit would help us interface with the Phidgets, but it appeared on closer examination that their interface did not provide any modules of assistance to us. Our project did not require complicated sets of lights or other output, nor thresholds, nor transitions. Combining our needs with the PTK interface would add more complication to our project.

Category: (choose one) lab user | lab visitor | lab professor | TA seeker User ID:_____

Root Concept Questions

Is the project high-level vision realistic?

<Strongly Disagree> <Slightly Disagree> <Neutral> <Slightly Agree> <Strongly Agree> <Unsure>

Comments:

_____
_____
_____
_____
_____

Does the list of stakeholders cover all the kinds of people who will be affected by (benefit or suffer from the successful or failed) implementation of our proposed system?

<Strongly Disagree> <Slightly Disagree> <Neutral> <Slightly Agree> <Strongly Agree> <Unsure>

Comments:

_____
_____
_____
_____
_____

Is the list of starting assumptions sufficiently comprehensive (i.e. it takes into account all the assumptions that you would make for the proposed system)?

<Strongly Disagree> <Slightly Disagree> <Neutral> <Slightly Agree> <Strongly Agree> <Unsure>

Comments:

_____
_____
_____
_____
_____

This system will be very useful in the lab
<Strongly Disagree> <Slightly Disagree> <Neutral> <Slightly Agree> <Strongly Agree> <Unsure>

Comments:

_____
_____
_____
_____
_____

Problem Scenario Questions

SCENARIO - Lab users

Please highlight all Interesting Scenario Features using the colored pen given to you.

"Interesting features" of a scenario are those that influence the actors' abilities to carry out their activities within a scenario. While the scenario will usually convey just one consequence of a given feature, they do usually have more (both positive and negative), resulting into some form of tradeoffs.

Does the problem scenario accurately describe most of the activities involved in the current practice (way of doing things)?

<Strongly Disagree> <Slightly Disagree> <Neutral> <Slightly Agree> <Strongly Agree> <Unsure>

Comments:

_____
_____
_____
_____
_____
_____

Does the problem scenario sufficiently highlight the challenges inherent in the current practice?

<Strongly Disagree> <Slightly Disagree> <Neutral> <Slightly Agree> <Strongly Agree> <Unsure>

Comments:

_____
_____
_____
_____
_____
_____

SCENARIO - Dr. McCrickard

Please highlight all Interesting Scenario Features using the colored pen given to you.

Does the problem scenario accurately describe most of the activities involved in the current practice (way of doing things)?

<Strongly Disagree> <Slightly Disagree> <Neutral> <Slightly Agree> <Strongly Agree> <Unsure>

Comments:

_____
_____
_____
_____
_____
_____

Does the problem scenario sufficiently highlight the challenges inherent in the current practice?

<Strongly Disagree> <Slightly Disagree> <Neutral> <Slightly Agree> <Strongly Agree> <Unsure>

Comments:

_____
_____
_____
_____
_____
_____

SCENARIO – TA Seekers

Please highlight all Interesting Scenario Features using the colored pen given to you.

Does the problem scenario accurately describe most of the activities involved in the current practice (way of doing things)?

<Strongly Disagree> <Slightly Disagree> <Neutral> <Slightly Agree> <Strongly Agree> <Unsure>

Comments:

_____
_____
_____
_____
_____
_____

Does the problem scenario sufficiently highlight the challenges inherent in the current practice?

<Strongly Disagree> <Slightly Disagree> <Neutral> <Slightly Agree> <Strongly Agree> <Unsure>

Comments:

_____
_____
_____
_____
_____
_____

SCENARIO – Visitors

Please highlight all Interesting Scenario Features using the colored pen given to you.

Does the problem scenario accurately describe most of the activities involved in the current practice (way of doing things)?

<Strongly Disagree> <Slightly Disagree> <Neutral> <Slightly Agree> <Strongly Agree> <Unsure>

Comments:

_____
_____
_____
_____
_____
_____

Does the problem scenario sufficiently highlight the challenges inherent in the current practice?

<Strongly Disagree> <Slightly Disagree> <Neutral> <Slightly Agree> <Strongly Agree> <Unsure>

Comments:

_____
_____
_____
_____
_____
_____

Problem Claims Questions

For each of the following claims, we would like you to help us with a number of things:

1. Help us identify more Pros and Cons that we could have omitted. Write these in the section provided below each claim
2. Help us identify the most important two or three of the Pros and Cons to address
    - Mark the most important Pro as P1
    - Mark the second most important Pro as P2, etc
    - Similarly for Cons, Mark them as C1, C2 and C3

   Optionally, if you feel that any Pro or Con needs to be re-written, you can cross it out below and re-write it in the space below the claim. Draw a line to link the two (the old one you crossed out and the new one you re-wrote)

Notification appears on computer screen

+ interruption is easy to notice, if working at the computer	
+ reaction is easy since you are already at the computer	
+ notification using text provides a lot of information	
+ can use of keyboard and mouse to find out more information	
– takes up screen space	
– must be at computer, or notification must be persistent for you to see it	

More Pros/Cons:

_____
_____
_____
_____
_____
_____
_____
_____

Finding all the lab users using MSN

+ easy to get status information about all at once	
+ intuitive to respond if needed	
– requires access to a computer	
– becomes your primary task	
– in order to track the lab's status, must keep checking the buddy list	
– no information of how long the particular user has been in that state for	
– no history of information regarding the state of the user over a period of time	

More Pros/Cons:

_____
_____
_____
_____
_____
_____
_____
_____

Using a tone for notification

+ no interference at all prior to notification	
+ tone easy to hear if at the computer	
+ tone is recognizable as MSN tone, but only because of conditioning to the program	
– tone has no direct association with user signing on	
– tone no help in planning or executing action	
– can't hear tone if not in room	

More Pros/Cons:

_____
_____
_____
_____
_____
_____
_____
_____

Using a transparent, fading window for notification

+ window may be recognized as MSN notification with experience	
+ text in window can be used to provide crucial info	
+ clicking on text for details is intuitive	
+ attention is drawn to window where action should take place	
– fading window requires fast action	
– may miss to see window if focused on another part of the screen	
– non-persistence means user must notice it at the time it comes up	
– nothing on window indicates it is an MSN window	

More Pros/Cons:

_____
_____
_____
_____
_____
_____
_____
_____

No dynamic interactive representation of lab information

+ decreases clutter that can distract lab users	
+ more desk and wall space can be dedicated to other (less dynamic but more information-rich) displays (like posters)	
+ allows for a rich, easily comprehensible array of various representations to be used as different forms of information	
+ discourages outsides from entering lab (good for when you want to get work done)	
– provides no sense of direction for a visitor	
– can lead to a bland and boring lab area	
– visitors leave with no good understanding of what research lab does	
– lessens the number of demo-able, interesting, eye-catching, non-traditional displays	

More Pros/Cons:

_____
_____
_____
_____
_____
_____
_____
_____

user		1	2	3	4	5	6	7	8	9	10						
root concept												average	stdev	a	b	c	***user***
	1	5	5	4	4	4	5	5	5	2	5	4.333	1.00	3.33	4.333	5.33	strong agree
	2	5	5	4	5	5	5	4	5	2	5	4.444	1.01	3.43	4.444	5.46	strong agree
	3	4	4	4	2	2	4	3	4	4	2	3.222	0.97	2.25	3.222	4.19	slight agree
	4	4	4	4	4	4	3	4	4	3	2	3.556	0.73	2.83	3.556	4.28	slight agree
scenario 1																	
	5	5	4	5	4	4	5	4	4	4	5	4.4	0.52	3.88	4.4	4.92	strong agree
	6	5	4	5	5	4	4	5	4	5	5	4.6	0.52	4.08	4.6	5.12	strong agree
scenario 2																	
	7	5	5	-	4	3	2	4	4	4	4	3.889	0.93	2.96	3.889	4.82	slight agree
	8	5	4	-	4	4	5	5	5	5	5	4.667	0.50	4.17	4.667	5.17	strong agree
scenario 3																	
	9	4	5	5	5	4	5	4	5	4	4	4.5	0.53	3.97	4.5	5.03	strong agree
	10	3	5	5	5	4	5	3	5	5	5	4.5	0.85	3.65	4.5	5.35	strong agree
scenario 4																	
	11	4	5	5	4	3	5	-		3	2	4	3.889	1.05	2.83	3.889	4.94	slight agree
	12	5	5	5	2	2	5	4	5	3	4	4	1.25	2.75	4	5.25	slight agree

claim 1/5											1st	2nd	3rd	**score**
pro1	3	1	1	1	2	1	1	2	1	1	7	2	1	26
pro2	2	2	2	2	3	2	2	3	1	2	1	7	2	19
pro3	1	3	-		3	1	4	3	1	1	3	4	0	4	16
pro4	4	4	-	-		4	3	4	4	2	-	0	1	1	3
con1	1	2	-		2	1	2	2	1	2	3	3	5	1	20
con2	2	1	1	1	2	1	1	2	1	1	7	3	0	27
claim 2/5														
pro1	1	1	1	1	1	1	1	1	1	2	9	1	0	29
pro2	2	2	-	-		2	2	2	2	2	1	1	7	0	17
con1	5	1	-		2	1	4	4	5	1	-	3	1	0	11
con2	1	2	-		3	3	5	1	4	4	1	3	1	2	13
con3	2	5	-		1	4	3	2	2	2	-	1	4	1	12
con4	3	3	-	-		2	1	3	3	5	-	1	1	4	9
con5	4	4	1	-		1	2	3	1	3	-	3	1	2	13
claim 3/5														
pro1	1	1	-		1	2	1	2	1	1	1	7	2	0	25
pro2	3	2	-		3	1	2	3	3	3	3	1	2	6	13
pro3	3	3	1	2	3	3	1	2	2	2	2	4	4	18
con1	1	2	-		2	3	3	2	1	2	-	2	4	2	16
con2	1	3	-		1	3	2	3	3	3	3	2	1	6	14
con3	1	1	1	-		1	1	1	2	1	-	7	1	0	23
claim 4/5														
pro1	4	2	-	-		4	4	2	4	2	-	0	3	0	6
pro2	2	3	1	2	3	2	3	2	3	-		1	4	4	15
pro3	3	4	-		3	2	3	3	3	1	3	1	1	6	11
pro4	1	1	-		1	1	1	1	1	4	1	8	0	0	24
con1	1	3	-		2	1	4	1	3	2	2	3	3	2	17
con2	2	2	-		3	1	1	1	2	4	-	3	3	1	16
con3	3	1	1	1	1	2	1	1	3	2		6	2	2	24
con4	4	4	-			2	3	2	4	1	3	1	2	2	9
claim 5/5														
pro1	2	3	-		1	1	1	2	2	-		3	3	2	17
pro2	2	2	-		2	2	3	3	3	2	2	0	6	3	15
pro3	1	1	-		3	4	2	4	1	3	1	4	1	2	16
pro4	-		4	1	-		3	4	3	4	1	-	2	0	2	8
con1	1	2	-	-		1	3	2	2	1	-	3	3	1	16
con2	2	4	-	-		1	2	2	4	2	1	2	4	0	14
con3	3	1	2	1	4	4	3	1	4	-		3	1	2	13
con4	4	3	1	2	4	1	4	3	3	3		2	1	4	12

P1, P2, P3, indicates the ranking of the Pros for a given claim. P1 is the most important, followed by P2 and P3. They should be maintained in this order.

C1, C2, C3, indicates the ranking of the Cons for a given claim. C1 is the most critical, followed by C2 and C3. They should be eliminated or mitigated in this order.

The score appears in the brackets to help provide an indication of relative importance between the Pros and Cons. It is derived from how many first(3 points), second(2 points) or third(1 point) place rankings that a given Pro/Con received from the 10 users who participated in the negotiation session. The highest score is 30 points, if a Pro/Con received 10 first place rankings. The lowest score is theoretically 0 points, if a Pro/Con did not receive any rankings at all

Italics denote additions to what we had given the participants.
---

1. Notification appears on computer screen
[a] + interruption is obvious, if working at the computer **P1 (26/30)**
[b] + reaction is easy since you are already at the computer **P2 (19/30)**
[c] + notification using text provides a lot of information **P3 (16/30)**
[d] + use of keyboard and mouse well-known
[e] - takes up screen space **C2 (20/30)**
[f] - must be at computer, or notification must be persistent **C1 (27/30)**
+ can easily have variable levels of comprehension
+ can easily reply in context
+ does not disrupt other lab users
- must have an MSN account

2. Finding all the lab users using MSN
[a] + easy to get all the information you need at once **P1 (29/30)**
+ provides a good overview of information about all users
[b] + intuitive to respond if needed **P2 (17/30)**
[c] - requires access to a computer **C3 (11/30)**
[d] - becomes your primary task **C1 (13/30)**
[e] - in order to track the lab's status, we must keep checking the list **C2 (12/30)**
[f] - no history of information regarding the state of the user over a period of time
[g] - no information of how long the particular user has been in that state for **C1 (13/30)**
+ no need to develop new system from scratch
- users must keep track of themselves actively
- not everyone has an MSN account

3. Using a tone for notification
[a] + no interference at all prior to notification **P1 (25/30)**
[b] + tone easy to hear if at the computer **P3 (13/30)**
[c] + tone is recognizable as MSN tone, but only because of conditioning to the program **P2 (18/30)**

[d] - tone has no direct association with user signing on **C2 (16/30)**
[e] - tone no help in planning or executing action **C3 (14/30)**
[f] - can't hear tone if not in room **C1 (23/30)**
+ does not use up screen space or grab input focus from other applications
- computer must have speakers and they must be turned on
- some tones may be annoying
- may disrupt other lab users

4. Using transparent, fading window for notification
[a] + no interference at all prior to notification
[b] + window may be recognized as MSN notification with experience
[c] + text in window can be used to provide crucial info **P2 (15/30)**
[d] + easy association between opening window for conversation, drawing on other forms of conversation initiation
[e] + opening application window is second nature to computer users
[f] + clicking on text for details is intuitive **P3 (11/30)**
[g] + attention is drawn to window where action should take place **P1 (24/30)**
[h] - fading window requires fast action **C2 (17/30)**
[i] - may miss to see window if focused on another part of the screen **C3 (16/30)**
[j] - non-persistence means user must notice at the time it comes up **C1 (24/30)**
[k] - nothing on window indicates it is an MSN window
+ does not disrupt other lab users

5. Lack of a dynamic interactive representation of lab status
[a] + decreases clutter that can interrupt people in the lab from their primary tasks **P1 (17/30)**
[b] + more desk and wall space can be dedicated to other (less dynamic but more information-rich) displays (like posters) **P3 (15/30)**
[c] + allows for a rich, easily comprehensable array of various representations to be used as different forms of information **P2 (16/30)**
[d] + discourages outsides from entering lab (good for when you want to get your work done)
[e] - provides no sense of direction for a visitor **C1 (16/30)**
[f] - can lead to a bland and boring lab area **C2 (15/30)**
[g] - lab may remain anonymous
[h] - visitors leave with no good understanding of personnel activity in the lab **C3 (13/30)**
[i] - lessens the number of demo-able, interesting, eye-catching, non-traditional displays
[j] - does not highlight what HCI is about
[k] - no history of status