The ContexTable: Building and Testing an Intelligent, Context-Aware Kitchen Table

Daniel Matthew Hoopes
Brigham Young University - Provo

Follow this and additional works at: https://scholarsarchive.byu.edu/etd

Part of the Construction Engineering and Management Commons, and the Manufacturing Commons

BYU ScholarsArchive Citation
https://scholarsarchive.byu.edu/etd/12

This Thesis is brought to you for free and open access by BYU ScholarsArchive. It has been accepted for inclusion in Theses and Dissertations by an authorized administrator of BYU ScholarsArchive. For more information, please contact scholarsarchive@byu.edu, ellen_amatangelo@byu.edu.
THE CONTEXTABLE: BUILDING AND TESTING AN INTELLIGENT, CONTEXT-AWARE KITCHEN TABLE

by

Daniel M. Hoopes

A thesis submitted to the faculty of
Brigham Young University
in partial fulfillment of the requirements for the degree of

Master of Science

School of Technology
Brigham Young University
April 2004
of a thesis submitted by

Daniel M. Hoopes

This thesis has been read by each member of the following graduate committee and by majority vote has been found to be satisfactory.

___________________________               ___________________________________
Date Richard G. Helps, Chair

___________________________               ___________________________________
Date Larry E. Wood

___________________________               ___________________________________
Date Barry M. Lunt
As chair of the candidate’s graduate committee, I have read the thesis of Daniel M. Hoopes in its final form and have found that (1) its format, citations, and bibliographical style are consistent and acceptable and fulfill university and department style requirements; (2) its illustrative materials including figures, tables, and charts are in place; and (3) the final manuscript is satisfactory to the graduate committee and is ready for submission to the university library.

___________________________               ___________________________________
Date Richard G. Helps
Chair, Graduate Committee

Accepted for the department

___________________________
Kevin L. Burr
Assistant Director, School of Technology

Accepted for the College

___________________________
Douglas M. Chabries
Dean, College of Engineering and Technology
ABSTRACT

THE CONTEXTABLE: BUILDING AND TESTING AN INTELLIGENT, CONTEXT-AWARE KITCHEN TABLE

Daniel M. Hoopes
School Of Technology
Master of Science

The purpose of this thesis was to design and evaluate The ContextTable, a context-aware system built into a kitchen table. After establishing the current status of the field of context-aware systems and the hurdles and problems being faced, a functioning prototype system was designed and built. The prototype makes it possible to explore established, untested theory and novel solutions to problems faced in the field.

A kitchen table was chosen as the application for these theories and solutions because kitchen tables are typically central to numerous activities in daily life in a home and this application has not been explored in the literature. Sensors and other electronics were embedded in the table surface, seat bottoms, and various items typically associated with a kitchen table such as a telephone, counter top, and cupboard. These sensors
allowed the system to gather information about users as they interacted with the
ContexTable.

One of the novel solutions to a current hurdle in the field was the learning
algorithm the system used to store, interpret, and utilize information gathered through the
system software and sensors. The pattern-matching learning algorithm and the language
in which it was programmed addressed current problems in context-aware systems
including: (1) the issue of modeling contexts, (2) the low quality of proactive system
decisions, and (3) the difficulty in creating a working design.

The algorithm addresses current hurdles by approaching them from a different
direction from prior systems. Instead of a rule-based system as used in the majority of
previous context-aware systems, the ContexTable uses a blank slate style and a pattern-
matching learning algorithm.

Twenty-five test users evaluated the prototype in both individual and group
sessions. Users were videotaped while completing a simulation of home/kitchen table
activities. Qualitative analysis of user reactions and information gathered through
questionnaires showed an overall positive reaction of test users and points to general
success of the prototype according to established criterion. Since the evaluation was
formative, the system was modified and improved after each user session.

Recommendations are made about further applications of the algorithm and about
how hardware could be improved to allow for long term and on-site studies.
ACKNOWLEDGMENTS

I would like to thank Brother Helps for his many suggestions, tips, expertise and general knowledge. Without his help, this project could not have come together and worked as well as it did. I would like to thank Dr. Barry Lunt and Dr. Larry Wood for their input and help at each stage of this research.

Without the support of my family, especially my parents, I could have never finished this thesis as quickly as I did. Thanks to all my friends who also supported me along the way.

Lastly, but most importantly, I want to thank my wife, Bri, for her unconditional support and love during this busy, uncertain time in our lives. You give true meaning to my life and without you all this would have no purpose.
# TABLE OF CONTENTS

LIST OF FIGURES AND TABLES

CHAPTER 1: INTRODUCTION

Background
Purpose of the Research
Research Questions
Justification
Methodology Of The Research
Definitions of Key Terms

CHAPTER 2: A REVIEW OF THE LITERATURE

Theoretical Background
Examples of Context-aware Systems
Bringing Context-awareness to the Kitchen Table
Current Intellectual Hurdles in Context-aware Systems
Design and Construction of a Context-Aware Kitchen Table
Conclusion

CHAPTER 3: METHODOLOGY OF RESEARCH

Methodology Overview
Design and Construction of a Context-Aware Kitchen Table
Implemented Solutions to Problems Described in Chapter 2
General Testing of Context-Aware Systems
Details of Methodology for This Study

CHAPTER 4: RESULTS AND ANALYSIS
LIST OF FIGURES AND TABLES

Figure 3.1. Force sensing resistor used in the seat bottoms. ............................................ 28

Figure 3.2. Arrangement and wiring of seated weight sensors. ........................................ 28

Figure 3.3. Overall photo of ContexTable showing some sensors and other items. ....... 30

Figure 3.4. Flow diagram of ContexTable pattern-matching algorithm. ......................... 32

Figure 3.5. The latest version of the interface the users encountered in test sessions.... 33

Figure 3.6. Flow diagram of simultaneous options during user experience............... 35

Figure 3.7. Sample page from version one of Test User Instruction set. ...................... 45

Figure 3.8. Sample page from version two of Test User Instruction set. ...................... 47

Figure 3.9. Sample page from third version of Test User Instruction set. ...................... 49

Table 4.1. Success Parameters For Evaluating A Context-Aware System............... 52

Figure 4.1. First page of all versions of instruction set. ............................................. 53

Figure 4.2. Second page of instruction set version one and two. ............................ 54

Figure 4.3. Second page of instruction set version three......................................... 55

Figure 4.4. Test User 1, Questionnaire #1.............................................................. 59

Table 4.2. Test User 2 Questionnaire................................................................. 62

Table 4.3. Test User 3 Questionnaire................................................................. 66

Table 4.4. Test User 3 Questionnaire #2........................................................... 66

Table 4.5. Test User 4 Questionnaire................................................................. 68

Table 4.6. Test User 5 Comments................................................................. 70

Figure 4.5. Instruction set version three; given to test user group three and four.... 75

Figure A.1. Sony VAIO touch screen enabled desktop system.............................. 99
Figure A.2. Touch Pen used to select items on screen...................................................... 99

Figure A.3. Two green NI screw terminal boards with white breadboard in middle. .... 100

Figure A.4. Voltage divider circuit & op-amp in voltage follower (buffer) mode...... 101

Figure A.5. Wiring Diagram for solenoid powered with 20V and software switching. . 102

Figure A.6. Desktop power supply and “cupboard door” opener........................................ 103

Figure A.7. Condenser microphone used to detect noise level.......................................... 103

Figure A.8. Remote “Wizard of Oz” toggle switches operated by moderator. .............. 104

Figure A.7. The first version of the interface................................................................. 105

Figure A.8. The second and final version of the interface.............................................. 105

Figure A.9. Presented to user after clicking “Today’s News.”......................................... 106

Figure A.10. Presented to user after clicking “Phone Directory.”................................. 107

Figure A.11. Presented to user after clicking “Personal Finances.”............................... 108

Figure A.12. Presented to user after clicking “Watch Cartoons.”................................. 109

Figure A.13. Presented to user after clicking “Roll Dice for Board Game.”............... 110

Figure A.14. Presented to user after clicking “Leave Reminder to Save Leftovers.”.... 111

Figure A.15. Opened for “Play Soft Music” or “Play More Entertaining Music.”........ 112

Figure A.16. Complete VI Hierarchy diagram. .......................................................... 114

Figure A.17. Entire VI hierarchy with system VIs not shown........................................ 116

Figure D.1. FSR data sheets. ......................................................................................... 243

Figure D.2. Reed relay data sheets. ................................................................................ 253

Figure D.3. Op amp data sheets..................................................................................... 255

Figure D.4. Photoresistor data sheet.............................................................................. 258
CHAPTER 1: INTRODUCTION

Background

Most computers today only perform actions when given explicit input by the user. This means that interaction between human and computer is symbolic to the degree that a user must click a button or type an entry in order for the computer to perform an action instantly or in the future. At the same time, computers are gaining the ability to perform more diverse tasks and are thus increasing in complexity. Because of limitations on human memory and the fact that many users do not know all functions available to them, there needs to be some kind of proactive decision-making by the computer to make the computer interface more effective and efficient. If this kind of capability existed in computers, they might be able to interact with us on a more collaborative level, instead of as slaves to explicit commands.

The notion of context-aware computing seeks to resolve the problem of needing explicit input and greater complexity by giving the computer system various means of knowing the context in which the system is being used at that time. Parameters of context include:

- Information about the current user and his preferences
- The way the computer has worked in the past
- The way a person is trying to engage the computer at present
- Location of use (in the case of mobile computing)
- The social situation (current collection of people)

In order for a computer to know this information, it must be designed with one or multiple sensing modalities and the software to interpret, compute, and implement
proactive presentations of services and information to the user, and store information about the context for future use. Sensors can provide contextual information such as user identity, number of users, location, user position (seated or not), changes in local variables such as objects near the computer system, and time of use.

Purpose of the Research

The kitchen table is the site of many diverse activities. People interact with their kitchen tables while doing any number of things such as: eating meals, reading the day’s news, watching TV, sitting to talk, paying bills, looking up phone numbers, giving first aid for cuts, doing children’s school projects, playing games, leaving notes for family members, listening to music, or planning family chores or vacations. With all the things done on the kitchen table, it remains a static, flat surface. It is proposed that a kitchen table could take more of an active part in helping people do the above activities.

The purpose of this research is to determine the effectiveness of a context-aware system in creating a collaborative (as opposed to slave/master) human-computer interaction style, using a kitchen table in a mock home environment for testing. Using a touch-sensitive screen and a proprietary interface, the context-aware kitchen table, or ContexTable, has sensors to enable it to know about numbers of people at the table, their identities, their current state (seated or not and some information about posture), the items on the table, and the time of use. The software is then able to interpret sensor information about current context cues and use past information to present the user(s) with items suitable for the current context. The interface will present information, tools, notifications, and entertainment appropriate for the current context. The users will be uniquely identified by seated weight with sensors in the seat bottom area.
Research Questions

Context-aware computing devices seek to present services and information to users appropriate for the current situation without raising the level of complexity (costs must be outweighed by benefits). This thesis hypothesizes that this system will illustrate and support this theory in a new application (a context-aware kitchen table) using a novel algorithm and computational strategy (blank slate learning of habits instead of context modeling).

Since this study will be a formative evaluation, the research will not be a comparison of the ContexTable to other similar systems. Rather, the prototype will be built and then tested qualitatively. During the course of this qualitative testing, the prototype will be adjusted and modified to correct bugs and problems, as they are uncovered. Various questions about suitable components and algorithms will be answered as the table is built. There are three questions related to this aspect of the research.

First, what is the most suitable algorithm for this project? The proposed solution is to build a learning program that begins as a blank slate (or “tabula rasa”) with no contextual cue rules. The system will then learn about the users during subsequent interactions with them and seek to provide information and services appropriate to the current context based on that information. A unique aspect of this strategy is that it allows for multiple descriptions of a single context. Current context-aware systems typically seek to pair one context with one contextual cue set model that will activate it.

For example, researchers concentrate on identifying the appropriate list of parameters and the respective values for each parameter that would most likely indicate
to the system that it should activate a certain service or interface. On the other hand, the ContexTable’s proposed strategy allows the system to generate multiple contextual cue sets that describe a single context, thereby allowing the system to overcome problems currently faced with existing strategies. Therefore, the researcher does not need to identify all the parameters and values before implementing the system. Rather, the system learns from user interaction. This strategy is discussed further in Chapter 3.

Second, what sensing devices and modalities are most suited and most economical for this prototype device? A set of sensors and actuators will be implemented based on common family/kitchen activities. The set of sensors will be just large and complex enough to demonstrate the effectiveness of the context-aware approach without overwhelming the project with sensor complexity issues.

Third, can the prototype system begin with no preset knowledge and learn to interact intelligently with the users (provide timely context help to the user(s)) without being an annoyance (as often occurs with incorrect guessing in many context-aware systems)? This is the main research question and is the subject of the bulk of the qualitative results discussed in Chapter 4.

Justification

The research topic is of importance because most work in context-awareness has been done with mobile computers where the location determines the context. However, in this example, the device is stationary and the different contexts emerge as the users engage the table in different ways. While some of these ideas have been explored before, the application of a kitchen table has never been attempted. In addition, this system will use a learning strategy that, while not academically significant on its own, has not been
applied to context-aware systems before. Therefore, this research explores the application of a new strategy in the field of context-awareness (pattern recognition learning with multiple cue sets) to a new type of system (an intelligent kitchen table) using only formative, not summative evaluation.

Another key aspect of this research is that it involves an actual, working system. Most work in the literature on context-awareness has been conducted using researcher-supplied data instead of actual values from sensors. This allows researchers to quickly evaluate different ideas and theories, yet represents a stumbling block when conclusions from these studies seek to be applied to actual situations. This research involves the use of actual hardware sensors and data gathered from real users and therefore will contain valuable information about how the theories explored affect test users.

Methodology Of The Research

The type of study will be a qualitative formative evaluation of using the ContexTable in a simulated home environment. The study will seek to evaluate the perceived effectiveness by test users of the ContexTable as a context-aware, interactive computer interface, or more simply, an intelligent appliance. To maintain good academic reporting standards, the first person has been removed from all reporting of results and procedures. However, note that in all places throughout this thesis, the terms “the moderator” (of test user sessions) and “the author” refer to the same person and are used when appropriate.

Delimitations

- The user test sessions will be conducted in a lab on BYU campus. Therefore, the users will have to imagine they are in the circumstances being simulated.
The results of this research, then, will be limited to evaluation of this prototype and will not extend to a comparison of this technology with others.

- Because it is a prototype, the design, usability and cost parameters will not be optimized. With respect to the hardware, proof of concept is the primary goal.
- To simplify the implementation of the theory, the issue of user privacy will not be directly addressed. However, this particular system does not disclose any sensitive information because the instrumentation does not inherently identify specific people by name.

Assumptions

- Test users will be able to appropriately evaluate the system even though all actions are a simulation and almost all items and services presented to the users are not genuine.
- Users can honestly evaluate the system and provide objective feedback suitable for this level of evaluation even with the system designer present.

Definitions of Key Terms

The following are definitions of terms relevant to this research:

1. Context-Aware Computing – current area of research and development in technology that deals with making computer systems easier to interact with by addition of software and hardware that gather information about a user, and interpret that information with the aim of presenting useful services and information in an appropriate user interface according to the current situation or context of the user.

2. Context Cues – different pieces of information gathered by a system that attempts to signal the system is being engaged in a particular situation.
3. Context Models – the notion that each individual context or situation that would define an instance when a context-aware system might provide information or services to the user must be defined by some set of parameters identifiable by the available hardware and software sensors.

4. Data Acquisition – the practice of gathering information from the environment with suitable electronic sensors and interfacing those electrical signal values with a computer that can then interpret and make use of them.

5. Formative Evaluation – research wherein a prototype is evaluated and modified successively with each round of feedback from test users. Different from summative evaluation in which something is compared with and evaluated against a competing solution.

6. LabVIEW – software created by National Instruments, Inc. that provides both a graphical programming environment and relatively simple interfacing with proprietary data acquisition hardware.

7. Qualitative Evaluation – exploratory and inductive in nature; differentiated from quantitative research that is typically confirmatory and deductive in nature.
CHAPTER 2: A REVIEW OF THE LITERATURE

With increasing memory and faster processors, computers gain the ability to perform an increasing number of tasks. At the same time, the user must deal with an increasing number of options in the user interface. Since human memory capabilities are limited and static, computers are becoming more difficult to use because users are presented with more options than they can deal with. Similarly, most computer systems await explicit input from the user before performing any action (Lieberman & Selker, 2000). There is typically no collaboration and computers come to be thought of as the user’s “slave.”

The notion of context-aware computing, or situated computing, seeks to solve the problem of overly-complex user interfaces while at the same time making the system more of a proactive collaborator in performing the actions the user desires. Simply stated, context-aware systems seek to offer sophisticated capabilities while maintaining a simple and intuitive user interface (ibid., p. 630).

This review of literature establishes the notion of context-aware computing as a possible solution to the overly complex user interface issue and presents various examples of context-aware systems that have been created. Next, the review identifies and substantiates the novel and important application of context-awareness to a kitchen table in a home setting. The review then elaborates various hurdles currently being faced in developing effective context-aware systems and also describes various unexplored solutions to overcome these hurdles. Lastly, literature is discussed that highlights key issues that would be involved in the building of a context-aware kitchen table.
Theoretical Background

In order to be helpful to the user, a context-aware system needs to “acquire and utilize information about the context of a device to provide services that are appropriate to the particular people, place, time, events and so forth” (Moran & Dourish, 2001, p. 89). Additional parameters for context might include user preferences and history of interaction (Lieberman & Selker, 2000).

These systems introduce a new interaction paradigm to the user interface. Most computer users currently interact with their systems through a GUI (Graphical User Interface) by means of mouse clicks and keyboard entries and in some cases menu-driven touch screens or verbal command. This traditional interface has been augmented with interfacing objects the user may grasp and manipulate to offer a TUI (Tangible User Interface) (Ishii, 2002). Both of these interfaces are in the foreground, or in other words: “the part…that is nearest to and in front of the viewer” (dictionary.com). Context-aware interfaces seek to move much of the interaction into the background, thereby becoming transparent to the user (Svanaes, 2001, p. 390).

Context-aware systems, then, need to be able to gather information about the current situation in order to present “appropriate services” to the user (Dey, 2003, p. 3). One problem with this idea is that computers have traditionally done the opposite: they are context-independent (Lieberman & Selker, 2000, p. 618). Context-independence simplifies the job of the programmer and makes systems easier to design. These simplifications transpose the burden of effort from the system designer to the end user. The end users have increasing demands placed on their attention and memory that often result in frustration and unused system abilities (ibid.).
Context-aware systems seek to fulfill the dream of Donald Norman where “we just engage in normal activities and the computation becomes invisible” (Norman, 1998, p. 10). The computation may become invisible, but the overall user experience is enhanced because users can “turn [their] attention away from computing, per se, and toward the other activities in which computing may play a role” (Moran & Dourish, 2001, p. 92). The interaction can reach this level of intuitiveness because, with context-awareness, the conversational bandwidth between the user and the system is increased from clicks and key presses to include natural actions and behaviors also (Dey, 2003, p. 2). Part of this increased bandwidth is created with the ability of the user to incorporate kinesthetic intelligence, or body movement and position, as a communication channel to the system (Svanaes, 2003, p. 398). With this higher level of conversation comes a decreased demand on the user for explicit inputs, thereby presenting more useful items to the user and lowering the probability of frustration (Petrelli, Not, Strapparava, Stock, & Zanacaro, 2000). This also frees the user to apply a higher portion of limited bandwidth to problem solving rather than using it up in interaction “overhead.”

Examples of Context-aware Systems

**Mobile Context-aware Systems**

The first context-aware systems primarily used physical location of the device to identify the context of the user. A good example is the Xerox ParcTab, an early PDA that is essentially a graphics “dumb” terminal connected wirelessly to a host that performs the computational duties (Schilit, Adams, & Want, 1994). Using a “cellular” approach, each room in the building was a “cell” and the system predicted context based on the room the user was in at the moment. The interface, for example, might present
instructions on using the coffee maker when in the break room and then provide note-taking space when in the meeting room.

A more recent mobile system is a context-aware PDA that acts as a personalized museum guide for each user (Petrelli, et al., 2000). Also employing a location-cue approach, the interface changes based on which exhibits are nearest to the device (and therefore user) and also customizes the interface based on past user interaction. Continuing with the PDA platform, researchers created displays that adjust brightness automatically and change screen orientation based on user positioning of the unit (Schmidt, Beigl, & Gellerson, 1999). Departing from handheld units, researchers at MIT have created a wearable context-aware system that includes a remembrance agent (DeVaul, Schwartz, & Pentland, n.d.). Although the device is wearable, the same theory is used wherein user location determines much of the basis for determining context.

**Stationary Context-aware Systems**

A second class of a context-aware system is stationary and changes the system interface based on cues besides physical location of the system (it is a constant). Illustrating the theory well is Weiser’s “Scoreboard” (television screen placed in a building lobby or break room, etc.) that shows information to users walking by that they are interested in and then changes for each user (Schilit, et al., 1995, p. 588). The E-Windshield, a car windshield that changes information presented both to those inside and outside the car is another example of a system that primarily uses cues other than location of the device to determine context. Inside the car, users will see information regarding possible obstructions, or points of interest (in this instance, functioning as a mobile context-aware system). From the outside, entire parking lots of cars, sensing their
surrounding autos, could be used to make large advertising space if each windshield presents a part of the ad (Selker, Burleson, & Arroyo, 2002).

Bringing Context-awareness to the Kitchen Table

With falling costs of sensors and computing devices in general, it is becoming easier to embed intelligence within anything around us (Cadiz, Shafer & Brumitt 2001). Indeed, entire buildings, rooms and even appliances are becoming “intelligent” (Moran & Dourish, 2001, p. 88). Research in context-aware systems has not, however, been applied to the kitchen table. The kitchen table is a very appropriate object with which to study context-awareness because it is considered a center of activity in today’s family home. The kitchen table is not only a place for meals, but also for many other things such as:

1. Discussion among family and friends
2. Board-games
3. TV-viewing
4. Paying bills
5. Doing homework and school projects
6. Performing first aid for child’s minor injuries
7. Providing a place to set groceries before putting them away
8. Looking up a phone number
9. Leaving reminders, etc.

Researchers have identified “both material and social circumstances” (Panayiotou, 2000, p. 2) which inherently involves the “collection of nearby people and objects” and the changes in them over time (Schmidt, et al. 1999, p. 894) as the main cues for context-awareness. Because a kitchen table experiences dramatic change in the
objects placed on it and in the people or person at it, the kitchen table becomes a fitting location to make use of the benefits of context-awareness technology. (see helps note for this paragraph)

Implementation of a context-aware kitchen table could be accomplished with the use of many wearable devices or small, portable systems carried by each member of the household, but placing the intelligence in the table itself is well supported (Shafer, et al., 2001). In addition, asking users to remember to carry sensing and identifying pieces adds to the burdens of the individual rather than reducing them, as is the aim of context-aware computing. This also agrees with Dourish’s goal of a “world around us imbued with computational power” (2000, p. 1).

With a stationary context-aware system the problem of decontextualization – difficulty in applying gathered information to future instances – can be disregarded because context information is more easily interpreted and used (Panayiotou, 2000). This delimitation may be made because the total possible number of contexts encountered is reduced since there will always be some kind of connection to kitchen tables. Although there are many other locations in a home where context-aware systems may be of use for the same reasons mentioned above, kitchen tables will be the application considered here because of the high number of interactions people have with tables in the course of a day, thereby increasing the ability to effectively evaluate such a system.

Current Intellectual Hurdles in Context-aware Systems

Modeling Contexts

Context-aware systems typically use sensors to gather information such as user(s) identity(ies), collection and changes in nearby objects, and other environmental
characteristics. “Logic sensors” gather information such as time, interaction histories, user preferences and software events (Schmidt, et al. 1999). All of this information needs to be used to identify the correct time and manner to present useful services to the user. Researchers who support the explicit naming of a context (eg. paying bills or watching TV) call this step “context modeling” (DeVaul, et al. n.d.). Similarly, Schmidt, et al. proposed that contexts should be identified with a unique name and that for each context, a set of features is relevant, and further, that each relevant feature has values determined by the context (1999). The problem of interpreting and defining context from gathered information is defined as a difficulty because of the costs to usability of getting it wrong and the ensuing annoyance to users (Erickson, 2002). In addition, in many circumstances, gathered contextual cues may be either too highly abstracted or have no abstraction, thereby making explicit modeling difficult (Schmidt, et al., 1999). Examples of highly abstracted cues include user attention level or emotional state; both of which are difficult to ascertain with current sensors. In addition, these cues are difficult to use for future context prediction because the range of possible values and configurations is large. A low abstraction cue includes factors such as light level or noise level and is simple to measure. However, since any given value of these cues occurs very often, they are applicable to so many contexts that their usefulness independent of other variables is diminished.

**Quality of Proactive System Decisions**

One reason for the difficulties described above is that consequences are dire for making a mistake. The annoyance when a context is incorrectly identified is high enough that users sometimes prefer not to have context sensing as opposed to keeping it for the
benefits it is designed to offer (Lieberman & Selker, 2000). Take the example of the Microsoft Office Assistant, commonly known as the “Paper Clip.” Many frustrated users just turn it off because it annoys more than it helps (Trott, 1998). One reason for the annoyance of some context-aware systems is that they are purely rule-based. If pre-defined conditions are met, the action is performed immediately. Many researchers have discussed the limitations and constraints of rule-based systems (Panayiotou, 2000; Petrelli, et al. 2000).

**Difficulty in Creating a Working Design**

Because context-awareness is a relatively new field, much current research focuses on the theoretical implications of the technology. Indeed, much of the software merely exists to substantiate a theory or illustrate a principle. What researchers have seen as a difficulty is taking these software systems and applying them in real-life applications and viewing their effectiveness. Many efforts have focused on creating toolkits (Dey, 2001), or infrastructures that simplify the task of designing context-aware systems by alleviating the need for end-user software designers to concern themselves with hardware I/O accessing and contextual data gathering and interpretation (Kim, Yae, & Ramakrishna, 2001). Indeed, if the software is kept “in the lab” and all sensor inputs are simulated, researchers “risk learning nothing about real user impacts and defeat the purpose of evaluation, wasting precious time and resources” (Edwards, Bellotti, Dey & Newman, 2003, p. 6). Simply put, “there is no point in faking components and data if you want to test for user experience benefits” (*ibid*, p. 7).
Hurdles Not Relevant to a Kitchen Table Application

There are many intellectual hurdles not applicable when context-awareness is applied to a kitchen table. Nevertheless, these hurdles are noteworthy because they help establish the delimitations and scope of this context-aware project. Since context-aware systems gather information about users’ locations and activities, much discourse has focused on privacy issues related to who can view what information gathered by a context-aware system (Ackerman, Darrell & Weitzner, 2001; Bellotti & Edwards, 2001). Users might not want a system administrator to have access to information about time spent in restrooms or which websites he visited, for example. However, as a delimitation of the current project, privacy issues will not be considered (see Chapter 1).

As stated, many context-aware systems reside in portable devices and are often embedded in wearable computers. In this case, the downfall of the system would be if the user either decides not to wear the device because it is uncomfortable or unsightly or it could merely be forgotten. In these cases, the benefits of context-awareness are lost because the user has not performed an explicit action (bringing the device). Although this hurdle does not apply to this treatment, it is important to restate that explicit actions required of the users should be minimized because they are “expensive” in terms the amount of attention and effort they require (Lieberman & Selker, 2000).

Possible Solutions to Intellectual Hurdles in Context-aware Systems

Modeling Contexts

Much of problem of modeling contexts comes from the “tradeoff between supporting extremely complex situations and providing a simple method for describing situations” (Dey, 2003, p. 8). Dey goes on to state that users are in the best position to
customize context-aware applications to meet their needs. This suggests that explicit context modeling may not be the best approach to this problem. Indeed, Beaudouin-Lafon and Mackay (2000) suggest that the “computer should respond flexibly, allowing users to react [to] and shape the current context, without having to explicitly define it in the computer’s terms” (p. 256). These same researchers go on to show that models work fine for officially sanctioned styles, but they fail when they do not support informal, user-created work patterns. Consider trying to make an outline different from the one the Microsoft Paper Clip wants you to create. The supposed context-aware helper has succeeded in making the possible impossible.

A possible solution to this is to not begin with rigid, pre-defined rules or models for context. Rather, begin with a “blank slate” and allow the system to learn preferences for actions to take based on user history and the probability that the current situation matches past iterations of the same sort. This will allow the system to fulfill Lieberman and Selker’s ideal of a system that “dynamically adapt[s] to context” (Lieberman & Selker, 2000, p. 20).

Interestingly, it seems that one potential weak solution - rule-based systems - led to the use of another weak solution with context modeling. If a system is based on rules, then when a condition set is met, there needs to be an explicit outcome—a pre-defined model for context appropriateness. If learning algorithms are used with “blank slate” architecture, then the constraint of using precisely modeled contexts may also be done away with. Lieberman and Selker elaborate one possible problem with this approach:

This involves an essential tradeoff: A conservative approach sticks closely to the concrete experience, and so achieves increased accuracy at the expense of
restricting applicability to only those situations that are very similar to the original. A liberal approach tries to do as much abstraction as possible, so that the result will be widely applicable, but at the increased risk of not being faithful to the user’s original intentions. (2000, p. 7)

With this paradigm the problem then shifts from one of pre-defined models and rules to a problem of what amount of resolution should the context information contain. In addition, the idea of blank-slate learning of contexts and actions open up the possibility of having more than one context cue set to identify a particular action to perform. To clarify, consider the possibility of a screen next to a kitchen table that can present a children’s broadcast or news in the morning based on a knowledge of whether those seated at the table are the children or an adult. In a rule-based system, it may be easy to stipulate that cartoons go with children and news with the adult. However, what will the system do if another adult is also seated with the first adult at times? Since there are two adults instead of one, this could cause the system to not identify the situation as “show news.” However, with blank-slate learning, if the selection to show news is made when only one adult is seated and when another is seated with him enough times, then both contextual cue sets can describe the “show news” context. In this manner, contexts are not too simplistically represented because, for example, “a second adult seated also” does not need to be removed from the cue set for “show news.” Bellotti and Edwards support this notion and caution against favoring computation over representation and state that systems “using simplistic representations of contextual information, while relying on overly intelligent machine interpretations of that information—are destined to fail” (2001, p. 210). It is somewhat akin to building a mountain where a molehill is the base.
Theoretical support for including a contextual cue such as “mother present” and not removing it from the cue set because it confounds explicit representation of the context model is found in cognitive science. Studies by Thomson and Tulving (cited by Reed, 2000, p. 166) in 1970 on the encoding specificity principle showed that the use of cue words when learning a word pair aided in retrieval of the target word even when the target word is not one that would be initially associated with the cue word. Thus, in the best model for intelligence—the human mind—cues for retrieval (like cues to indicate system contexts) may be more arbitrary than initially thought and aid in retrieval (defining context action to perform) rather than hinder it.

Thus when humans integrate from past experience (Erickson, 2002), context cues are not explicitly predefined and the items or contexts to which they apply are not explicitly named. More simply, context-awareness is like taking pictures; you gather the information present at the time the actions are recorded (Petrelli, et al. 2000). Just as with a picture, what falls within the viewfinder is not defined, it is merely what is present and available.

Quality of Proactive System Decisions

To illustrate the annoyance of incorrect proactive decision-making by a context-aware system, the Microsoft Paper Clip was discussed. Researchers have shown that a possible solution to the problem of incorrect identification of a user’s context (e.g. changing the format of a document contrary to the wishes of the user) is to involve the user in the control loop (Erickson, 2002). Beaudouin and Mackay suggest “computers should complement inadequate skills, such as short term memory, without trying to take over intellectual skills requiring human judgment” (2000). An example of this could be
if a system detects that context cues indicate that an action is appropriate, it could present the option to perform the action to the user with a dialog box asking for permission to continue and present the service. Going a step further, “more ambitious context-aware computing occurs when the system actively drives or trades off control with the user” (Selker & Burleson, 2000, p. 10).

In addition to involving the user in the control loop when a system desires to present a service to the user, the mere fact that a system begins with a blank-slate and learns based on user history should make the quality of the decisions better.

**Difficulty in Creating a Working Design**

In order to take a context-aware software solution and apply it to a real-life situation, care must be taken from the first stages of software design. A possible solution to this problem is to use software that already has interfacing built in for sensor and other hardware input. LabVIEW is an example of a programming language (uses constructs similar to C) that is designed to work with data acquisition hardware taking both analog and digital inputs (LabVIEW Resources, n.d.). Because the hardware interfacing is already designed, it is an ideal platform for working out theoretical postulates. Adding the “real-life” aspect is as easy as changing an on-screen “fake” input to one from the data acquisition board. More about using LabVIEW for programming as well as infrastructure will be discussed in the next section.

Another difficulty of bringing software from theory to reality is that of uniquely identifying objects that can be cues to a given situation. Consider needing to identify if a cereal bowl and child are present so the system can decide whether or not to play cartoons. Recognizing that there is a cereal bowl on the table is a difficult task to
perform for computers currently. Most researchers have solved this problem by just adding some kind of tag to all objects the system may need to identify as present or not. Some ways of tagging could be using infrared active tags (DeVaul, Dunn, Schwartz & Pentland, n.d.; DeVaul, Schwartz & Pentland), bar codes (Svanaes, 2001), or radio frequency identification (RFID) to identify things. Additional methods of simplifying the implementation of a theoretical context-aware system will be given later while describing how a context-aware kitchen table might be built.

Design and Construction of a Context-Aware Kitchen Table

Hardware Issues

Electronics sensors are the means by which data from the outside world is gathered and presented to the computer system. Sensor choice is important when enabling a kitchen table environment to determine the context in which it is being engaged. Dey, Abowd and Salber (2001) emphasize the sensing of physical attributes such as:

1. Time (of day, day of week, or both)

2. Place

3. People

4. Physical artifacts

5. Computational artifacts

First, time is a simple cue to gather because most systems running context-aware software also have an internal clock. In essence, time is a free cue (DeVaul, Schwartz ,& Pentland, n.d.). Although easy to acquire, time is also an important cue as it is a strong indicator of family activities at a kitchen table. Second, place is not considered because
once installed, the place of a kitchen table is constant. Additionally, even though a table may be installed in many different rooms within a given home or building, the placement is relatively static after initial placement.

Third, people and the collection of people must be identified. Because the existing system model (Selker & Burleson, 2000) of a table involves sitting in a chair, determining the identity of the seated person can be accomplished by measuring seated weight. Since primarily only the members of a given household would use the ContexTable, the resolution of a seated weight sensor can be relatively low by instrumentation standards and still correctly determine the identity of the person(s) sitting at the table. Fourth, physical artifacts can be identified as present by using a tagging model described above or another simpler method if appropriate. Fifth, computational artifacts present can be identified in the same way as physical artifacts. For example, the fact that the user has a PDA while using the system would be indicated by the covering of a sensor on the table. In addition to these parameters, the system should use “logical sensors” (Schmidt et al., 1999) that provide the software with information about current system states.

Selker and Burleson (2000) point out another aspect of hardware design. They state that design and aesthetics are important and can affect the ability to use the system. In addition, they point out the need for systems to have “seductive” interfaces. This means that the system should:

1. Create instinctive emotional responses
2. Connect with personal values and goals
3. Create ongoing surprises
4. Enable discovery of the unexpected or deeper value

In addition, the system should provide explicit congruency (have the ability to accomplish the explicit intended goal), and implicit comfort (meet needs broader than its specific task). The same researchers urge context-aware system designers to provide an augmented task model, or allowing the user to do more than they initially expected (ibid.). An example might be a kitchen table that senses that the father is not present and reminds the family to save leftovers for him automatically.

Software Issues

Traditionally, designers of context-aware systems have used object-oriented procedural code and scripting when writing software for context-aware systems (DeVaul, et al., n.d.; Dey, et al., 2001; Kim, et al., 2001; Brown, et al., 1997). On the other hand, other researchers made their software a kind of toolkit that allows designers of context-aware software to “plug in” to their context-aware outputs without having to deal with the more challenging areas of hardware interfacing and gathering useful context information (Dey, et al., 2001). This allows designers to think about their applications from a higher level and gives them architectural services or features around which they can work (Dey, 2003). Providing this context-aware application infrastructure has proved difficult because the limitations of the designer of the infrastructure are arbitrarily carried over to the application designer. In addition, it is difficult to attach actual sensors because the designer of the toolkit cannot know exactly which sensors the application designer will choose. The benefits of having a ready-made software infrastructure are overshadowed by the difficulty of interfacing sensors and other hardware (Edwards, et al., 2003).
As a framework for the goals of a context infrastructure, Edwards et al. give a minimal set of features a context-aware software infrastructure should provide:

1. Useful abstractions for representing context
2. A query and event-based information mechanism for acquiring context
3. Persistent store of context data for later use
4. Cross-platform and cross-language compatibility to allow use by many applications
5. A runtime layer that could support multiple applications simultaneously

LabVIEW is a programming language designed to work with proprietary data acquisition hardware that may help to overcome these problems. The application to this research will be discussed in Chapter 3: Methodology under the subheading “Difficulty in Creating a Working Design.”

Another suggestion by Edwards et al. (2001) when creating context-aware systems is to keep the test applications lightweight and not very sophisticated. These will allow an initial proof of concept and are more valuable on the first iteration of a system design because of the “return on investment,” they state. Pier (2000) also supports “keeping it simple” so over-zealous software designers do not make their software an annoyance rather than a help in the first iterations.

Conclusion

Although context-aware computing is a relatively new field, a great deal of groundwork has been laid to assist in future studies. The strong theoretical bases, although not all congruent, allows system designers to choose a defined paradigm and
design around it. The strong theoretical bases will ease system designers’ tasks, as there is at least a map for the road they wish to construct
CHAPTER 3: METHODOLOGY OF RESEARCH

Methodology Overview

The research methodology for the ContexTable is a mixture of various methods because the project involves essentially two parts. The goal was to create a prototype system to test a theory of functionality. First, the prototype must be built. Second, the prototype must be tested to find out if the objective was met or not. Although either of these tasks offers significant research depth and opportunity on its own, in order to create a usable system, both parts were completed. This chapter discusses the building and testing of the prototype ContexTable, a context-aware kitchen table. Code, circuit diagrams and a parts list of the hardware used to complete the prototype are detailed in the appendixes.

Design and Construction of a Context-Aware Kitchen Table

Hardware

As discussed in Chapter 2, a context-aware system should be able to sense and make use of data regarding the following parameters:

1. Time
2. Place
3. People
4. Physical artifacts
5. Computational artifacts

First, time and date are obtained directly from the software used to program the ContexTable. Second, as discussed earlier, place is not considered because once installed, the place of a kitchen table is constant.
Third, people and the collection of people are identified by seated weight. The sensors used to detect seated weight information are force-sensing resistors (FSRs). This is a low-cost (about $5/sensor compared to about $100/sensor for a load cell) method of
determining force or weight (depending on orientation) applied. In this particular application, four half-inch round FSRs (Figure 3.1, Interlink Part No. 402) were used in a square configuration (see Figure 3.2). The FSRs are sandwiched between two layers of ¼” Masonite wood with one at each corner. The wood layers are affixed so they remain together as one unit and are therefore suitable for placement below a seat pad in determining seated weight on a very low-resolution scale (only able to differentiate between absence of person, and presence of adult or child).

Two sensors were wired in series (one pair in front and one pair in back) to simplify the installation. Having a sensor pair in front and in back also gives the system rudimentary information on user posture. If there is more weight on the front sensors, the user could be said to be leaning forward and vice versa. This information is useful because it gives the system additional information about user posture instead of just the weight of the person as might be determined if the sensors were used in another arrangement or location.

Fourth, the system needed to identify physical artifacts. The task of identifying objects on a table without using any “tags” placed on objects is relatively intensive and usually involves some kind of digital camera using edge and color detection. Because this project involves a prototype and the focus is not on computer object recognition, this approach was not pursued. In addition, placing tags on all possible objects would force the user to perform actions outside the normal “user model” of a kitchen table (such as scanning a bar code over a scanner each time an object is placed on the table). Other methods such as RFID were cost-prohibitive, but would be suitable. Instead, a simpler method was chosen that still gave the desired effect.
Some of the photosensors used to identify objects on the table

False “cupboard” opened with attached solenoid.

Phone with sensor to detect if handset is lifted.

False “plates” and “board game” used for testing.

Remote control wired to send key press information to system.

Touch screen.

Figure 3.3. Overall photo of ContexTable showing some sensors and other items.

Photosensors were embedded in the table surface (see Figure 3.3). One sensor was placed at each location a user might set a plate or bowl and additional sensors placed at the center of the table, and in a location a user might set a checkbook. These sensors merely detect presence or absence of light. Thus, when the sensor is covered it will register a dark voltage level output. This could indicate that an object is covering the sensor. Since the sensors are placed in the center of specific locations where certain objects are usually placed (such as at the center of a placemat), this usually indicates the presence of the respective object (a plate or bowl). However, it could be that the user has merely placed his arm or a piece of paper over the sensor and the system would register an incorrect record in the database.

However, as will be seen in Chapter 4: Results and Analysis, this does not automatically nullify that entry in the database for two reasons. First, it could be that the user customarily places his arm in this position when performing this action and the
system should recognize it as a context cue. Second, it could be it was done unintentionally and this should not be used as a context cue. If this is the case, then it merely takes the system longer to build up enough surety (high enough score) for the context before prompting the user to perform the action. This is one of the strengths of this particular system; its ability to take in multiple context cue sets for any given action request. As opposed to most other systems, this system does not require one set of cues to represent one context-triggered action.

Fifth, computational artifacts present can be identified in the same way as physical artifacts. For example, the fact that the user has a PDA while using the system would be indicated by the covering of a sensor on the table. Since this is a simplified prototype, it is forgiven that it is not likely a user will place the PDA in the precise location to cover the photosensor in real life. In future research, other sensor technology, such as RFID, could increase the system capability without extreme monetary costs or burden on user actions.

In addition to the five parameters outlined above by Dey, et al. (2001), the system uses “logical sensors” (Schmidt et al., 1999) that provide the software with information about current system states. Since LabVIEW either opens an outside program or executes a subroutine (subVI, such as the software dice or simple text reminders) it is simple to program the system to record all important software events. The software keeps a record of which action the user has requested.
As described in Chapter 2, a current problem in context-aware systems is the difficulty in modeling contexts. In summary, researchers have found it difficult to create usable representations of what defines a context and similarly, what contextual parameters should be counted as cues to trigger a proactive system action. The solution explored in the ContexTable was to take on the problem from a different angle. Figure 3.4 summarizes the system’s algorithm as it relates to what goes on in the background.

**Figure 3.4.** Flow diagram of ContexTable pattern-matching algorithm.

**Implemented Solutions to Problems Described in Chapter 2**

**Modeling Contexts**

As described in Chapter 2, a current problem in context-aware systems is the difficulty in modeling contexts. In summary, researchers have found it difficult to create usable representations of what defines a context and similarly, what contextual parameters should be counted as cues to trigger a proactive system action. The solution explored in the ContexTable was to take on the problem from a different angle. Figure 3.4 summarizes the system’s algorithm as it relates to what goes on in the background.
Instead of finding a way to model and represent contextual cue sets with fidelity, the ContexTable uses a paradigm akin to that proposed to Petrelli, et al. (2000) in that it takes figurative “snapshots.” These snapshots are limited to the type, number and placement of sensors attached to the context-aware system. Snapshots are recorded as an entry in a database of all snapshots recorded any time a user selects a system action (ie. opening a program).

![Figure 3.5](image)

*Figure 3.5.* The latest version of the interface the users encountered in test sessions.

The last version of the menu for the test users is a simple display of the available programs and services the ContexTable can execute (see Figure 3.5). The time of day shows users what time of day it is because, in the simulation, they were imagining the time of day to allow the test to be completed in 20 minutes.

*Machine Learning*

As users go about their daily actions, the system successively records the information available to it by the sensors connected. If the user asks the system to
perform an action and the system determines the current snapshot to be close enough to a snapshot in the database (within a programmed range), then the database does not record a new entry, but rather updates the existing, matched entry with a higher “score,” indicating the same action has been selected by the user when all sensor values were similar enough to recorded values to warrant being called the same (in terms of the accuracy needed for context-awareness).

When the system encounters a current snapshot matching one in the database, it must then determine if the action has been performed enough times (by determining if the “score” is above a certain threshold) to warrant a proactive system suggestion to the user. If it is above the first threshold, then the user is prompted to accept or decline an action.

Figure 3.6 summarizes the flow of the user interface. This visual, on-screen prompt is accompanied by a simulated computer voice asking the same question. The voice prompt is to aid the user in case they are not seated directly in front of the monitor at the time of the suggestion.

If the user declines the suggestion, the score is reset to zero so that the user is not prompted again when the same conditions are met. This is because the user has just told the system that the database record that prompted the action is not one where the user actually wants the action to be performed. If the user accepts the suggestion, then the program is opened (or other action performed by system) and the score for the snapshot in the database is increased by one. In this manner, the system records that the suggestion it presented was a correct one. The system continues to monitor current sensor values, comparing them with sensor values (snapshots) in the database about every quarter second.
When a current snapshot matches one in the database and the user is prompted with a suggestion, there is another possible system action. If the user has verified the system prompt enough times to bring the score above the second threshold, then the user is no longer prompted to accept or deny the suggestion and the action is performed automatically. The user is notified of this action with the simulated voice signal.

Because the testing performed here involved a quick, 20 minute simulation, the thresholds were set significantly lower than they may be in an actual setting so the test users could have the desired experience in a short time. With a higher threshold in an actual scenario, the number of suggested actions would be lowered in the beginning because it would take more repeated situations to get the score high enough. This increases the time before the system begins to suggest actions to the user, but also ensures
that the only actions suggested to the user are those determined to be relatively common. This topic will be discussed further in Chapter 5: Conclusions and Recommendations.

This machine learning is very rudimentary and the Artificial Intelligence field offers numerous other, more complex and possibly more effective, learning algorithms (neural nets, Bayesian networks, fuzzy logic, etc). However, the elegance of a machine learning system is evident not from its complexity, but from its ability to perform a needed function effectively. The software created effectively performs all necessary duties for this particular application. While there were numerous constraints to the hardware and user interface, the algorithm showed it could be applied to more complicated implementations with minor revisions. This will also be discussed further in Chapter 5.

**Quality of Proactive Decisions**

In the previous chapter, the Microsoft Paper Clip was used as an example of poor proactive decision-making on the part of a supposed context-aware system. Erickson (2002) suggests involving the user in the control loop to solve the problem of a system performing incorrect actions automatically (i.e. changing formatting at inappropriate times). This means the system does not (at least initially) perform actions for the user automatically without verification from the user. As described above, the software for the ContexTable prompts to user to accept or reject suggestions for actions. The goal with respect to user interaction was to ask as little user intervention as possible while using the ContexTable. This type of system allows the user to go about normal activities (starting programs from a menu and changing objects/persons at/near the system) and the system merely asks yes or no questions at times. Then, after verifying an action enough
times (rejection need only happen once), the system begins to aid the user by performing actions with no intervention from the user. At this point, the action has been verified enough times that the system suggests only desired actions.

*Difficulty in Creating a Working Design*

As Edwards, et al. (2003) point out, if the software is written to always use “faked” data or is otherwise simulated, then researchers “risk learning nothing about real user impacts, and defeat the purpose of evaluation.” To get over this hurdle, a software language designed to work mainly with proprietary data acquisition hardware called LabVIEW G, was used. LabVIEW’s best use may be as a tool for rapid prototyping of context-aware systems and not for final, end-user versions, yet with upcoming versions of the software, it may be robust enough to handle end-user applications also.

First, LabVIEW code can be created on any computer with or without the proprietary data acquisition hardware and it is cross-platform functional between Linux, Windows, and Macintosh. This allows software designers to work out theories without having yet added hardware and but rather using numerical data entered by hand using convenient on-screen LabVIEW dials. The task of then adding hardware is taken care of by LabVIEW’s packaged code and hardware. Analog and digital sensors can be added and interfaced in minutes (LabVIEW Resources, n.d.). Changing the “faked” data to real sensor input data would then be complete. Indeed, this exact process was used for the ContexTable and worked flawlessly as a debugging tool while hardware was still being tested.

A second reason LabVIEW is a valuable tool for rapid prototyping is the graphical programming language it uses. This allows someone not as familiar with
procedural code to work up algorithms very similar in construction to those used in C, although the data flows use block diagrams and LabVIEW’s proprietary “wiring” system. Additionally, LabVIEW can run procedural code written in text such as Java, C and C++ within its own constructs. LabVIEW already has provisions to interface all ActiveX enabled programs. Since all Microsoft programs use ActiveX, this provides a powerful way for context-aware system designers to execute, customize and present programs and services to a user. These would be the “hooks” called for by Edwards et al. that allow “modification of existing applications (e.g. web browser, calendar, e-mail)” (2003). Programs not accessible with ActiveX calls could be interfaced as done traditionally in context-aware systems with a call to a function from within LabVIEW constructs using C, C++ or Java, as mentioned.

LabVIEW aided in creating a “lightweight” (Edwards, et al., 2001) program with the necessary constrictions and capabilities that allowed users to get a feel for the possibilities of such a system and allowed the author to effectively evaluate the system.

Limitations of LabVIEW Programming

With all the capabilities and benefits of LabVIEW as a development environment for context-aware systems, it has a significant downfall. The language, using ActiveX calls, could open, customize and present a program to the user, yet it was not able to determine when the user had closed the program after use. This is significant because of the manner in which the learning and proactive actions algorithm was designed. When a test user is using a service presented automatically by the system, the user will often close the program or service before changing the objects or people present at the table (ie. closing the morning news before standing and removing the bowl from the sensor). The
program compares current values to database records many times a second and since all cues are present to initiate a proactive action, the system will present the prompt to the user the moment the user has just closed that service or program. This will clearly not allow the program to function as desired.

The system needed some software cue that told the system when the user had closed the service or program outside the LabVIEW environment. This capability could not be identified and was therefore not implemented. Perhaps with greater LabVIEW programming experience and more time, the issue could have been resolved. However, a workaround was devised which allowed the users to have the desired experience. A simple, moderator-controlled hardware switch (interfaced with LabVIEW hardware and software) was installed which allowed human intervention to overcome the software limitations. Therefore, while the user was testing the system, the moderator merely flipped the switch when the user exited the program. This allowed the system to have the needed information about the closure of the outside service or program the system had activated. The switch was flipped inconspicuously so the user was unaware and could evaluate the system for the merits it could have if this limitation were overcome.

General Testing of Context-Aware Systems

*Examples From the Literature*

The second major part of the formative evaluation is to test the prototype. Other context-aware systems have been created and have been tested in various manners. It is helpful to begin by delineating the stage of use this evaluation seeks to cover. Selker and Burleson (2000) provide a generalized framework in which to view the progression of a
user when first encountering a context-aware system. They state that it is normal for a user to pass through the three stages of:

1. Learning
2. Frustration
3. Mature use

So, it is natural if users are not fully satisfied with the device the first time they use it. This evaluation will seek to cover the “Learning” stage and perhaps the “Frustration” stage. The definition of “Mature Use” will be adjusted slightly for this research because the system is a prototype and users will have limited time (minutes not days) to get used to it.

As an assessment of success of a context-aware system, Schmidt et al. (1999) lay down specific objectives. Success is then measured by the degree to which the system meets those objectives. Another context-aware system, HP’s CoolTown (Svanaes, 2001) is evaluated by degree to which it departs from the current user interfaces in drawing on more of Gardner’s (1983) seven intelligences. The intelligences currently drawn on by user interfaces are primarily logical-mathematical, linguistic, and visual intelligence. An additional intelligence used by CoolTown, for example, is bodily-kinesthetic because movement and body positioning affect computer actions.

Various systems are evaluated by presenting the abilities of the prototype with proof-of-concept applications. Systems evaluated in this way include the Xerox PARCTab (Brown, et al. 1997) and the E-Windshield (Selker, et al., 2002). This type of testing is inherently devoid of rigorous scientific method, yet is suitable for projects of small scale where proof-of-concept is preferred over widely applicable academic impact.
In addition to these methods, some researchers provide general success parameters for a context-aware system. First, the system can be deemed not a success if it results in more distractions than helpful actions. Second, the system should provide information and services relevant to the task the user wishes to accomplish (Dey, 2003). Third, the user should be able to concentrate on the task and not the computer. In essence, the computer becomes as “invisible” as possible (Petrelli, et al., 2000). Fourth, the system succeeds if the “technology becomes incorporated naturally, even subliminally, into the user’s work practice” (Pier, 2000, p. 2). Last, another test is to remove the technology after the users have reached the “mature user” stage and see what the reactions of the users are (ibid).

Formative Versus Summative Evaluation

The first steps in the creation of any new system or even any piece of software are prototyping and evaluation. The prototyping of this system has been described above. Next, the prototype must be evaluated. In evaluation, formative evaluation and summative evaluation must be differentiated.

Most people are familiar with summative evaluation, which seeks to determine if the system that has been built is actually the system that was envisioned and specified. In summative evaluation, it is important to know whether the usability specifications and goals have been met or exceeded (Rosson, Carroll, 2002).

Formative evaluation is aimed at improving a design prototype, not merely measuring overall quality. In formative evaluation, the system is not under scrutiny as a whole, but rather each part is looked at during test iterations to determine changes that need to be made. These changes are made before more test users perform an evaluation.
In this manner, formative evaluation seeks to improve a system repetitively until it is deemed ready to begin summative evaluation. (Ibid.)

Pancer (1996) elucidates the differences between these types of evaluation when referring to assessment of social change programs:

Formative evaluations are designed primarily to provide information that can be used in helping to form, develop, or improve the program. Summative evaluations, on the other hand, are intended to provide information that can be used in “summing up” the program’s worth or value. Summative evaluations typically occur at a later stage in the program’s development, and they are designed to provide a kind of report card for the program, indicating the extent to which the program has achieved its major objectives.

It should be noted that Pancer does not mention the specific use of quantitative studies versus qualitative ones when referring to formative and summative evaluations. Presumably, both types of studies are valid and useful at either stage of evaluation and depend on the nature of the subject in question.

Research with the ContexTable focuses mainly on the formative evaluation stage. Since this is the first time a system of this type has ever been built, many iterations of the “test-improve-retest” cycle had to be performed before the system could be tested by a larger number of subjects. However, after the major problems were identified and corrected, users began to experience the system in the manner that fulfilled all the design and user experience specifications laid out. At this point, the research began to touch on summative evaluation.
It must be made clear that the main thrust of this study is not summative evaluation. If this were the case, then much more quantitative work would have been required so the results could be extrapolated to have meaning on a greater scope. Still, since the prototype reached a point where no major additional changes needed to be made and the user could compare the experience to other similar situations and draw meaningful conclusions, the study could be said to have begun into the realm of summative evaluation.

**Qualitative Versus Quantitative Study**

As noted above, this study does not seek to provide quantitative results that would aid in induction of a broader meaning by the strength of statistical significance. On the contrary, the study is qualitative. When deciding which type of study to utilize, the purpose of the study is the key. The following describes this difference well. Take note that although it speaks of social “programs,” the reasoning is applicable to any time the qualitative vs. quantitative study question arises.

If the purpose is to gain a holistic, in-depth understanding of the program’s impact from the perspective of those involved in the program, a qualitative approach would be the method of choice. If, on the other hand, the purpose of the evaluation is to determine the extent to which all program participants have achieved the goals established by the program, a quantitative approach would be preferable. (Pancer, 1996)

In addition, qualitative studies are useful for the same reason that quantitative studies are sometimes inadequate. Specifically, user experiences with regard to a new type of human computer interaction scenario are complex things that are unique to the social,
historical, and temporal context in which they are embedded. Further, each person involved in the program will have an experience that is to some extent unique and cannot be adequately described by looking at numerical scores on a few standardized measures (Ibid).

Pancer (1996) goes on to describe the basic parts of a qualitative study. These parts usually include “(a) in-depth, open-ended interviews; (b) direct observation; and (c) review of program documents and records.” Although these parts were not adhered to in the strictest sense, the parts applied to this study are described below.

Details of Methodology for This Study

Since this is a qualitative and a formative study, the atmosphere is different from a typical quantitative study. This study is even different from a typical qualitative study that is not formative. For example, some qualitative studies involve one-way mirrors and a certain measure of “sterility.” However, because the prototype is in such an early stage of development and initially contained many bugs, the study was less rigorous in terms of protocol that ensures statistical significance. Therefore, users will be selected from among acquaintances and volunteers entering the lab after seeing signs advertising the testing. The technical background of each user is listed in Appendix E. Since the ContexTable represents a departure from all former computer interaction for all users testing the system, such a “sterile” study would not be possible because the user would reach too many points of question or frustration to be able to have the desired experience. The reason for this is that, to use Selker and Burleson’s (2000) nomenclature, the user cannot employ an existing system model to complete tasks. Since a new channel of
communication has been opened up between the system and the user, the style of use is completely different from traditional computer systems.

**Data Gathering**

The standardized materials given to the test users were (1) a printout with a one-page introduction to how the system works and how to interact with it and (2) a questionnaire. The main method of gathering data from the test users was by means of a questionnaire filled out by test users after going through the simulation of using the ContexTable system. The second method of gathering feedback from test users is from conversation with the test users. Lastly, all test user sessions and subsequent discussions were video taped. These tools are described in the following sections.

![Figure 3.7. Sample page from version one of Test User Instruction set.](image-url)
Instruction Set

The test users were given a standardized printout of the instructions for using the system during the simulation. All users read this printout before being allowed to ask the moderator any questions. The users were assured they were not under scrutiny, but rather the system was. Test users were also advised they could ask the moderator questions along the way. The questionnaire itself was one of the subjects of formative evaluation. The questionnaire changed along with the capabilities and level of functionality of the system. There were two main changes made to the questionnaire and accompanying instruction set. At first, the instructions (see Figure 3.7) focused on guarding the users from engaging in activities that would cause the system to malfunction. In addition, after the problems creating the need for the specific instructions had been fixed, the instructions were adjusted slightly, but still not a great deal (see Figure 3.8).

Subsequently, the responses to the questionnaire and comments made pointed to the fact that the test users did not understand the system capabilities correctly. Test users believed that they had to conform all actions to a pre-defined script in order for the system to react correctly. However, the key strengths of this particular context-aware system are precisely that the user does not need to conform to a predefined script for the system to react correctly. As described in Chapter 2, this system does not work on the paradigm of context modeling where the focus is on defining cues to activate a given context action. Instead, users merely perform actions and selections as they normally would and the system reacts to repetition. In the short time allowed for users to help evaluate the system, it was helpful for users to follow a predefined set of actions and
selections, because it allowed them to see the capabilities of the system in the minimal amount of time.

<table>
<thead>
<tr>
<th>Check</th>
<th>I would like to:</th>
<th>Instructions: (Follow exactly)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Read today’s news while eating breakfast. See instructions →</td>
<td>1. If not done, place a check mark next to this item. You will use it each time you are at “Morning.”</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2. If not already doing so, sit down at a chair.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3. Select “Morning.”</td>
</tr>
<tr>
<td></td>
<td></td>
<td>4. Place a bowl or plate on one of the placemats.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>5. Using the special pen, touch “Today’s News.”</td>
</tr>
<tr>
<td></td>
<td></td>
<td>6. Close the window when done.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>7. Remove bowl or plate from placemat.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>8. Go to next step: “Noon.”</td>
</tr>
<tr>
<td></td>
<td>Watch cartoons while I eat breakfast.  See instructions →</td>
<td>1. If not done, place a check mark next to this item. You will use it each time you are at “Morning.”</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2. If not already doing so, sit down at a chair.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3. Select “Morning.”</td>
</tr>
<tr>
<td></td>
<td></td>
<td>4. Place a bowl or plate on one of the placemats.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>5. Using the special pen, touch “Watch Cartoons.”</td>
</tr>
<tr>
<td></td>
<td></td>
<td>6. Close the window when done.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>7. Remove bowl or plate from placemat.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>8. Go to next step: “Noon.”</td>
</tr>
<tr>
<td></td>
<td>Call a store I’m going to later today.  See instructions →</td>
<td>1. If not done, place a check mark next to this item. You will use it each time you are at “Morning.”</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2. If not already doing so, sit down at the chair by the phone.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3. Select “Morning.”</td>
</tr>
<tr>
<td></td>
<td></td>
<td>4. Lift up the phone.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>5. Using the special pen, touch “Phone Directory.”</td>
</tr>
<tr>
<td></td>
<td></td>
<td>6. Use online phone directory as needed.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>7. Close window when done.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>8. Hang up phone.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>9. Go to next step: “Noon.”</td>
</tr>
</tbody>
</table>

Figure 3.8. Sample page from version two of Test User Instruction set.
However, at the same time, the user was not seeing the main strength of the system; the freedom in actions and selections allowed. Therefore, the instruction set was changed (see Figure 3.9) to enable the user to experience the freedom of actions and selections of which the system is capable. In this instruction set, the user was given no explicit instructions on actions or selections to perform. User reactions to the different instruction sets will be described fully in Chapter 4: Results and Analysis.

*Questionnaire*

The purpose of the questionnaire was to evaluate the user experience during the system use simulation. For the first users, the questionnaire (see Figure 4.4.) focused on whether or not the user had the desired experience while using the system. However, when the main problems had been corrected for later users, version three of the questionnaire (see Figure 4.6) focused more on the quality of the user experience and how the user would rate the experience against normal daily activities. In this manner, the study began to touch on summative evaluation because the system was not under scrutiny to determine what needed to be changed for future users, but rather to determine the nature of the user experience as compared to the traditional user experience.

Some things on the questionnaire remained consistent. For example, all of the questionnaires had questions that asked the test users to think of possibilities for application in other areas of life. The reason for this was that the author posits that if users generally enjoyed the system and believed it to have merit, then they would likely begin to see the possibilities for application in other realms. However, if the user had a negative experience or believed the system to have less utility than costs, the author supposed users would likely give few or no suggestions desired use in other daily items.
An example of another question that did not change was on whether the system provided any surprises that kept the interest of the user. This is in accord with the suggestion from Selker and Burleson (2000) that a successful system should create ongoing surprises to generate interest and urge the user to use the system more. Other questions remained consistent throughout testing, and are apparent in the next chapter.

![Figure 3.9. Sample page from third version of Test User Instruction set.](image)

The exact changes made to the questions and which ones were deleted and added will not be explained here. In the next chapter on Results and Analysis, the changes will
be explained as the answers to those questions are discussed along with the reasons for these changes.

*Video Taping of Test User Sessions*

The second method of gathering data was from the video taping of user test sessions. Because many comments and user observations were made during the course of the session and were not recorded on paper in the questionnaire, the videotapes provide some of the most telling data. In the video, users are not preparing responses as they might be when answering a questionnaire, so in some respects the reactions and comments on the videotapes are more truthful than questionnaire responses.

In addition, users were engaged in casual conversation after the session before answering the questionnaire (and after) about their experience while using the ContexTable. These responses figure prominently in Chapter 4: Results and Analysis. The video tape records were reviewed and notes were made of the contents. These notes are found in Appendix C.
CHAPTER 4: RESULTS AND ANALYSIS

Data Presentation

Order of Presentation of Test User Results

Test users performed a simulation of actual use of the ContexTable in a home setting. Since this study was a formative evaluation of a prototype, the data will not be presented as a cumulative summary of all users’ reactions and responses to questionnaire items. This is because both the user experience and the questionnaire changed during the course of the test period. Therefore, the results will be presented in the order in which users tested the system, on a mostly individual basis.

A reasonable progression of user experiences can be based around Selker and Burleson’s (2000) user experience levels of (1) Learning, (2) Frustration and (3) Mature Use. The 25 test user results were divided roughly by how they fit into these phases. Group One experienced the learning and frustration phases for a large portion of the time. Group Two had varying amounts of learning and frustration and some even experienced or were able to envision mature use. Group Three users were almost all able to forgo almost all frustration and pass from a shorter learning phase than the prior two groups to something that might be called mature use for this simplified prototype. Users from Group Four had experiences similar to those of Group Three, yet the sessions were conducted with groups of users instead of individual test users.

Review of Success Parameters

Before continuing, some success parameters for a context-aware system from the literature review in Chapter 2 will be reviewed (see Table 4.1).
Table 4.1. Success Parameters For Evaluating A Context-Aware System.

<table>
<thead>
<tr>
<th>Success parameter</th>
<th>How to test for fulfillment</th>
<th>Applicable here?</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Degree to which the system draws on other of Gardner’s seven intelligences than the typical computer system. (Svanaes, 2001)</td>
<td>Whether intelligences other than logical-mathematical (such as body-kinesthetic) are used.</td>
<td>Not easily quantitatively measured, but will be discussed.</td>
</tr>
<tr>
<td>3. Information and services relevant to the task at hand. (Dey, 2003)</td>
<td>Ask users in questionnaire.</td>
<td>Applicable, tested and presented.</td>
</tr>
<tr>
<td>5. Ability to incorporate technology naturally into normal “work” practice. (Pier, 2000, p. 2)</td>
<td>Ask users in questionnaire.</td>
<td>Applicable, tested and presented.</td>
</tr>
<tr>
<td>6. Reactions of user after removing technology after having reached mature user phase. (Pier, 2000, p. 2)</td>
<td>Not tested because users not able to reach “mature user” stage in short time allotted.</td>
<td>Applicable, but not tested.</td>
</tr>
</tbody>
</table>

Style of Presentation

In general, as Pancer (1996) states, “the major purpose of a qualitative evaluation report is to give the reader a sense of what it was like to be involved in the program, and how the program affected those who participated.” More specifically, the same work goes on to remind “a qualitative evaluation would likely include many quotes and descriptions.” Some user experiences will be described in more detail than others. The earlier users typically had more problems and for the most part, more comments. Later users, proceeding through the experience more smoothly, generally had fewer comments. This could be because they did not think about the system providing the experience and what it might be able to do with more work, but rather focused on the specific experience they had. In addition, the later users’ questionnaires were provided more quantitative measures and can therefore be compiled to a concise summary of results of this group.
Therefore, early users are described individually and later users will be described as a summary of their experiences.

<table>
<thead>
<tr>
<th>Evaluating the ContextTable</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1 of 4</strong></td>
</tr>
<tr>
<td><strong>General Instructions:</strong></td>
</tr>
<tr>
<td>1. The purpose is to see how users react to a “smart” kitchen table.</td>
</tr>
<tr>
<td>2. You will be asked to:</td>
</tr>
<tr>
<td>a. Go through an imaginary scenario as you use this kitchen table (videotaped).</td>
</tr>
<tr>
<td>b. Fill out a short survey.</td>
</tr>
<tr>
<td><strong>What is this thing?</strong></td>
</tr>
<tr>
<td>1. The ContextTable, a “smart” kitchen table has a number of sensors attached to a computer.</td>
</tr>
<tr>
<td>2. A main menu has selections for things to do.</td>
</tr>
<tr>
<td>3. The table uses its sensors to learn your preferences when you repeat the same actions over time.</td>
</tr>
<tr>
<td>4. Because this is a prototype, many things are not perfect. For example:</td>
</tr>
<tr>
<td>a. You have to put things in the right spot for the sensor to be able to know it is on the table.</td>
</tr>
<tr>
<td>b. Other items you will use are fake. Try to imagine they are real.</td>
</tr>
<tr>
<td><strong>How does it work?</strong></td>
</tr>
<tr>
<td>1. First, the system learns from your habits.</td>
</tr>
<tr>
<td>a. After a few repetitions, the system will start to suggest things to you when it thinks you might need them.</td>
</tr>
<tr>
<td>b. You can either accept the suggestion, or if it is wrong, reject it so it will not be suggested again.</td>
</tr>
<tr>
<td>c. After you do the same thing enough times, it will stop asking you and just do whatever it is the program thinks you need.</td>
</tr>
<tr>
<td>2. Second, there are some things that will trigger programs to pop up without you having to “teach” it your preferences. You’ll see those in a minute.</td>
</tr>
</tbody>
</table>

Any questions? (At any point, just ask if you are unsure about anything.)

It is helpful to my record keeping to read the directions out loud while doing the items. Also, please think out loud while trying to do things.

Remember, this is not a test of your abilities; it is a test of my system (which is not perfect).

*Figure 4.1. First page of all versions of instruction set.*
Test User Experience

All test users, upon entering the test area were given a standard set of instructions. The first page (see Figure 4.1.) contained a brief introduction to the system and how it works. After describing what would be happening in the next 20 minutes, the user was reassured both on this page and verbally that the system was what was being tested, not them. Great effort was taken to assure users they were not being evaluated for their abilities, but rather the system was.

The second page (see Figure 4.2.) was a large print reminder that the system is slow and to be patient (so the system had time to perform proactive actions before users selected them) and to follow the instructions exactly. However, as the test session instruction set changed and the user sessions were more free form, the instruction to follow directions exactly was removed. It was then replaced with a reminder to the users to be honest and that their answers would not hurt the feelings of the moderator no matter what they responded. (see Figure 4.3.)

<table>
<thead>
<tr>
<th>Important:</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Follow Instructions EXACTLY</td>
</tr>
<tr>
<td>2. Allow time for programs to open, computer is VERY SLOW.</td>
</tr>
</tbody>
</table>

*Figure 4.2. Second page of instruction set version one and two.*
After reading these general instructions, the test users were asked if they had any more questions or comments before beginning the session. The test users were told that this is not like a typical test where the moderator had to be behind a one-way mirror, etc. Instead, they could ask questions as they proceeded through the simulation. Establishing this free and easy atmosphere helped users to feel more comfortable about the situation and to feel more at ease while performing the test.

The user then sat at the ContexTable and began the test by reading the instructions on the table for the simulation. The changes made to the instructions are detailed below. During the simulation, the users regularly conversed with the moderator and asked questions as necessary. The moderator provided answers that were mostly minimalist in hopes of letting users discover more about the system on their own.
After the simulation, the users were asked if they had any questions or comments before proceeding to the questionnaire. The reason for this was to clear up any misconceptions the user might have about the capabilities of the system. Therefore, when the user filled out the questionnaire, they could get the best feel for what the system can do if they were not limited to the quick 20 minute time limit imposed here for convenience to the test subjects.

All users filled out a questionnaire, yet the questionnaire changed as more subjects tested the system. Therefore, the questions and, in turn, the responses changed over the course of the tests.

Group One: Learning to Frustration Phases, Instruction Set Version One

One problem discovered among the first users was in the paper instruction set given to users. Versions one and two of the instruction set were difficult to understand for most users (see Chapter 3, Figure 3.7. & 3.8.). Users were asked to place a checkmark next to item they chose to make sure they repeated the same action each time the simulated “day” was repeated. Therefore, many of these first test users were not able to have the desired experience that would allow them to evaluate the system most accurately because of the added complications of the instructions, the change in time of day and the need for placement of items in a certain location. In a real situation, all of these procedures would not be needed. Ideally, the user would simply go about normal activities and the system would begin to ask yes or no questions. The fact that the instructions were so explicit distracted the users from the desired experience.
Test User 1

Summary

Overall, for Test User 1, the system made the correct suggestions. However, the code written to properly delay the opening of services repeatedly was not working. The limitations of the LabVIEW programming language described earlier make it impossible to know when the user closed a service opened by the ContexTable. Because of this code the system began to repeatedly open services over and over after the user closed the service. With this problem occurring often, the moderator needed to intervene numerous times during the test to correct invalid suggestions and suitably change parameters manually to fix the problem. However, while experiencing these problems, the user made an interesting observation that exhibited his understanding of the functioning of the system when he said: “So, if I go to ‘morning,’ it will stop?” This showed the user was learning that interacting with this system involved more than mere symbolic interaction with the system by explicit (version one) instructions. Instead, he saw that he could change a cue, in this case “time,” and thereby adjust the actions of the system. Since this was learned after just a few minutes interaction with the system, it points toward the system being intuitive and somewhat natural to use, even with its bugs and problems.

One thing the moderator noted was that the system lacked and would benefit from audible notifications to accompany on-screen notifications. This would draw the user’s attention to the system when it performed “proactive” suggestions and displayed messages for users. Another problem noted by the moderator was the small size of notification messages to the user on the screen. The users had a hard time seeing and
reading the text and it was also difficult to navigate the services with the small size. This suggested the text and the “close window” button should be made larger.

*Test User 1 Questionnaire*

The questionnaire (see Figure 4.4.) shown here (and for all subsequent users) is not an exact copy of theirs, but a summary to show their answers. Some of the answers are of note. The answer to #4 illustrates Selker and Burleson’s “Frustration Phase.” Answer #5 showed that the user did not completely understand the full capabilities of the system. The moderator explained that he could have done things in any order and at any time he wished and this surprised him. This was the main reason the instruction style was changed later on to allow users more freedom in what actions they performed and at what times.
Test User 1 Questionnaire # 1

1. Did the ContexTable do anything helpful for you?
   Yes, TV channel, and the loud noise, I thought that was helpful.

2. What surprised you the most about using the ContexTable?
   After the 3rd round having it recognize what I wanted it to do, by just having to change
   the time of day. It really fits what I want to do because it seems I always do the same
   kinds of things at the same time of day.

3. Are there any other actions you wish the table could do that it can’t currently do?
   Make food for me! (facetiously) Maybe have more positions than the limited number
   of sensors. So that you can place objects anywhere on the table.

4. Were there any times you thought the ContexTable was annoying?
   When it kept repeating the same thing that I had already done.

5. Was the “No you guessed wrong” button enough of a help in dealing with these issues?
   Yeah it was fine, but I wish there was the ability to have more complex scheduling, like
   doing one thing one day and another thing another day.

6. What other things were not pleasing to you?
   “Sitting down. The seat sensor is kinda riding me there.” I’m assuming that when this is
   implemented, you’re not gonna have wires coming off the remote.

7. Anything else you would like to add?
   “It’s definitely a prototype, but I think it’s a great idea.”

---

Figure 4.4. Test User 1, Questionnaire #1.
Test User 1 Questionnaire #2

1. Would you want this kind of technology built into other daily things? How about your cell phone (knows to vibrate if in a class or ring if outside), ear (warms up for you in morning depending on when you wake up), or bed (sets the alarm automatically based on when you go to sleep), etc.? If so, how would it be helpful?
   *Yes, it would be helpful for cell phone especially.*

2. Did you find this system to be helpful or an annoyance? Please circle a number below.

<table>
<thead>
<tr>
<th>Extremely Annoying</th>
<th>Not a Help or an Annoyance</th>
<th>Extremely Helpful</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td>6</td>
</tr>
<tr>
<td>7</td>
<td>8</td>
<td>9</td>
</tr>
<tr>
<td>10</td>
<td>10</td>
<td></td>
</tr>
</tbody>
</table>

3. Did the system provide you with services/prompts that were helpful to the task at hand?

<table>
<thead>
<tr>
<th>The services/prompts provided were completely useless</th>
<th>The services/prompts would now make a difference to me</th>
<th>The services/prompts were extremely helpful to tasks I was trying to accomplish</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td>6</td>
</tr>
<tr>
<td>7</td>
<td>8</td>
<td>9</td>
</tr>
<tr>
<td>10</td>
<td>10</td>
<td></td>
</tr>
</tbody>
</table>

*Please explain: Things were pretty self explanatory.*

4. If the system automatically changed the time and all you had to do is go about your normal activities and occasionally answer yes/no questions, would this system free you from computer use or bog you down with more?

<table>
<thead>
<tr>
<th>I would have to worry MORE about using the computer, I think</th>
<th>It would NOT make a difference in how I use computers</th>
<th>I would be able to pay LESS attention to using the computer and concentrate more on meaningful tasks</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td>6</td>
</tr>
<tr>
<td>7</td>
<td>8</td>
<td>9</td>
</tr>
<tr>
<td>10</td>
<td>10</td>
<td></td>
</tr>
</tbody>
</table>

5. Do you think you could learn to use a system like this and incorporate it naturally, maybe even subliminally into your normal work practice?

<table>
<thead>
<tr>
<th>It is too complicated to be able to work it into my normal activities</th>
<th>It would NOT make a difference in how I use computers</th>
<th>This kind of communication/collaboration with other computer would be natural, almost subliminal</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td>6</td>
</tr>
<tr>
<td>7</td>
<td>8</td>
<td>9</td>
</tr>
<tr>
<td>10</td>
<td>10</td>
<td></td>
</tr>
</tbody>
</table>

6. While you were using the system, were there any surprises or things that kept your interest and made you want to continue using it? If so, what?

*The morning breakfast routine was nice. I have a routine, and this system would be perfect for it.*

7. Can you think of any other daily items that might benefit from this kind of technology?

*Showering, watching TV.*

8. Do you follow a schedule enough to make this kind of thing useful?

<table>
<thead>
<tr>
<th>No, I am completely erratic</th>
<th>Sometimes yes, sometimes no</th>
<th>I am completely with my routine</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td>6</td>
</tr>
<tr>
<td>7</td>
<td>8</td>
<td>9</td>
</tr>
<tr>
<td>10</td>
<td>10</td>
<td></td>
</tr>
</tbody>
</table>

*Figure 4.5. Test User 1, second questionnaire.*
Test User 2

Summary

Test User 2’s experience was much like Test User 1’s in that many bugs had not yet been fixed. In addition, Test User 2’s use brought two more program flaws to light. First, he was unable to reach the phone and the touch screen while at the same place. The phone position was changed to make this easy. Next, the sound-activated notification was inadvertently cued numerous times, causing great frustration. The microphone sensitivity had been turned up because it was in a remote position. The microphone position was changed to be closer to the user, thereby allowing a lower sensitivity setting on the microphone.

Test User 2 Questionnaire

For this and all questionnaire discussions below, the questionnaire answers will be summarized in a table. The questions are abbreviated to save space. Refer to full questionnaires above for the entire question.

Some questionnaire answers from Test User 2 were of note. Even though the user had so many bugs to deal with, he was still able to see the benefits of the system by realizing some of the further possibilities of this system in questions #1 and #2. Mostly, the focus of Test User 2’s comments were on the possibilities of making a system that actually performed physical actions for the user such as opening the front door for friends (#7).
Table 4.2. Test User 2 Questionnaire

<table>
<thead>
<tr>
<th>Question</th>
<th>Response</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Do anything helpful?</td>
<td>The emergency one was pretty good. If someone’s hurt, they could yell 911.</td>
</tr>
<tr>
<td>2. Surprised by anything?</td>
<td>The program interacts with the computer and gets you online. And also this remote was pretty neat, how it shows you the TV guide.</td>
</tr>
<tr>
<td>3. Things you wish it could do?</td>
<td>“Make my dinner” (facetiously) it would be neat to play games, like chess and what time of day and the mood you’re in, it would do the level of challenge you’d want it. “In for an easy game of solitaire because I just want to wind down.” Or “I want to play a hard game of chess.”</td>
</tr>
<tr>
<td>4. Annoying?</td>
<td>Constant buzzing because the sound sensor is too sensitive.</td>
</tr>
<tr>
<td>5. “No” button good enough?</td>
<td>I think that would be good if you could use voice command. Not long sentences, just one little thing.</td>
</tr>
<tr>
<td>6. Others things not pleasing?</td>
<td>Other than the plates were empty (facetious) I think it’s pretty neat, I really like this idea.</td>
</tr>
<tr>
<td>7. Anything else?</td>
<td>Something like the cupboard door opener but for my front door while I’m watching TV.</td>
</tr>
</tbody>
</table>

**Test User 3**

**Summary**

Test User 3 was able to have a better experience with the system because the basic bugs had been fixed. The attempt at using a software delay to keep programs from opening repeatedly for user was abandoned. From this test user on, the system used a moderator-controlled hardware switch to let the system know when the user had closed programs. This is sometimes called a “Wizard of Oz” situation where a human performs actions the test users suppose are being performed by the system.
Various problems were reinforced while observing Test User 3 and other new problems became apparent. First, the fact that the explicit instructions were distracting to use became very clear. The user had to navigate a packet, which included the initial instructions and the questionnaire while using the system. The instructions to follow to simulate a day covered 3 pages, necessitating flipping pages while using the system. This distracted the user from the key experience of using a context-aware system. In addition to being difficult to handle the packet, the users mistakenly supposed the system required them to follow explicit steps to get a desired response. This undermined the goal the testing which was partly to exhibit a context-aware system that does not follow rules to elicit system responses.

The reason for having the explicit instructions (versions one and two) was so the users could see the wide-ranging abilities of the system. However, because the main strength and unique characteristic of this particular system was overshadowed, this was eventually changed. The next 4 users continued to receive the explicit instructions although their weakness became more and more apparent the more times the author saw them used. Nevertheless, these first 6 users were still able to provide useful feedback on the system and their experience with it.

Another problem noticed while observing Test User 3 was the problem users had while changing the time of day manually on the screen. Users had to click a circular dial but the exact place to click was not clear. This compounded the inconvenience of a step the user should never have had to take because in a real setting the time of day would merely be recorded by the system instead of selecting it. The next 3 users continued with
this style of time change until it was changed to a more easily understood slider with Test User 7.

The moderator also noticed that there was not feedback to the user to accompany the physical opening of the artificial cupboards. According to Norman (1998), verifying actions the system has taken to the user helps the user to know the action has been taken and therefore that he does not need to try further. For the first 2 users, Test User 1 and Test User 2, there was only a nondescript computer “ding” when the channel flipping prompt was displayed. Software was acquired and configured that allowed the author to specify any text for the system to say at any point in the program. The voice was a simulated computer readily available from Microsoft’s “Agents” development site.

At this point, with user Test User 3, the voice only prompted the user to be patient and wait for the slow system to open a service or program for the user. Basically, the computer voice was programmed to remove the necessity for the moderator to make the same comment or give the same instructions repeatedly for each test user. As more prompts and helps became obvious, they were added as voice statements from the system.

Other software issues on a smaller scale also became evident while observing Test User 3. The user had to be reminded to not accidentally cover a sensor on the table with the instruction packet while the system was recording the cues. Simplifying the instructions to one sheet and taping them down to the table later solved this. In addition, the title bar and many of the choices for the user to click were too small to easily select using the “touch pen” and the Sony touch screen. Making the title bars and fonts larger by about double later solved this.
Test User 3 Questionnaire

A few of the questionnaire responses by Test User 3 are notable. This user was one of the users who answered a questionnaire on initial use and then was contacted at a later time to fill out a newer questionnaire weeks later when the questions had been revised (see Test User 3 Questionnaire #2). Although Test User 3 was a bit annoyed by some of the bugs in the system, and she noted that she didn’t think this kind of system would be very helpful to her (questionnaire 1, #1), she added that she already has a program that does similar things to what this system does (q 1, #5). With the added complications of the bugs and the cumbersome instruction set, she most likely was not able to grasp that this system is able to do what her program (which she likes) does and that it takes it a step further and adds automatic learning and proactive suggestions.
Table 4.3. Test User 3 Questionnaire

<table>
<thead>
<tr>
<th>Question</th>
<th>Response</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Do anything helpful?</td>
<td>Naw, these things are routine enough that I do them pretty quickly myself. Now, on the other hand, doing the dishes would be good… or autoprogramming itself to watch TV. First-aid program good idea – bad in a house of noisy kids.</td>
</tr>
<tr>
<td>3. Things you wish it could do?</td>
<td>Naw [added later from conversation by moderator: cabinet solenoid being so loud and close to microphone that it activated the first aid prompt]</td>
</tr>
<tr>
<td>4. Annoying?</td>
<td>I don’t precisely turn off cartoons and then move bowl. These are similar sets of circumstances but no order.</td>
</tr>
<tr>
<td>5. “No” button good enough?</td>
<td>Perhaps. There’s a nifty program I like to use on my computer that opens files and programs at a scheduled time of day. If it knew to turn on Internet, turn such and such on, and play music… that'd be nice.</td>
</tr>
<tr>
<td>6. Others things not pleasing?</td>
<td>bed – Wake up J___! [name of test user] In the morning, it is difficult to get everything together quickly as some things (news, breakfast) are skipped over as they take too long to do and then undo</td>
</tr>
</tbody>
</table>

Table 4.4. Test User 3 Questionnaire #2

<table>
<thead>
<tr>
<th>Question</th>
<th>Response</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Want in other things?</td>
<td>That sort of integration would be fantastic, pending that there is a &quot;kill all&quot; sort of option when you don't need a car to start sort of thing. Lights on and off also good, TV/VCR recording.</td>
</tr>
<tr>
<td>2. Helpful or annoyance?</td>
<td>5</td>
</tr>
<tr>
<td>3. Relevant to task at hand?</td>
<td>7</td>
</tr>
<tr>
<td>4. Free from computer use or bog down?</td>
<td>8</td>
</tr>
<tr>
<td>5. Could use it naturally?</td>
<td>8</td>
</tr>
<tr>
<td>6. Kept your interest?</td>
<td>n/a</td>
</tr>
<tr>
<td>7. Other items?</td>
<td>n/a</td>
</tr>
</tbody>
</table>
Group Two: Learning to Pseudo-Mature Use Phases, Instruction Set Version One

The second group of users was able to interact with a system where the major bugs were worked out. These users had much the experience desired by the author, yet the instruction set was still not optimal because of its explicit nature. Thus, this group of users can still be classified to be using an inferior instruction set, yet since there were not major bugs, they were still able to have a relatively positive experience with the system and present useful information about more mature use of the system for analysis than those who mostly struggled through mishaps and had to use a large measure of imagination to give an evaluation of their experience.

Test User 4

Summary

With test user Test User 4 the explicit instruction packet still overly complicated the user experience. However, with Test User 4 the computer voice prompts were noticeably pleasing and helpful to the test users. When Test User 4 was asked if he had any comments after use, he gave a number of suggestions. The quantity and depth of understanding in his suggestions and comments points toward a successful experience with the ContexTable. If the experience were not the desired one, it is supposed that the user would not have been prompted to desire the application of the technology to other areas of life and likely would not have brainstormed about the benefits of such a system. A test user suggesting other items for which this technology could be useful signals that the user has had a positive experience with the system. It is proposed that if a user begins to brainstorm about applications of this technology to other daily items, then the system has successfully displayed its capabilities to at least the proof of concept stage. This
same correlation (between suggesting other daily items for application and general success of this system) will be used as a general marker for system success throughout the remainder of the chapter.

Test User 4 stated that he is interrupted and therefore loses his train of thought very easily. He stated that he would desire a system such as this one in cars, trucks, planes and trains to aid in the sequential actions sometimes forgotten by people. He gives the example from his own life of the landing preparation sequence for a large airplane. He tells of a situation that caused a plane to almost land with the landing gear not deployed before someone caught the error just before touchdown. He noted that a system such as this one could be implemented in an airplane as an aid to crewmembers. He said that it would be effective if a crewmember were alerted to a possible mistake and given the opportunity to simply accept or reject the option, as in this simulation. In this manner, the system is not performing the actions, but merely acting as a “remembrance agent” for the crew in vital sequences.

Table 4.5. Test User 4 Questionnaire

<table>
<thead>
<tr>
<th>Question</th>
<th>Response</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Do anything helpful?</td>
<td>Yes, it anticipated my habits.</td>
</tr>
<tr>
<td>3. Things you wish it could do?</td>
<td>Yes, same goofy voice each time.</td>
</tr>
<tr>
<td>4. Annoying?</td>
<td>I don’t precisely turn off cartoons and then move bowl. These are similar sets of circumstances but no order.</td>
</tr>
<tr>
<td>5. “No” button good enough?</td>
<td>Part of the time.</td>
</tr>
<tr>
<td>7. Anything else?</td>
<td>Bed and alarm example. Also, such a system would be useful in a large 18-wheeler: many complicated steps.</td>
</tr>
</tbody>
</table>
Test User 4 Questionnaire

Test User 4 has a strong technical background so none of the capabilities were very new to him. He had issue with the some of the constraints of the system in this prototype stage (#3, #4), but that is normal considering the limitations of asking users to evaluate a prototype.

Test User 5

Summary

Test User 5 tested the system with explicit (version two) instructions also. He also experienced the same problems and reinforced the need for a change to the size of buttons and selections and with the difficulty in time of day selection. However, Test User 5 was able to proceed from Learning to Mature Use rather quickly. One thing that appeared to be a big help was the surprises he had while using the system that helped him to maintain interest and seek to stick with the system even while he was not used to it (Selker & Burleson, 2000). Comments Test User 5 made when the system gave him a prompt or activated a service automatically clearly show his interest in and wonder of the system. His comments (transcribed from video tape) are summarized in Table 4.6.
Table 4.6. Test User 5 Comments

<table>
<thead>
<tr>
<th>System Action</th>
<th>User Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Music suggested</td>
<td>“That’s awesome, huh, huh!” (repeated twice)</td>
</tr>
<tr>
<td>Music automatically opened</td>
<td>“Nice!” it didn’t say “do you want to…”</td>
</tr>
<tr>
<td>Second time music was opened</td>
<td>“Hey, it kept my volume setting!”</td>
</tr>
<tr>
<td>Please wait, I am opening a program for you</td>
<td>“That’s awesome!”</td>
</tr>
<tr>
<td>Opened first aid help after loud noise</td>
<td>“No way! Frostbite, let’s do frostbite!”</td>
</tr>
<tr>
<td>User repeated action for first aid</td>
<td>Laughed when it opened – seemed to be exploring the capabilities of the system.</td>
</tr>
<tr>
<td>Opened local TV listings</td>
<td>“How did it know where to go?”</td>
</tr>
<tr>
<td>End of session</td>
<td>“I want to keep going and trying different combinations.”</td>
</tr>
</tbody>
</table>

The statements above show that it is most likely the user had a positive experience with the system. His interest in continuing and trying different combinations of cues and actions he had not had a chance to explore shows that he had become comfortable with the style of interaction allowed with a context-aware system. After only a few minutes, the user stopped referring to the instruction set and merely proceeded to interact with the system in what seemed to be a natural manner. The extra bandwidth between him and the system was being utilized and the user was enjoying it.

Test User 5 Questionnaire

The suppositions about the Test User 5 enjoying the system are supported in his questionnaire. In #5 he even states that he would like to have such a system himself, saying “It’d be fun…if money weren’t an issue, I’d get one.” However, he also pointed out that this kind of thing might be more useful in the morning when his habits are more concrete (#4) and not as much at other times of the day when his habits are not as strong. This statement was relatively common among test users. However, what they might not
have seen is that the system works equally well when user do not do things every day, but it just takes longer for the system to learn the habits of the user(s). Therefore, in a long-term use setting, the user comments on this issue would likely be different.

Test User 6

Summary

Test User 6 was the last to use the inferior, explicit instruction set before moving to version three. Nevertheless, his experience with the system was a positive one as evidenced by his comments and the number of possible applications he explored in conversation with the moderator. When the system automatically opened up the day’s news when he placed his bowl on the table, he remarked, “That’s sweet.” He then removed the bowl and repeated the action to see what would happen. The fact that he learned that he could initiate system actions by moving around objects on the table is strong evidence that Gardner’s Body-Kinesthetic Intelligence is being used. In this manner, the system interaction has moved away from symbolic (clicking icons) and towards intuitive (going about normal activities). At this point of the test session, Test User 6 comments, “That would be pretty sweet if you just sat down and you put your breakfast in front of you on the table and your computer automatically opened up whatever your usually read in the morning.”

When he was trying to activate the first aid by yelling “Help!,” the system did not initially react to him. This was a common problem with the system, and at times, help from the moderator was required to cause the system to respond correctly. However, Test User 6 repeated yelling “Help!” multiple times with no adverse comments until the system responded with the first aid prompt. This could be because he had confidence the
system would do something useful and/or interesting. Here we see another confirmation that the system meets Selker/Burleson’s requirement for successful context-awareness of making the user want to explore and discover more about system. When computer recognized and presented prompt, the user smiled broadly and said, “That’s cool!”

**Test User 6 Questionnaire**

In the questionnaire, Test User 6’s feelings about the system are clear. He noted that he thought the calling for help would be particularly helpful for someone who is older and living alone if it could be attached to an automatically dialed phone (noting that the first aid is also helpful). His experience was so positive that he began to see the possibilities of this kind of system being applied further. He commented,

> “Could you do it where if you take your shower at 6:00 in the morning and at 5:58 it gets the water up to the correct temperature and you jump in your shower and your don’t have to do any of that stuff? You know what else would be so cool. It knows that you set the alarm for 5:50 but except on Wednesdays, you set it for 6:50 because you don’t have to be in until 9:00, or whatever, so it changes your alarm and you don’t have to change that.”

Enjoying the learning progression of the system, he said, “That was really cool when it didn’t ask me if I wanted to open Wells Fargo [bank access], it just opened Wells Fargo.” He continued to brainstorm applications for the technology and how it might help a mother when dealing with children.

> “If you pushed it to the right level, I definitely think it could be a useful thing. Particularly the help feature, how cool for a mom when her kids sat down for cereal if the TV turned on automatically. But then, not just things at a table, but
all around the house you could do a lot of things for a mom, like waking kids up
on time so that she wouldn’t have to do them herself. If you think out there if this
thing were expanded exponentially, then it would be really helpful.”

Test User 6 placed an important and interesting condition on his willingness to
incorporate a system like this in his daily life because of his relatively moderate technical
experience. He would only want something like this on the condition that it was user-
friendly enough. He said that if it were too difficult to set up and understand, you might
as well not use it. Because Test User 6 was concerned about if the system would too hard
for most typical, non-technical users if taken to a higher level, the moderator explained
that the system really just works by allowing the user to go about normal activities and
then just begins asking yes/no questions. The user then agreed, “if it were just based on
answering yes/no questions, then definitely, it is great.”

With this concern aside, the user began again to state scenarios he might deem
this kind of system most useful. He said,

“I just envision waking up, getting into a shower that is already hot, then getting
out and my breakfast is all ready and CNN is on. Then, when I’m done, it takes
my bowl and washes it off and then my car is outside warming up for me so I can
just head out.”

This quote shows that the user has caught the vision of the possibilities of context-aware
systems, and, at least in his mind, has already moved to the true mature user phase.
Again, the fact that the user has caught the vision and can see other applications in life is
being used as support that this particular system has exhibited enough of the success
characteristics to be deemed a viable context-aware system even.
Group Three: Learning to Pseudo-Mature Use, Instruction Set Version Two

This last group received a new instruction set. This instruction set (See Figure 4.5.) merely outlined how the user might use this ContexTable but did not give any concrete instructions on how to go about using the system. The instructions gave a skeleton sequence that showed how the system works best, but allowed users to make their own decisions as to what to place on the table and at what times they would perform the different actions. This gave a freedom that allowed the users to see more of the true nature of and strengths unique to this system, and is referred to as instruction set version three. While this instruction set was improved because it allowed a better showcase of the abilities of the system, it also required slightly more help and intervention from the moderator initially. However, later in the simulation the users required less help than the other groups because they were incorporating the new human-computer interaction paradigms into their actions, and not just following instructions by rote.
The large and most of the smaller bugs of the system had been worked out by the time these users encountered the system. Almost no changes were made to the system while these users tested the system and it seemed that users moved easily through the simulation from a learning phase to a point where it could be called pseudo-mature use or they understood the system well enough to project this kind of use in their imagination. Either way, the fact that this system is a prototype, and the simulation was time-

Figure 4.5. Instruction set version three; given to test user group three and four.
compressed from months to minutes, means that the possibilities of the ideas proposed by
the system are as important as the actual interaction with this system.

Group Three Summary

As stated above, Test User 7 was the first to go through the use simulation using
the new, looser instruction set. In addition to this change in protocol, or instruction set,
the time of day selector for the users was changed to a slider instead of a rotary dial. This
slider proved to be much easier to understand and navigate for this and all subsequent
users. Because of this greater ease of use, this and all coming users had a more uniform
experience and their comments were similar. All users in this group were individual
users, yet their reactions to the simulation and their questionnaire answers are only
summarized here. The summary is divided into two parts: (1) moderator observations &
spoken comments, and (2) questionnaire answers.

Moderator Observations & Spoken Comments

Test User 8 had a very technical background and appeared to grasp the manner of
interaction with the system readily. He implemented very complex, but not necessarily
sensible combinations of sensor cues for different contexts. It was more like he was
playing with the system like a beta user, trying to see where it would fail. Because he did
not consistently recreate his complex combinations of cues, the system did not initially
react as he thought it should. This led to a short time of frustration by Test User 8.
However, the system just took longer to learn his habits and then began to respond as he
had hoped.

Test User 8 and Test User 9 both experimented and played with the “voice
activated” first aid help extensively. This could be because they believed it actually
understood their words. The moderator later explained that it only responded to loud noises at this point in development. It is interesting that these and numerous other users were most interested in and impressed with the “voice activated” capability of the system. This appears to go against the premise of this research that learning is more effective than rule-based context-aware systems. This might show that rule based systems are easier to demonstrate to new users and new users might be more initially impressed or intrigued by them. However, a learning system, while harder to demonstrate, might be a more effective system in actual long-term use. Take the example of the much-hated Microsoft “Paper Clip.” In a simulation setting, it would probably be well received by users asked to do things that show the unique capabilities of that system. However, in actual use, it was not well received (Trott, 1998). The data here point toward this system being effective in actual use although some users may not be as initially impressed because some of the rule-based capabilities were more easily showcased in the short time allotted for the test.

Learned or rule-based, the sound activated first aid help and the computer voice prompts were very well received in general by all users. This supports the notion that users prefer other means of interaction with computer systems than the purely symbolic pathways allowed with the keyboard and mouse.

One test user’s comments stand out in this group. Test User 13’s background and area of study is in human computer interactions, explaining his interest in and comments regarding this system. At one point during simulated use, Test User 13 exclaimed, “Patent this baby!” He also noted that this system could keep daydreamers or people who have a hard time staying on task to keep busy on the right things. He also noted that
it would be a helpful remembrance agent for many different professions where procedures must be followed, such as in medicine or repair work. It would probably be prohibitive to write a custom program for every procedure in every profession, so this kind of program could be installed and it would begin to learn the habits of the professional. Then, when she departed from the norm, the system would ask a simple, yes/no question suggesting the absent step. Test User 13 thought this system would be very helpful in such areas.

Users had positive reactions to the automatic prompts of the system after it had learned their habits. Comments of the 14 users from Group Three users are listed here (see Table 4.7.) to show how users commonly reacted to the system.

Table 4.7. Compiled Test User Comments from Group Three

<table>
<thead>
<tr>
<th>Short Comments</th>
<th>Longer Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>“Hey, it got what I wanted!”</td>
<td>“Based on what you have on the table, you’re limited, but when I think about other things that I’d want, I mean you could have a whole automated house.”</td>
</tr>
<tr>
<td>“Oh yeah, so it learns your habits!”</td>
<td>“I like that, that’s really neat. I like how it thinks for you. That’s me, every morning, it’s the same routine over and over and over, and it’s actually time-based really, so.”</td>
</tr>
<tr>
<td>“Ah, hah, hah, smart!”</td>
<td>“That’s nice! Yeah, sure, let’s see where it goes to. The TV guide, hey, what do you know. That would be useful”</td>
</tr>
<tr>
<td>“That’s neat, I like that!”</td>
<td>“It’s pretty neat. I got the idea of what to do and how it works and it responded well.”</td>
</tr>
</tbody>
</table>
These comments show a generally positive overall feeling for the quality of the proactive prompts of the system. Also, the comments show sparked interest and a desire to discover more about the system and continue interaction with it. It is understood that the creator of the system is also the moderator and this could make users give positive comments for fear of hurting the feelings of the system creator. However, since body language, almost inaudible video-taped comments, casual discussion and questionnaire answers all point toward the same positive overall experience, this problem is not perceived to be a major threat to the strength of the data gathered.

**Questionnaire Answers**

A question that asked about if users could envision themselves using this kind of technology in other daily items was intended to show if the users had a positive or negative experience. If the experience were generally positive, they would probably want it, and if it were generally negative, they would likely not want it integrated into other items they use daily. Thirteen of the fourteen users from this group listed items they would want the technology built into. One user noted, “I think as long as I can still easily override the system if I want, it would be great to have it usually do my repeated tasks.”

Another question asked if the users had any surprises that kept their interest and made them want to continue using the system. One user commented, “Each time through the computer learned more - it was interesting to see each time what the computer could do.” This shows that the system, while being a departure from the accepted interaction standards of most users still has the ability to create interest in continued use with discovery and useful actions.
There were many more responses to questions yet they will not all be discussed. For a full listing of all questionnaire answers, see Appendix E. Many of the last questionnaires had quantitative aspects. There were 11 respondents plus or minus about 2 because of respondents leaving the item blank. Users were asked to rate their feelings on a scale of one to ten on different aspects of their experience. The first question asked whether the system was more of a help or an annoyance. Marking one was defined as “Extremely annoying,” five was “Not a help or an annoyance” and ten was “Extremely helpful.” The average of all responses was 8.1; showing that most users likely felt the system provided helpful items. This data suggests that the system provided a high help to distraction ratio and fulfills success parameter #2 from the list at the beginning of this chapter.

Further, a question asked if the system provided services and information relevant to the task at hand. One was “Completely useless,” five was “Would make no difference to me,” and ten was “Extremely helpful.” The average of all responses was 8.6; showing that overall users felt the services and information the system provided were helpful to the tasks they were trying to accomplish in the simulation. This data suggests that the system fulfills success parameter #3 from the list at the beginning of this chapter.

Next, a question asked if this kind of system would free the user from computer interaction or require more work to use. One was “I would have to worry MORE about using the computer,” five was “It would not make a difference,” and ten was “I would be able to pay less attention to using the computer and concentrate more on meaningful tasks.” The average of all responses was 8.1; showing that overall users felt the system would free them from computer use and allow them to concentrate more time to life and
less to using a computer. This data suggests that the system fulfills success parameter #4 from the list at the beginning of this chapter.

Another question asked if the users felt they could incorporate this kind of technology naturally, even subliminally into their normal practices. One was “It is too complicated to be able to work into my normal activities,” five was “It would not make a difference,” and ten was “This kind of communication/collaboration with my computer would be natural, almost subliminal.” The average of all responses was 8.5; showing that overall users thought they could work smoothly with this kind of system in daily routines. This data suggests that the system fulfills success parameter #5 from the list at the beginning of this chapter.

Last, a question asked if users follow their routines closely enough to make this kind of system useful. One was “No, I am completely erratic,” five was “Sometimes yes, sometimes no,” and ten was “I am compulsive with my routines.” The average of all responses was 8.5; showing that, overall, users think they follow routines closely enough to allow the system to work correctly. Interestingly, even if this score were lower, the system could still be made to work. If the user does not follow a routine as closely, the software has one variable that can be easily changed. This one change will allow greater variability in the data while still providing information and services to the user. The only negative to this approach is that is more likely the system will make incorrect suggestions. This data does not support any of the accepted success parameters for a context-aware system, but is applicable to this system because it uses a novel learning technique not before applied in context-awareness.
Group Four: Test Users in Group Settings, Instruction Set Three

Group four, the last group, consists of the results from two user sessions that involved more than one user.

*Group Usage Session: Test User 16 & Test User 17*

This group had two individuals. One user was sighted and the other was legally blind. The experience of the sighted user (16) was generally similar to other users. The blind user could locate a person by sound and some visual cues, but she lives life as most blind people do. This was an interesting session because the possible uses for disabled persons were explored. The test session was relatively uneventful, yet the blind user, Test User 17, made some interesting comments afterward. She noted that this kind of system would be extremely helpful to her, as noted in the first answer in her questionnaire. She noted that she was impressed with how quickly the system learned habits and that it would be able to help her in many daily activities. She brought up the idea that this kind of system could be connected somehow to an emergency line to local authorities that sent an automated message if she stopped responding to the program’s prompts. The author notes that the system could easily be programmed so that if a user is not activating any cues as they normally would be during a day, it would ask the user if everything is all right. If there is not response and the user has not set a vacation time to deactivate it, the system could alert authorities to a possible accident and send help.

She also thought of some more ideas that would be of great help to her. A system like this might be programmed to help her remember where items are in her house. The author suggests that the system might even be able to help her match clothing if it knew which clothes were being taken out a closet. If the user removed two colors that the
system does not have recorded as being taken out together, the system could ask if the user is sure she wants to wear, say, sky blue with navy blue. Since she is blind, such prompts could be of use in helping her to have an independent lifestyle.

*Group Usage Session: Test User 23, Test User 24 & Test User 25*

Test users 18 to 22 were included in the Group Three as part of the summary of that group. Users 23 to 25 were the last users to go through the simulation. Before this group tested the system, additional changes to the hardware and software were made. Since changing the time of day was an unneeded distraction for test users being that they would not have to do so in an actual use situation, this was changed. Two remote switches were added, similar to the “Wizard of Oz” switch used to activate the time delay to take care of the software limitations discussed earlier. The off/on combinations of these two switches, working through digital lines to the software, were made to change the time of day. The software for this is discussed in Chapter 3. This way, the users did not have to change the time of day, but rather the time of day was displayed on the screen where other users had made the selection.

The next change for this group of users was to remove all written instructions. Instead, the moderator explained the idea behind the ContexTable and how it worked. Essentially, the users were given an oral version of instruction set version three. After the three users were comfortable that they understood how to interact with the ContexTable, the simulation began. The length of time needed to understand how the system worked and how to interact with it was about the same as with individual users, showing that with a short explanation, most people can easily understand the new
avenues of communication open between them and the context-aware system. This is a testament to the intuitive nature of the system.

The simulation flowed much as the other simulations did, with a few differences. First, there was interaction among the user group when deciding what to do. In addition, it was interesting to see that the users wanted to test the “Leave reminder to save leftovers” option in the menu. In this manner, the system was being used to differentiate between different social situations. The system worked flawlessly when differentiating between the absence or presence of one of the users in this situation. One problem arose with the microphone used in the first aid prompt. Apparently the slight changes in the loads on the circuitry from the added moderator-controlled switches caused the threshold activation level of the microphone to change. Initially, the first aid prompt was inadvertently being cued often. The microphone was merely moved to a less prominent position to solve the problem.

Several comments were made by the users when the system began to proactively prompt for actions. Users said things like: “Oh, it just came up by itself?”, “So that time it didn't ask, it just did it” and “It’s learning!” These point toward a positive experience because the users both enjoyed and appreciated the prompts and were slightly surprised by them, leading hopefully to increased desire to continue using the system.

When the system performed actions without asking after the initial acceptances of the prompts, the user exclaimed, “I love it that it opens the cupboards for me when I get home and put the groceries on the counter” and “Wow, that’s cool stuff.” Again, these comments show the overall positive experience of the users.
In video recorded discussion, the users were engaged by the moderator in a mini focus group on the technology and how it could affect their lives. Test User 23 noted that he appreciated the concept and overall idea because he thought it would make his life easier by taking care of the little things. This same user began to talk about how he could see a whole house hooked up to a central server running this kind of system. He was asked if he would prefer this kind of system where the user interacts by teaching the system his habits over a system that had pre-written rules. Even when reminded that rule based system can react faster to the user (such as the first aid help or the remote control from this simulation), he noted that it would depend on the circumstances. Test User 25 said that she preferred it the way the ContexTable was designed. She added that she thought it was more effective to have some things be based on rules such as emergency services and things that do not happen often enough at similar times of the day to warrant a teaching style of interaction. However, she did appreciate the fact that the other things could be taught. She noted that she would prefer to interact with a system with most of the options in a teaching scenario.

When he continued by saying that pretty soon our whole house will be automated and a system like this one will just tell everything what to do, Test User 25 said that it would just make people really lazy. After the moderator suggested that maybe it would just let us focus on other, perhaps more important things, she said, “Yeah, that’s right.”

Test User 23 finished with a good summation:

“I think that this is the future. If you don’t think so, you’re just being ignorant. Of course people are going to do this, because it makes your day easier. That’s the
whole point, so we can focus more on things that we really enjoy doing and less on things that are just day to day.”
Summary

Context-aware systems seek to solve the problem of increasing computer complexity and difficulty of interaction. Much of the work to date has been performed on mobile systems such as PDAs and cellular phones because of the supposed ease in determining context of use based on location. All of these systems use some kind of modeling whereby the system determines the services or information to provide to a user depending on whether predefined conditions – the context model – meet the needed values.

The ContexTable built and tested for this research departs from other context-aware systems in various ways. First, the ContexTable is a stationary context-aware system that seeks to determine context of use by sensing people at and objects on or around the table. Another key difference is the ContexTable uses a learning algorithm instead of modeling. This means the ContexTable only suggests services and information to users after they have done so themselves a certain number of times. This solves the problem of knowing how to correctly model a context situation because the user teaches the correct parameters to the system.

Another unique quality of the ContexTable is the software and hardware it is based on. The LabVIEW graphical programming language allows the system designer to get over a hurdle confronted by many other researchers in the field: the difficulty in creating an actual, working system for testing. In addition, because of the way the software was written, it is portable and easily scalable to other applications and settings and operating systems.
The ContexTable, being a prototype, needed to be tested to represent a valid contribution to the field of knowledge. For various reasons, the style of testing employed was qualitative. Within the area of qualitative research, the sub-category of formative evaluation was the style of assessment for the system.

Conclusions

Data was gathered by having test users simulate normal interaction with the system in a 20-minute period. Data discussed in Chapter 4 suggested that the system fulfilled many of the requirements found in the literature to describe a successful context-aware system. In addition, since the evaluation was formative, many changes were made to the system in the course of testing. By the end of the testing phase, the system was taken to a point where the author feels it shows the principles he initially desired to exhibit and is ready to be taken to the next level of development.

More specifically, the data suggest the success parameters outlined in the literature (see Ch. 3, pg. 40 and Ch. 4, Table 4.1) were met in various instances. First, it was observed that various users were able to draw on others of Gardner’s “Seven Intelligences” than those used with a typical computer system while interacting with the ContexTable. Second, users deemed the system to present more helpful services than distractions. Third, most users communicated that the system presented information and services relevant to the task at hand. Fourth, users responded they were able to concentrate more on the task than on the use of the computer system. Last, most users were observed to have been able to incorporate interactions with the ContexTable naturally into their normal activities (at least as far as these simulations could determine).
These comments refer to the system in its current state, or after the improvements made throughout the formative evaluation.

Recommendations

Future work on this system can be in either the software or hardware and will hopefully allow many of the suggestions of test users to be brought to life.

Software

With the software, the prevailing desire of test users was that the system might be able to understand spoken answers to prompts. These would be merely yes and no answers, but it would remove the need for the user to be seated in front of the system to interact with it after initial use. This capability would greatly heighten the usability of the system and further release users from the constraints of current desktop systems.

A simple change to be made to the software if it were to be used in an actual setting and not in a time-compressed simulation as with testing here would be to remove the need for the user to select the time of day and merely record the time within the software. This is a simple matter as the code has already been written but was disabled for testing. Next, the system should have more options for services and information than the 10 allowed on the simulation screen devised for these tests. Possibly, the custom selection screen used here could be removed altogether and the software could be written to interface the operating system and gather information while the user interacts with his customary program selection interface.

An important adjustment to the software that would make it able to deal with people who do not follow routines as tightly as hoped would be helpful. This would allow the system to react correctly when users do not recreate their contextual cues well
enough for the system to recognize it. This weakness is common to rule-based systems as well, but the ContexTable has the strength of eventually being able to learn these erratic behaviors with enough time. This problem was most apparent with situations such as that the user will not pick up the phone at the same time each day to look up a phone number. A possible solution is to compare database entries to find values that change greatly and do not appear to signal a particular context. Because the system records the program selected in the database entry, this comparison could be carried out without substantial additional programming.

A number of test users expressed the desire for a “Kill All” button that would allow them to turn off the additional system capabilities. This kind of a “pause” button would be important if users are not in the mood for this kind of interaction with their computers at a particular time, or if there is a unique situation such as a party where the system’s suggestions may be inappropriate.

**Hardware**

One strength of the ContexTable is that adding more capable hardware is relatively simple. Since the software interacts with the outside world through National Instruments’ Data Acquisition boards, the system designer has easy access to digital I/O and analog inputs. Hardware such as more complex actuators and self-contained robotics would allow the system to fulfill some of the desires of test users to perform physical tasks. While the addition of something like a robotic arm may be cost prohibitive, other desires are more easily accomplished. With the increasing use of electronics in home appliances, the ContexTable could easily send signals to and gather information from
many different parts of the home. Users listed many objects they wished could be integrated into this system.

The possibilities of creating an “intelligent home” are only limited by which daily objects have electronics inside. A laundry basket could signal the need to do a wash and fill the washer with water and soap. The refrigerator could signal the need to go shopping by sensing a low shelf weight. A home security system connected to law enforcement authorities could alert them when the resident does not respond to prompts from the system. The alarm could set itself based on when you go to sleep. Then, the lights could automatically turn on when you should wake up and the shower could heat up before you get in based on the alarm clock. Then, your breakfast items could be presented to you while the daily news (or other entertainment) is already playing at the correct time. The garage door could open and your car could be warming up and defrosting (or cooling down with the A/C) as signaled by the ContexTable.

**Other Recommendations**

All testing to date has been on a simulated time scale of minutes instead of months. It would very valuable to test this system on a longer term and in a more realistic setting such as in the home of a family. This would provide the needed data to see if the system’s strengths over a rule-based system come out.

Lastly, since the software was written to be independent of the application (a table, in this case) it would be interesting to try this software in other applications. For example, instead of one ContexTable system controlling preferences for a car, a car could have its own context-aware system.
The software algorithms alone are simple when compared with genetic algorithms or neural networks. However, this system may have emergent properties not apparent at this time. The author hypothesizes that this system could be used in a network with other iterations of the same software in a layered scheme. Iteration in a lower layer would feed information up to higher levels in the form of the “scores” of each of the database entries. Higher-level instances would feed information down to lower layers in the form of answers to yes/no questions from the human user. Although this application has not been tested and is purely theoretical, the effort needed to change the code to work in this way is not prohibitive. Since all of biology is based on the emergent complexities that arise from the aggregate of vast numbers of lower-level, simpler parts, this theory has some support.
REFERENCES


LabVIEW Resources, (n.d.)


Qualitative and Quantitative Debate, The (n.d.), Writing@CSU: Writing Guide, http://writing.colostate.edu/references/research/observe/com2d3.cfm


Appendix A: Hardware and Software Specifics

**Hardware**

*The Desktop Unit*

*Figure A.1.* Sony VAIO touch screen enabled desktop system.

At the core of the ContexTable is a typical desktop PC system with a few special abilities that made the system suitable for this kind of application. The system used is a Sony VAIO (model PCV-LX920) with a Sony touch screen.

*Figure A.2.* Touch Pen used to select items on screen.
The screen works only with a special “pen” provided with the system. This was beneficial because it allowed users to get a better feeling of leaving behind the keyboard and mouse to engage a system in a new way. Users touch the screen with the touch pen and it acts the same as a mouse click at the touched point. The system was augmented with two National Instruments (NI) PCI Data Acquisition (DAQ) cards. These cards are attached to respective screw terminal boards (see Figure A.3). The boards provide access to eight analog inputs and 16 digital I/O lines each.

![Screw terminal boards](image)

**Figure A.3.** Two green NI screw terminal boards with white breadboard in middle.

To gather information from the surroundings, various sensor arrays were built and attached to the DAQ cards. The sensor arrays discussed earlier in this chapter were attached to the DAQ cards after some signal amplification. Each FSR was wired in a
voltage divider circuit (see Figure A.4). The measuring resistor (RM) value was 3K ohms. Pertinent excerpts of Interlink Electronics’ FSR data sheet are provided for reference at the end of this appendix.

The CDS photoresistors were implemented in the same kind of circuit. This allowed the circuit to be run from a single power supply set at 5V DC and wired in parallel for all the individual sensor arrays. The measuring resistor used for the CDS photoresistor circuit was 5.1K ohm. The power supply is a standard desktop unit (see Figure A.5).

Figure A.4. Voltage divider circuit & op-amp in voltage follower (buffer) mode.
All of the circuits described above were created on a breadboard and this board, along with the two National Instruments screw terminal boards, were mounted to an aluminum case to simplify the setup. The aluminum case also holds a simple circuit to actuate the solenoid used to pop open the false cupboard door.

**Figure A.5.** Wiring Diagram for solenoid powered with 20V and software switching.

The solenoid (see Figure A.5) is powered with +20V DC from the same desktop power supply and is switched through a reed relay (Magnecraft Model W172DIP-5) switched with a digital output line from the National Instruments board.
The microphone used to gather sound information to activate the emergency first aid capabilities is a simple condenser microphone (see Figure A.7). It is powered with 5V from the screw terminal board and returns a variable analog signal proportional to the sound level detected by the microphone.

Unlike many test users supposed, the microphone did not recognize any language, but merely reacted to loud noises. It was implemented in this way to simplify the project and because the focus of the project was not voice recognition.
The remote “Wizard of Oz” switches (See Figure A.8, discussed in depth in Chapter 4) used to change the time of day and to perform the duties that went beyond the capabilities of the software were simply toggle switches with long wires attached to digital lines on the National Instruments boards (Models NI PCI-6014 and NI PCI-6024E Basic Multifunction I/O Boards).

Software

The User Interface

The user sees a very simplified menu that hides almost all of the different things going on in the code in the background. The numbered list is the options the user could select while using the ContexTable. The “Time of Day” selector was where most users made the selection for the time of day in order to accelerate time during the simulation. After a few changes, the rotary dial in the first version was replaced with a slider and was easier for users to interact with. In the last test session, the slider was just an indicator to show the user group the time of day (as switched remotely by the moderator). In
addition, the “End Program” button was changed to read “The ContexTable” because it was not necessary for users to be distracted by the extra button.

Figure A.7. The first version of the interface.

Figure A.8. The second and final version of the interface.

After the user selected each option, a service or program is opened by the ContexTable. They will be shown here in the order they are given on the user menu.
1. *Today’s News.* This selection opens Internet explorer and shows CNN.com which allows the user to see the news of the day. The screen might look something like Figure A.9. With all services and prompts presented to the user, the program could not close them. The user had to close each item the normal way: by clicking the “x” in the top corner of the screen, or selecting the appropriate button on LabVIEW programs (such as “All Done,” or “Close Window”).

![Figure A.9. Presented to user after clicking “Today’s News.”](image-url)
2. **Phone Directory.** This selection opens whitepages.com in Internet Explorer.

*Figure A.10. Presented to user after clicking “Phone Directory.”*
3. **Personal Finances.** This selection opens wells Fargo.com. Admittedly, not all test users will belong to Wells Fargo Bank, but the point is illustrated nonetheless.

![Figure A.11. Presented to user after clicking “Personal Finances.”](image)
4. *Watch Cartoons.* To simplify the hardware, a flash animated web cartoon was used called homestarrunner.com. This allowed the animation to be loaded on the hard drive and presented easily. An example is shown in Figure A.12.

*Figure A.12.* Presented to user after clicking “Watch Cartoons.”
5. Roll Dice for Board Game. This selection brought up a LabVIEW window with a program written to provide the user with a set of numbers between one and six as would two dice. The screen appears as in Figure A.13.

![Figure A.13. Presented to user after clicking “Roll Dice for Board Game.”](image-url)
6. Leave Reminder to Save Leftovers. This selection brought up a bright display notification reminding the user(s) to save leftovers for a missing family member as in Figure A.14. This notification was also accompanied by a computer-simulated voice to aid in getting the user’s attention.

![Reminder Notification]

*Figure A.14.* Presented to user after clicking “Leave Reminder to Save Leftovers.”
7. and 8. Play “Soft Music” or “More Entertaining Music.” This selection brought up Windows Media Player and automatically began playing either a selection of soft music or something more entertaining, depending on the selection made by the user. Either way, the screen presented is similar to Figure A.15.

![Figure A.15. Opened for “Play Soft Music” or “Play More Entertaining Music.”](image)

9. Open Cupboards. This selection did not open any visual prompt for the user, but rather activated a solenoid which popped open a mock-cupboard box. The audible click was also accompanied by a computer-simulated voice, which alerted the user to the fact that the cupboard door had been opened.
Appendix B: Software: Code Documentation

Because the code is graphical, documentation takes a great deal of space and the hierarchy of the individual components is easily confused. Figure A.16. above shows the complete VI hierarchy with system VIs. The “main” program controlling all subVIs and the other core algorithms is called learn 2_2_2004. This program and all subVIs it uses are held in learnsave.llb.

Figure A.17. on the next pages shows the same VI hierarchy, but with system VIs removed so the entire diagram is more clearly displayed.
Figure A.16b. Complete VI Hierarchy diagram (continued).
Figure A.17. Entire VI hierarchy with system VIs not shown.

Figure A.17. shows this same hierarchy, but excludes VIs that are system VIs called by the program and not written by the author. This hierarchy is referenced by the documentation below, which gives a smaller version of this hierarchy for each subVI and disregards parts of the above diagram that are not pertinent.
Below, the code for the “main” ContextTable program is documented. The documentation for graphical code can be confusing because there are many hidden aspects of the code from conditional loops and statements. LabVIEW code maintains all of these hidden windows for easy viewing in the software, but they are not as easily navigated when printed. The code below begins with the highest-level loop and proceeds downwards. Each time a new hidden window is displayed, all the accompanying windows are also displayed so that upon careful examination, they could be pieced back together to form the original code.

After the documentation of the “main” program, each subVI will documented similarly, as described above.
Figure A.18. Front panel of the ContexTable program. Only the lower portion was visible to the test users.
Note: To save space in the code documentation, all figures below are not labeled using APA style. Each unlabeled figure is either understood by examining diagrams above it, or is described in the code comments.
loops|My Documents|Thesis stuff|Table Software|Logs|

- adds current date and time to file name to distinguish file records.
- creates a log file of the data gathered after each running of program. Each log is time stamped and found in the path to the left.

"Delay Proc" below is currently being activated with remote, moderator controlled "Wizard of Oz" hardware switches because LabVIEW cannot determine time of closing of programs. See "Man Delay" subdiagram far left of diagram.

loads the existing array to find if the current values match any existing row. If there is a match, then a score should be increased. If the values represent a new data set, then a new row of entries should be created.
I am opening a program for you.
130
Do you want to play soft music?

Incrs Score

Bar until Any

is My Documents/My M.

If True

else wait, I am opening a program for you.
since 1st loop has no data, waits for 1st user action.
default value: title of menu
(do nothing)

Open Browser

Reset using local variable after read

Choose a Program

http://www.whitepages.com/

Reset using local variable after read

Choose a Program

Choose a Program
Data Population loop: populates the array with data upon user action.

1. Resolution (1 high, 5 low)
   - Match all no score

2. Valid Row
   - add new row b/c a prog was selected to run

3. occurs when a new set of values (row) does not match (within limits) any of the ones in the existing array

4. searches the existing array to find if the current values match any existing row. If there is a match, then a score should be increased. If the values represent a new data set, then a new row of entries should be created.
occurs when a new set of values (row) does not match (within limits) any of the ones in the existing array.

add new row b/c a prog was selected to run

do not add a row b/c no prog was selected

increases the score of the row where the match was found so that it can be executed automatically when score is above thresh.

occurs when the new set of values (row) has a match (within threshold) of a row in the existing array.
do nothing to the existing array. this part of the case is executed many times a second and is constantly checking to see if anything has changed since the last time values were polled.
Single Sensor Actions:
This code will perform actions that only need one sensor to activate them. This will make it so that
one action (lifiting up phone, cold food, weight on table) will activate the context instead of it being
caught. This removes the problem where an action that is not performed often will not be recognized
for too long at a time.

This program will
open a browser
and display a web
site with information
on first aid and
emergency with a
sound mixer, such as
a scream from a hurt
person.

Open a browser with
www.askagap.com if user
lits up phone.

These currently disabled
because the SMS/MMS
been entered. I couldn’t
decide if I wanted to
activate it any time a
phone is lifted, or only
when the user teaches it
to do so in certain
circumstances.

To re-enable, enter digits for
each of the 4 values such as
1 2 3 4 which would be the
voltage levels that appear when
the phone is lifted.

If remote button pressed
quickly (as in searching for
a TV program to watch) then
online TV guide is presented.
Decrease Score.vi

Connector Pane

Front Panel

Block Diagram

List of SubVIs and Express VIs

Position in Hierarchy

Array Height.vi
Front Panel

Block Diagram

List of SubVIs and Express VIs

Position in Hierarchy

brwsr vis cls cartoons.vi
Front Panel

Block Diagram

List of SubVIs and Express VIs

Position in Hierarchy

leftovers.vi

Connector Pane
Front Panel

Don't forget to save leftovers for the missing family member!

Block Diagram

Don't forget to save leftovers for missing family member.
Don't forget to save leftovers for missing family member.
List of SubVIs and Express VIs

**speak.vi**

D:\Thesis stuff\Table Software\SubVI's\speak.vi

Position in Hierarchy

```
speak.vi
```

160
Connector Pane

What to say → speak

Front Panel

What to say

Block Diagram

List of SubVIs and Express VIs

Position in Hierarchy
Increase Score.vi

Connector Pane

Front Panel

Block Diagram

List of SubVIs and Express VIs

Position in Hierarchy

match all for autorun.vi
Connector Pane

Resolution (1 high, 5 low)  Match all no score  Boolean
Current Values  Entire Array
Entire Array

Front Panel

Current Values  Entire Array  Boolean

Resolution (1 high, 5 low)

Block Diagram

List of SubVIs and Express VIs

auto run program exact no score.vi

D:\Thesis stuff\Table Software\SubVI's\auto run program exact no score.vi

auto run program prog hi.vi

D:\Thesis stuff\Table Software\SubVI's\auto run program prog hi.vi

auto run program exact.vi
D:\Thesis stuff\Table Software\SubVI's\auto run program exact.vi

Position in Hierarchy

auto run program exact no score.vi

Connector Pane

Resolution (1 high, 5 low)
Current Values
Whole Array
Within Limits?
**Front Panel**

**Block Diagram**

**List of SubVIs and Express VIs**

- **Break row to elements 2.vi**  
  D:\Thesis stuff\Table Software\SubVI's\Break row to elements 2.vi

- **Within limits 2.vi**  
  D:\Thesis stuff\Table Software\Within limits 2.vi
Break row to elements 2.vi
Within limits 2.vi

Takes multiple elements and determines whether they are close enough to their counterparts. Match each element with its corresponding number (2 with 2B). Allowable error determined by Resolution with 1 being high resolution (low allowed error) and 5 or more being less and less resolution, on to any number desired.
Block Diagram

List of SubVIs and Express VIs

**compare.vi**

D:\Thesis stuff\Table Software\learnsave.llb\compare.vi
Position in Hierarchy

compare.vi

Connector Pane

Front Panel

<table>
<thead>
<tr>
<th>New Element</th>
<th>Within Limits?</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

Old Element from array

| 0.00         |

Error Threshold

| 0            |
auto run program prog hi.vi
**Connector Pane**

Resolution (1 high, 5 low)  
Current Values  
Whole Array  

**Front Panel**

Current Values  
Whole Array  
Resolution (1 high, 5 low)  

**Block Diagram**

Whole Array  
Current Values  
Resolution (1 high, 5 low)  

**List of SubVIs and Express VIs**

- **Break row to elements 2.vi**  
  D:\Thesis stuff\Table Software\SubVI's\Break row to elements 2.vi  

- **within limits prog hi.vi**  
  D:\Thesis stuff\Table Software\SubVI's\within limits prog hi.vi
Position in Hierarchy

within limits prog hi.vi

Connector Pane

Resolution (1 high, 5 low)

Within limits?

2B
3B
4B
5B
6B
2
3
4
5
6
Front Panel

Block Diagram

List of SubVIs and Express VIs

- compare.vi
Position in Hierarchy

auto run program exact.vi

Connector Pane
**Front Panel**

**Block Diagram**

**List of SubVIs and Express VIs**

- **Break row to elements 2.vi**
  - `D:\Thesis stuff\Table Software\SubVI's\Break row to elements 2.vi`

- **Within limits.vi**
  - `D:\Thesis stuff\Table Software\learnsave.llb\Within limits.vi`
Within limits.vi

Takes multiple elements and determines whether they are close enough to their counterparts. Match each element with its corresponding number (2 with 2B). Allowable error determined by Resolution with 1 being high resolution (low allowed error) and 5 or more being less and less resolution, on to any number desired.
List of SubVIs and Express VIs

compare.vi

D:\Thesis stuff\Table Software\learnsave.llb\compare.vi
Position in Hierarchy

match all for autorun with score.vi

Connector Pane

Resolution (1 high, 5 low)  Match all for autorun
Threshold Array
Outgoing Array

Front Panel

Threshold
Array
Outgoing Array

Resolution (1 high, 5 low)
List of SubVIs and Express VIs

**auto run program.vi**
D:\Thesis stuff\Table Software\SubVI's\auto run program.vi

**auto run program 3.vi**
D:\Thesis stuff\Table Software\SubVI's\auto run program 3.vi

**auto run program 2.vi**
D:\Thesis stuff\Table Software\SubVI's\auto run program 2.vi
auto run program.vi

Connector Pane

Resolution (1 high, 5 low)
Current Values
Whole Array
Threshold
Within Limits And Above Thre...
Front Panel

Block Diagram

List of SubVIs and Express VIs

- **Break row to elements.vi**
  
  D:\Thesis stuff\Table Software\SubVI's\Break row to elements.vi

- **Within limits.vi**
Position in Hierarchy

Break row to elements with score.vi

Break row to elements.vi

Front Panel

<table>
<thead>
<tr>
<th>Incoming Array</th>
<th>Second Element</th>
<th>Third Element</th>
<th>Fourth Element</th>
<th>Fifth Element</th>
<th>Sixth Element</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
Connector Pane

Front Panel

Block Diagram

List of SubVIs and Express VIs
auto run program 3.vi

Connector Pane

Resolution (1 high, 5 low)
Current Values
Whole Array
Within Limits?

Front Panel

Current Values

Whole Array

Resolution (1 high, 5 low)
**List of SubVIs and Express VIs**

- **Within limits 2.vi**
  
  D:\Thesis stuff\Table Software\Within limits 2.vi

- **Break row to elements 2.vi**
  
  D:\Thesis stuff\Table Software\SubVI's\Break row to elements 2.vi
Position in Hierarchy

auto run program 2.vi

Connector Pane

Front Panel
List of SubVIs and Express VIs

**Within limits 2.vi**

D:\Thesis stuff\Table Software\Within limits 2.vi

**Break row to elements 2.vi**

D:\Thesis stuff\Table Software\SubVI's\Break row to elements 2.vi
## Front Panel

![Front Panel Diagram]  

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Device 1 Scaling</th>
<th>Device 2 Scaling</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Time of Day (Military)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
</tr>
<tr>
<td>Noon</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Score</th>
<th>Hour</th>
<th>1</th>
<th>2</th>
<th>3 prog</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8 (none)</th>
<th>Z.1 seat 1</th>
<th>Z.2 seat 2</th>
<th>Z.3 seat 3</th>
<th>Z.4 seat 4</th>
<th>Z.5 seat 5</th>
<th>Z.6 seat 6</th>
<th>Z.7 seat 7</th>
<th>Z.8 seat 8</th>
<th>none</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

---

191
List of SubVIs and Express VIs

sensor device 1.vi

D:\Thesis stuff\Table Software\SubVI's\sensor device 1.vi

Timestamp.vi

D:\Thesis stuff\Table Software\SubVI's\Timestamp.vi

sensor device 2.vi

D:\Thesis stuff\Table Software\SubVI's\sensor device 2.vi
Position in Hierarchy

sensor device 1.vi

Connector Pane

Front Panel

Block Diagram

List of SubVIs and Express VIs
Position in Hierarchy

Timestamp.vi

Connector Pane

Front Panel

Block Diagram

List of SubVIs and Express VIs
Position in Hierarchy

sensor device 2.vi

Connector Pane

Front Panel

Block Diagram

List of SubVIs and Express VIs
Position in Hierarchy

Validate incoming row.vi

Connector Pane

Front Panel

Block Diagram

List of SubVIs and Express VIs
Position in Hierarchy

match all for autorun no score.vi

Connector Pane

Front Panel
Block Diagram

List of SubVIs and Express VIs

**auto run program prog hi.vi**

D:\Thesis stuff\Table Software\SubVI\auto run program prog hi.vi

**auto run program exact.vi**

D:\Thesis stuff\Table Software\SubVI\auto run program exact.vi
Position in Hierarchy

brwsr vis cls.vi

Connector Pane

Front Panel
**Block Diagram**

```
SGDocVw.IWebBrowser2 2

Visible

Navigate
• URL
• Flags
• TargetFrameName
• PostData
• Headers

https://online.wellsfargo.com/
```

**List of SubVIs and Express VIs**

**Position in Hierarchy**

```
Learn
Save
```

**browser any file.vi**

**Connector Pane**

```
Enter a URL and run the VI Visible?
```
Front Panel

Block Diagram

List of SubVIs and Express VIs

Position in Hierarchy

Digital Signal.vi

Connector Pane
Front Panel

Block Diagram

List of SubVIs and Express VIs
Position in Hierarchy

analog trigger.vi

Connector Pane

Front Panel

Block Diagram
List of SubVIs and Express VIs

Position in Hierarchy

rapid press.vi

Connector Pane

Front Panel

device
digital channel
line
line state
line state

Timeout for rapid press?* Rapid Press

Number of remote presses in time limit.

Input from shift register (remote presses)
Output to shift register (remote presses)

Input from shift register (just activated?)
Output to shift register (just activated?)

Number of remote presses i...
Output to shift register (r...
Block Diagram
List of SubVIs and Express VIs

**open browser.vi**
D:\Thesis stuff\Table Software\SubVI's\open browser.vi

**brwsr vis any.vi**
D:\Thesis stuff\Table Software\SubVI's\brwsr vis any.vi

**dialog box.vi**
D:\Thesis stuff\Table Software\SubVI's\dialog box.vi

**speak.vi**
D:\Thesis stuff\Table Software\SubVI's\speak.vi

---

**Position in Hierarchy**

---

dialog box.vi
Yes, you guessed right.

No, you guessed wrong.
Block Diagram
1. Create the front panel.
2. For each control you add, right-click the Event structure and select Add Event Case from the shortcut menu.
3. Select the control in the Event Sources list and select the Value Changed event.
4. In the resulting new frame of the event structure, add the code to execute when the control changes value.
5. If you need this dialog box to return more information than “Cancelled?”:
   a) Create an indicator after the loop and wire your information to that indicator.
   b) Attach your indicator to a terminal of the connector pane.
   c) Use “Hide Indicator” to ensure that it does not appear on the front panel.

For an example that illustrates using the Event structure, refer to examples\general\uievents.llb\New Event Handler.vi

For more information about Event structures, refer to Help » VI, Function & How-To Help and find Event structures in the index.
List of SubVIs and Express VIs

Position in Hierarchy
Delay prog.vi

Connector Pane

Front Panel

Block Diagram
List of SubVIs and Express VIs

Position in Hierarchy
sound trigger.vi

Connector Pane

Front Panel

Device

Codialional Retrieval

mode (off) channel index (0)
no change 0
slope (rising) level (0.0) hysteresis (0.0)
no change 0.0000 0.0000
skip count (0) offset (0)
0

waveform data

0

# Scans To Read

0

Scan Rate

0.00

Time Limit

0.00

Input Limits

Buffer Size

Channels

0
List of SubVIs and Express VIs

open browser.vi
D:\Thesis stuff\Table Software\SubVI's\open browser.vi

brwsr vis any.vi
D:\Thesis stuff\Table Software\SubVI's\brwsr vis any.vi
dialog box first aid.vi

D:\Thesis stuff\Table Software\SubVI's\dialog box first aid.vi

speak.vi

D:\Thesis stuff\Table Software\SubVI's\speak.vi

---

**Position in Hierarchy**

---

**Connector Pane**

String ——> [Icon] ——> Cancelled?
Yes, someone is hurt.
No, we're just a little excited.
1. Create the front panel.
2. For each control you add, right-click the Event structure and select **Add Event Case** from the shortcut menu.
3. Select the control in the **Event Sources** list and select the **Value Changed** event.
4. In the resulting new frame of the event structure, add the code to execute when the control changes value.
5. If you need this dialog box to return more information than "Cancelled?":
   a) Create an indicator after the loop and wire your information to that indicator.
   b) Attach your indicator to a terminal of the connector pane.
   c) Use "Hide Indicator" to ensure that it does not appear on the front panel.

For an example that illustrates using the Event structure, refer to examples\general\ui\events.llb\New Event Handler.vi

For more information about Event structures, refer to **Help > VI, Function, & How-To Help** and find Event structures in the index.
List of SubVIs and Express VIs

Position in Hierarchy
Roll dice.vi

Connector Pane

Front Panel

First Di  Second Di

Roll Dice

All Done
List of SubVIs and Express VIs

Timestamp second.vi

D:\Thesis stuff\Table Software\SubVI's\Timestamp second.vi

Position in Hierarchy

---

Timestamp second.vi

Connector Pane

Front Panel
Block Diagram

List of SubVIs and Express VIs

Position in Hierarchy

Manual Delay.vi

Connector Pane

Front Panel
**Block Diagram**

List of SubVIs and Express VIs

Position in Hierarchy

---

digital input.vi

Connector Pane

---

Front Panel

---
Block Diagram
List of SubVIs and Express VIs

Position in Hierarchy
Appendix C: Transcribed Notes from User Session Tapes

Test User 1

- Program made correct suggestions
- Delay not working correctly (frustration, although users were very polite)
- Moderator needed to make numerous interventions to correct invalid suggestions
  - “so if I go to morning, it will stop” [doing the incorrect action]
    - user learned new interaction methods in minutes, showing that interaction method is intuitive
- no sound feedback
- small user notifications that were hard to see and click with pen
  - Test User 1 Questionnaire answers (first users answered questionnaire into camera, not on paper)
    - did the system open anything useful to you? yes, tv channel, and the loud noise, I thought that was helpful
    - what surprised you the most? After the 3rd round having it recognize what I wanted it to do, by just having to change the time of day. It really fits what I want to do because it seems I always do the same kinds of things at the same time of day.
    - Other actions you wish the table could do that it can’t currently do? Make food for me (facetiously). Maybe have more positions than the limited number of sensors. So that you can place objects anywhere on the table
    - Were there any times you thought the ContexTable was annoying? When it kept repeating the same thing that I had already done.
    - Was the “No you guessed wrong” button enough of a help in dealing with these issues? Yeah it was fine, but I wish there was the ability to have more complex scheduling, like doing one thing one day and another thing another day.
      - moderator responded that that was possible and the user was surprised. Apparently, this was not made obvious to test user. He felt as though he was being forced to do certain actions at certain times.
    - What other things were not pleasing to you? “Sitting down. The seat sensor is kinda riding me there.” I’m assuming that when this is implemented, you’re not gonna have wires coming off the remote.
      - “it’s definitely a prototype, but I think it’s a great idea.

Test User 2

- Unable to reach computer to select item while using phone. Later changed that to allow phone use while seated close to system
- Noise sensor activated incorrectly, with a small noise multiple times.
  - Test User 2 Questionnaire answers
- Do anything helpful? The emergency one was pretty good. If someone’s hurt, they could yell 911.
- What surprised you most? The program interacts with the computer and gets you online. And also this remote was pretty neat, how it shows you the TV guide.
- Other actions you wish it could do? “make my dinner” (facetiously) it would be neat to play games, like chess and what time of day and the mood you’re in, it would do the level of challenge you’d want it. “In for an easy game of solitaire because I just want to wind down.” Or “I want to play a hard game of chess.”
- Annoying? Constant buzzing because the sound sensor is too sensitive.
- Wrong button enough? I think that would be good if you could use voice command. Not long sentences, just one little thing.
- Other things not pleasing? Other than the plates were empty (facetious) I think it’s pretty neat, I really like this idea.
- Anything else? Something like the cupboard door opener but for my front door while I’m watching TV

Test User 3

- Instruction format diverted attention and was overly complicating the experience.
- Need to touch in white area for time of day change.
- No audio feedback when cupboards are opened.
- Music always opened up twice
  - changed code
- had to advise user to not cover sensors inadvertently
- title bar options (maximize and close program) were too small to easily select with the touch pen.
  - made title bars bigger.
- Flipping through a packet of instructions was annoying
  - made it a single page and small set of instructions separately given to user and retrieved before test starts.
- “Please wait, I am opening a program for you” first implemented.
- Need to make channel flipping one talk.
- Test User 3 Questionnaire
  - See thesis body

Test User 4

- Explicit instructions still overly complicating test user experience and not allowing him to have the desired experience.
- From the moderator’s perspective, the audio feedback accompanying the notifications were helpful and pleasing to this and future users.
- Multiple pages still distracting to users.
- Test User 4: Any questions or comments?
- User describes a possible scenario where airplane crewmembers interact with a system like this and it reminds them if a step was not taken. It does not perform all actions, yet if something is left out, the crew is prompted to accept or reject suggestion. Specific example of a “wheels up” landing discussed. Either the system could detect the missing step, or since the same thing happened two weeks later, the system could forewarn the crew about similar steps taken and what they led to (wheels up landing, very dangerous).

- I get interrupted and lose my train of thought very easily. That’s why I tend to lose things.

- If there’s a system that requires a quick response. “Shouldn’t your flaps be at 45 degrees at this point?”

- I would want this built into things like cars, trucks, planes and trains with an easy ability to opt out.

Test User 5

- Explicit instructions still
- Still had title bars too small
- Still problem with time of day (clicking on white area)
- When music opened “that’s awesome, huh, huh” (said it twice)
- No audio instructions on dice (realized they needed to be added).
- User did not concern himself with following instructions word or word and had an experience more like what moderator desired
- Audio confirmation of user notifications
- “Nice!” it didn’t say “do you want to…”
- “Hey, it kept my volume setting!”
- “that’s awesome” when “Please wait, I am opening a program for you.”
- “No way! Frostbite, let’s do frostbite!”
- “How did it know where to go?” when opening TV listings
- User did sound activation again and laughed when it opened. Seemed to be exploring capabilities of system.
- “That’s awesome!”
- “I want to keep going and trying different combinations.”

Test User 6

- Explicit instructions
- User had to flip through pages and was distracting
- “That’s sweet” after do you want me to show you today’s news.
- User went back and repeated morning action to see what would happen. User had learned how to interact naturally and smoothly with the system and began to see possibilities. Evan remarks: “That would be pretty sweet if you just sat down and you put your breakfast in front of you on the table and your computer automatically opened up whatever your usually read in the morning.”
User had to repeat saying “Help!” many times, but continued. Could be because he had confidence the system would do something useful and/or interesting. Shows that it meets Selker/Burleson’s requirement for making the user want to explore and discover more about system. When computer recognized and presented prompt, user smiled broadly and said “That’s cool!”

Anything helpful?
  - calling for help. Someone older or living alone. To call for help and even just the first aid. (user saw possibilities)
  - there has to be consistency in your actions for it work correctly.
  - “Particularly impressed with the breakfast thing.”

Surprised you the most? How quickly it learned.
“Could you do it where if you take your shower at 6:00 in the morning and at 5:58 it gets the water up to the correct temperature and you jump in your shower and your don’t have to do any of that stuff?”
“You know what else would be so cool. It knows that you set the alarm for 5:50 but except on Wednesdays, you set it for 6:50 because you don’t have to be in until 9:00, or whatever, so it changes your alarm and you don’t have to change that.”
Wasn’t anything that was particularly annoying. Maybe a better voice.
That was really cool when it didn’t ask me if I wanted to open wells fargo, it just opened wells fargo.
“If you pushed it to the right level, I definitely think it could be a useful thing. Particularly the help feature, how cool for a mom when her kids sat down for cereal if the TV turned on automatically. But then, not just things at a table, but all around the house you could do a lot of things for a mom, like waking kids up on time so that she wouldn’t have to do them herself. If you think out there if this thing were expanded exponentially, then it would be really helpful.
I would want this on the condition that it was user-friendly enough. If it’s too difficult to set up and understand, you might as well not use it.
User was concerned about taking this to a higher level if it would be too hard for the user. Moderator explained that it really just begins asking yes/no questions. User agreed that if it just based on answering yes/no questions, then definitely, it is great.
The drawer opens, it pulls things out, laundry, and lots of different things. I could think of a million things it could do for you that
When I chose my computer, I didn’t choose the Toshiba because it had all these extra buttons and the dell didn’t. I don’t want something if it makes my life more complicated.
“I just envision waking up, getting into a shower that already hot, then getting out and my breakfast is all ready and CNN is on. Then, when I’m done, it takes my bowl and washes it off and then my car is outside warming up for me so I can just head out.”
  - User has caught vision and, at least in his mind, has moved to the “mature user” phase.
Test User 7

- New method of instructions
- Instructions not taped down
- Slider for time of day, easier to navigate for users
- Required more intervention from moderator for user to get through learning stage.
- Video unavailable, see answer sheet.

Test User 8

- New method
- User’s background very technical.
- User implemented very complex, but not necessarily sensible combinations of sensor cues to signal a context.
- System worked correctly and after the user was frustrated a bit, the system learned habits.
- Took a little longer to learn because user did not recreate cues correctly a few times.
- User enjoyed noise activated response greatly, perhaps believing it used voice recognition. Moderator later explained response to loud noise only.

Test User 9

- User experimented greatly with noise activation.
- Showed that system provided surprises that urged the user to continue using.
- However, unfortunately, this capability was not one of the one unique to this system (it was rule-based and not taught). This might show that rule based systems are easier to demonstrate to new users and new users might be more initially impressed or intrigued by them. However, a learning system, while harder to demonstrate, might be a more effective system when in actual use.

Test User 10

- Instructions finally taped down.
  - Most users most impressed with sound activation. This reinforces the desire for users to get away from symbolic interaction with computers and to leave the keyboard and mouse.
  - Test User 11
    - Voice feedback on all items completed. Seems users are able to interact with system more naturally.
    - User very interested in actual voice recognition.

Test User 12

- Housewife.
Very imaginative. Mimed all actions accompanying the situations she was in. Showed she was able to immerse herself in the experience and get a good idea for what this system might be like in real life.

Test User 13

User academically involved in user interface issues as a graduate student.
Comments reflect depth of user understanding and ability to see possible uses.
Remote control clicking “Patent this baby!”
System could help daydreamers and prompt the person to keep going and keeping on task. “yeah, yeah it would.”
Since you can’t write a program for every person, such as in the medical field, this kind of thing is great because it just learns the user’s habits.
Reactions to customers, so customer is an entity. Like at Smith’s.
Moderator: Buying similar items (you usually buy X when you buy Y…)

Test User 14

Very technical.
After opening music “Hey, it got what I wanted!”

Test User 15

Video not working, not able to note in-use comments

Test User 16 (group)

With blind user, Test User 17, below

Test User 17 (group)

With sighted user, Test User 16, above
Help a blind person do something and
Not responding to commands as you should be; system alerts authorities or friends.
Closet that could tell you where your jeans are. Or if your colors don’t match, it suggests you one that does.

Test User 18

Do I need to do it the exact same way as last time?
  • Moderator: Yes, because it takes longer to learn otherwise.
  “Oh yeah, so it learns your habits!”
  “Based on what you have on the table, you’re limited, but when I think about other things that I’d want, I mean you could have a whole automated house.”
Test User 19

- **Good one to show during defense.**
- Cartoons? “ah, hah, hah, smart!”
- Soft music “That’s neat, I like that!”
- 2nd time: “so it did it all by itself, it didn’t ask me”
- 2nd breakfast: “Smart”
- “I like that, that’s really neat. I like how it thinks for you. That’s me, everymorning, it’s the same routine over and over and over, and it’s actually time-based really, so.”
- First aid: “That’s cool!”
- Channels “that’s nice! Yeah, sure, let’s see where it goes to. The TV guide, hey, what do you know. *That* would be useful”
  - shows that user is engaged and interested to see what surprises might follow.
- I like the remote how it makes you flip through quite a few before it makes you stop.
- User wanted to know if system could do more than one thing at once.
- It would know “it’s breakfast time for the person that likes to watch the cartoons. And then at 9:00, this is breakfast time for the person that likes to read the news.

Test User 20

- Today’s news: “Sweet!”
- Ent. Music: ditto
- Leftovers: “That’s sweet!”
- 2nd news: “That’s cool”
- please wait…”: emphatic nodding and “that’s cool”
- don’t forget to save leftovers: emphatic nodding
  - body language more telling that questionnaire most times
- first aid help: pleased laughing and “Sweet”

Test User 21

- user liked open cupboards and repeated the action twice.
  - selker and burleson
- open cupboards: “Cool”
- “It’s not going to just keep doing it if I leave it there. So I have to pick it up and then put it down again.”
- “What if I change my routine. What if I start going shopping every day?” - it would work. that’s a good point.
- It’s pretty neat. I got the idea of what to do and how it works and it responded well.”
- Moderator: you didn’t have to do what I said.
Test User 22

- Today’s news? “Oh, yeah! Voice activated, eventually?”
- “This is pretty, cool, though, an intelligent table!”
- open personal finances?: “Yeah, how did you know!”
- news?: “You’re amazing!”
- news: “It didn’t even ask me anymore.”
- Cupboards (just playing): “Whoa, look at that!”
- User repeated and taught cupboard action to system until it learned with no questions. When it did it without asking, he pushed back in his chair and exclaimed: “Ha, ha, that’s way cool!”

Test User 23, Test User 24 & Test User 25

- More instructions given to group than any other.
- No written instructions to follow
- Moderator changed time of day
- Microphone too sensitive after small changes made in circuit
- Users took same amount of time to learn the system.
  - showing that length of time was not dependent on the instructions, but just getting used to it.
- “Oh, it just came up by itself?” Test User 25
- “so that time it didn’t ask, it just did it” Test User 24
- “Smart Test User 23
- “It’s learning!” Test User 25
- “I love it that it opens the cupboards for me when I get home and put the groceries on the counter Test User 23
- “Wow, that’s cool stuff” Test User 23 when opened music
- if people could interact with their computers in other methods besides just clicking
- “the overall idea, the concept, is something that I appreciate, because it makes your day easier, and they’re trying to look for consistencies. It makes the day run smoother by taking care of all the little things. “ Test User 23
- where would you see it most useful? “I like the automatic setting of the alarm, depending on what day it is. How it can learn your schedule.” Test User 24
- “Usually when you sit down, you want to check your email. I wish I could have something like this that would deal with my email and learn my habits.” Test User 23
- “one thing that I think would be annoying would be if it opened my email and I just wanted to do something else, then I would have to wait for it to open.” Moderator: but it would know the time of the day and it would only do it if you had done that at that time of the day. otherwise, for random times you sat down, it wouldn’t have learned anything, it wouldn’t do anything. “oh OK”
- “eventually your whole house is hooked up to one computer that acts like a server and it tells everything in your house what to do” Test User 23
“then it would make us really lazy” Test User 25
mod: it would just let us concentrate on different things
“yeah, that’s right” Test User 25
“I think it would be cool if something opened my garage door when I picked up my keys” (others agreed emphatically)
“or, if you open the racks on top of your car, then all the doors open because you most likely have a lot of stuff in your hands”
mod: or, if it senses you are carrying your skis out to the car, it would pop the racks open for you
when mod explaining some suggestions of earlier users, bnh said: “dude, it’s gonna get like that and it’s going to be weird.”
Mod: what do you think it’s like to interact with a system like this?”
“I think that it’s the future. If you don’t think so, you’re just being ignorant. Of course people are going to do this, because it makes your day easier. That’s the whole point, so we can focus more on things that we really enjoy doing and less on things that are just day to day.” Test User 23
So, you think this is a good example of…”
“I think it’s the future, yeah.” Test User 23
mod: do you prefer teaching it, or the automatic ones (first aid, and remote clicking)?
“I think it would depend on the circumstances. I think it would be better to have it learn, but if it’s an emergency, not. Because otherwise it could be annoying.”
“I think what you’ve chosen to do is right. The emergencies and the clicking are the right things.” Test User 25
Appendix D: Data Sheets for ContexTable Components
Figure D.1. FSR data sheets.

State-of-the-Art Pointing Solutions for the OEM

FSR®
Force Sensing Resistor®
Integration Guide and
Evaluation Parts Catalog

400 Series Evaluation Parts
With Suggested Electrical Interfaces

INTERLINK ELECTRONICS
546 Flynn Road - Canfield, CA 93012
(805) 484-4883 • Fax (805) 484-4889
http://www.interlinkelectronics.com

243
FSR® Integration Guide & Evaluation Parts Catalog
With Suggested Electrical Interfaces

Force Sensing Resistors® – An Overview of the Technology .............................................. Page 3
Force vs. Resistance ........................................................................................................ Page 3
Force vs. Conductance ..................................................................................................... Page 4

FSR Integration Notes – A Step-by-Step Guide to Optimal Use ........................................... Page 6

FSR Usage Tips – The Do’s and Don’ts ............................................................................ Page 8

Evaluation Parts Catalog – Descriptions and Dimensions ................................................ Page 9

General FSR Characteristics .......................................................................................... Page 12
Simple FSR Devices and Arrays ..................................................................................... Page 12
For Linear Pots ................................................................................................................ Page 13

Glossary of Terms ........................................................................................................... Page 14

Suggested Electrical Interfaces – Basic FSRs ................................................................. Page 16
FSR Voltage Divider ...................................................................................................... Page 16
Adjustable Buffers .......................................................................................................... Page 17
Multi-channel FSR to Digital Interface ........................................................................... Page 18
FSR Variable Force Threshold Switch ............................................................................ Page 19
FSR Variable Force Threshold Relay Switch ................................................................. Page 20
FSR Current-to-Voltage Converter ................................................................................. Page 21
Additional FSR Current-to-Voltage Converters ............................................................... Page 22
FSR Schmitt Trigger Oscillator ...................................................................................... Page 23

Interlink Electronics manufactures custom FSR devices to meet the needs of specific customer applications. FSR devices can be produced in almost any shape, size, and geometry.
To discuss custom design or to obtain a quote, contact Interlink Electronics at (805) 484-8855.
Force Sensing Resistors
An Overview of the Technology

Force Sensing Resistors (FSR) are a polymer thick film (PTF) device which exhibits a decrease in resistance with an increase in the force applied to the active surface. Its force sensitivity is optimized for use in human touch control of electronic devices. FSRs are not a load cell or strain gauge, though they have similar properties. FSRs are not suitable for precision measurements.

Force vs. Resistance

The force vs. resistance characteristic shown in Figure 2 provides an overview of FSR typical response behavior. For interpretational convenience, the force vs. resistance data is plotted on a log/log format. These data are representative of our typical devices, with this particular force-resistance characteristic being the response of evaluation part # 402 (0.5" [12.7 mm] diameter circular active area). A stainless steel actuator with a 0.4" [10.0 mm] diameter hemispherical tip of 60 durometer polyurethane rubber was used to actuate the FSR device. In general, FSR response approximately follows an inverse power-law characteristic (roughly 1/R).

Referring to Figure 2, at the low force end of the force-resistance characteristic, a switch-like response is evident. This turn-on threshold, or "break force", that swings the resistance from greater than 100 kΩ to about 10 kΩ (the beginning of the dynamic range that follows a power-law) is determined by the substrate and overlay thickness and flexibility, size and shape of the actuator, and spacer-adhesive thickness (the gap between the facing conductive elements). Break force increases with increasing substrate and overlay rigidity, actuator size, and spacer-adhesive thickness. Eliminating the adhesive, or keeping it well away from the area where the force is being applied, such as the center of a large FSR device, will give it a lower rest resistance (e.g. stand-off resistance).
At the high force end of the dynamic range, the response deviates from the power-law behavior, and eventually saturates to a point where increases in force yield little or no decrease in resistance. Under these conditions of Figure 2, this saturation force is beyond 10 kg. The saturation point is more a function of pressure than force. The saturation pressure of a typical FSR is on the order of 100 to 200 psi. For the data shown in Figures 2, 3 and 4, the actual measured pressure range is 0 to 175 psi (0 to 22 lbs applied over 0.125 in²). Forces higher than the saturation force can be measured by spreading the force over a greater area, the overall pressure is then kept below the saturation point, and dynamic response is maintained. However, the converse of this effect is also true, smaller actuators will saturate FSRs earlier in the dynamic range, since the saturation point is reached at a lower force.

**Force vs. Conductance**

In Figure 3, the conductance is plotted vs. force (the inverse of resistance: 1/r). This format allows interpretation on a linear scale. For reference, the corresponding resistance values are also included on the right vertical axis. A simple circuit called a current-to-voltage converter (see page 21) gives a voltage output directly proportional to FSR conductance and can be useful where response linearity is desired. Figure 3 also includes a typical part-to-part repeatability envelope. This error band determines the maximum accuracy of any general force measurement. The spread or width of the band is strongly dependent on the repeatability of any actuating and measuring system, as well as the repeatability tolerance held by Interlink Electronics during FSR production. Typically, the part-to-part repeatability tolerance held during manufacturing ranges from ±15% to ±25% of an established nominal resistance.
FSR Integration Notes  
A Step-by-Step Guide to Optimal Use

For best results, follow these seven steps when beginning any new product design, proof-of-concept, technology evaluation, or first prototype implementation:

1. Start with Reasonable Expectations (Know Your Sensor)
The FSR sensor is not a strain gauge, load cell or pressure transducer. While it can be used for dynamic measurement, only qualitative results are generally obtainable. Force accuracy ranges from approximately ±5% to ±25% depending on the consistency of the measurement and actuation system, the repeatability tolerance held in manufacturing, and the use of part calibration.

Accuracy should not be confused with resolution. The force resolution of FSR devices is better than ±0.5% of full scale force.

2. Choose the Sensor that Best Fits the Geometry of Your Application
Usually sensor size and shape are the limiting parameters in FSR integration, so any evaluation part should be chosen to fit the desired mechanical actuation system. In general, standard FSR products have a common semiconductor make-up and only by varying actuation methods (e.g. overlays and actuator areas) or electrical interfaces can different response characteristics be achieved.

3. Set-up a Repeatable and Reproducible Mechanical Actuation System
When designing the actuation mechanics, follow these guidelines to achieve the best force repeatability:

- Provide a consistent force distribution. FSR response is very sensitive to the distribution of the applied force. In general, this precludes the use of dead weights for characterization since exact duplication of the weight distribution is rarely repeatable cycle-to-cycle. A consistent weight (force) distribution is more difficult to achieve than merely obtaining a consistent total applied weight (force). As long as the distribution is the same cycle-to-cycle, then repeatability will be maintained. The use of a thin elastomer between the applied force and the FSR can help absorb error from inconsistent force distributions.

- Keep the actuator area, shape, and compliance constant. Changes in these parameters significantly alter the response characteristic of a given sensor. Any test, mock-up, or evaluation conditions should be closely matched to the final use conditions. The greater the cycle-to-cycle consistency of these parameters, the greater the device repeatability. In human interface applications where a finger is the mode of actuation, perfect control of these parameters is not generally possible. However, human force sensing is somewhat inaccurate; it is rarely sensitive enough to detect differences of less than ±50%.

- Control actuator placement. In cases where the actuator is to be smaller than the FSR active area, cycle-to-cycle consistency of actuator placement is necessary. (Caution: FSR layers are held together by an adhesive that surrounds the electrically active areas. If force is applied over an area which includes the adhesive, the resulting response characteristic will be drastically altered.) In an extreme case (e.g., a large, flat, hard actuator that bridges the bordering adhesive), the adhesive can present FSR actuation
• Keep actuation cycle time consistent. Because of the time dependence of the FSR resistance to an applied force, it is important when characterizing the sensor system to assure that increasing loads (e.g., force ramps) are applied at consistent rates (cycle-to-cycle). Likewise, static force measurements must take into account FSR mechanical setting time. This time is dependent on the mechanics of actuation and the amount of force applied and is usually on the order of seconds.

4. Use the Optimal Electronic Interface
In most product designs, the critical characteristic is Force vs. Output Voltage, which is controlled by the choice of interface electronics. A variety of interface solutions are detailed in the TechNote section of this guide. Summarized here are some suggested circuits for common FSR applications.

• For FSR Pressure or Force Switches, use the simple interfaces detailed on pages 16 and 17.

• For dynamic FSR measurements or Variable Controls, a current-to-voltage converter (see pages 18 and 19) is recommended. This circuit produces an output voltage that is inversely proportional to FSR resistance. Since the FSR resistance is roughly inversely proportional to applied force, the end result is a direct proportionality between force and voltage; in other words, this circuit gives roughly linear increases in output voltage for increases in applied force. This linearization of the response optimizes the resolution and simplifies data interpretation.

5. Develop a Nominal Voltage Curve and Error Spread
When a repeatable and reproducible system has been established, data from a group of FSR parts can be collected. Test several FSR parts in the system. Record the output voltage at various pre-selected force points throughout the range of interest. Once a family of curves is obtained, a nominal force vs. output voltage curve and the total force accuracy of the system can be determined.

6. Use Part Calibration if Greater Accuracy is Required
For applications requiring the highest obtainable force accuracy, part calibration will be necessary. Two methods can be utilized: gain and offset trimming, and curve fitting.

• Gain and offset trimming can be used as a simple method of calibration. The reference voltage and feedback resistor of the current-to-voltage converter are adjusted for each FSR to pull their responses closer to the nominal curve.

• Curve fitting is the most complete calibration method. A parametric curve fit is done for the nominal curve of a set of FSR devices, and the resultant equation is stored for future use. Fit parameters are then established for each individual FSR (or sensing element in an array) in the set. These parameters, along with the measured sensor resistance (or voltage), are inserted into the equation to obtain the force reading. If needed, temperature compensation can also be included in the equation.

7. Refine the System
Spurious results can normally be traced to sensor error or system error. If you have any questions, contact Interlink Electronics' Sales Engineers to discuss your system and final data.
FSR Usage Tips  
*The Do’s and Don’ts*

- **Do** follow the seven steps of the FSR Integration Guide.
- **Do**, if possible, use a firm, flat and smooth mounting surface.
- **Do** be careful if applying FSR devices to curved surfaces. Pre-loading of the device can occur as the two opposed layers are forced into contact by the bending tension. The device will still function, but the dynamic range may be reduced and resistance drift could occur. The degree of curvature over which an FSR can be bent is a function of the size of the active area. The smaller the active area, the less effect a given curvature will have on the FSR’s response.
- **Do** avoid air bubbles and contamination when laminating the FSR to any surface. Use only thin, uniform adhesives, such as Scotch® brand double-sided laminating adhesives. Cover the entire surface of the sensor.
- **Do** be careful of kinks or dents in active areas. They can cause false triggering of the sensors.
- **Do** protect the device from sharp objects. Use an overlay, such as a polycarbonate film or an elastomer, to prevent gouging of the FSR device.
- **Do** use soft rubber or a spring as part of the actuator in designs requiring some travel.

- **Do not** kink or crease the tail of the FSR device if you are bending it; this can cause breaks in the printed silver traces. The smallest suggested bend radius for the tails of evaluation parts is about 0.1” [2.5 mm]. In custom sensor designs, tails have been made that bend over radii of 0.03” (0.8 mm). Also, be careful if bending the tail near the active area. This can cause stress on the active area and may result in pre-loading and false readings.
- **Do not** block the vent. FSR devices typically have an air vent that runs from the open active area down the length of the tail and out to the atmosphere. This vent assures pressure equilibrium with the environment, as well as allowing even loading and unloading of the device. Blocking this vent could cause FSRs to respond to any actuation in a non-repeatable manner. Also note, that if the device is to be used in a pressure chamber, the vented end will need to be kept vented to the outside of the chamber. This allows for the measurement of the differential pressure.
- **Do not** solder directly to the exposed silver traces. With flexible substrates, the solder joint will not hold and the substrate can easily melt and distort during the soldering. Use Interlink Electronics’ standard connection techniques, such as solderable tabs, housed female contacts, Z-axis conductive tapes, or ZIF (zero insertion force) style connectors.
- **Do not** use cyanocrylate adhesives (e.g. Krazy Glue®) and solder flux removing agents. These degrade the substrate and can lead to cracking.
- **Do not** apply excessive shear force. This can cause delamination of the layers.
- **Do not** exceed 1mA of current per square centimeter of applied force (actuator area). This can irreversibly damage the device.
**Descriptions and Dimensions**

**Active Area:** 0.2" [5.0] diameter

**Nominal Thickness:** 0.012" [0.30 mm]

**Material Build:**
- **Semiconductive layer**
  0.004" [0.10] PES
- **Spacer adhesive**
  0.002" [0.05] Acrylic
- **Conductive layer**
  0.004" [0.10] PES
- **Rear adhesive**
  0.002" [0.05] Acrylic

**Connector options**
- a. No connector
- b. Solder Tabs (not shown)
- c. AMP Female connector

---

**Active Area:** 0.5" [12.7] diameter

**Nominal thickness:** 0.018" [0.46 mm]

**Material Build:**
- **Semiconductive Layer**
  0.005" [0.12] Ultem
- **Spacer Adhesive**
  0.006" [0.15] Acrylic
- **Conductive Layer**
  0.005" [0.13] Ultem
- **Rear Adhesive**
  0.002" [0.05] Acrylic
- **Connector**
  a. No connector
  b. Solder Tabs (not shown)
  c. AMP Female connector

*Dimensions in brackets: millimeters • Dimensional Tolerance: ±0.015" [0.4] • Thickness Tolerance: ± 10%*
# General FSR Characteristics

These are typical parameters. The FSR is a custom device and can be made for use outside these characteristics. Consult Sales Engineering with your specific requirements.

## Simple FSR Devices and Arrays

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>VALUE</th>
<th>NOTES</th>
</tr>
</thead>
<tbody>
<tr>
<td>Size Range</td>
<td>Max = 20&quot; x 24&quot; (51 x 61 cm) Min = 0.2&quot; x 0.2&quot; (0.5 x 0.5 cm)</td>
<td>Any shape</td>
</tr>
<tr>
<td>Device thickness</td>
<td>0.008&quot; to 0.050&quot; (0.20 to 1.25 mm)</td>
<td>Dependent on materials</td>
</tr>
<tr>
<td>Force Sensitivity Range</td>
<td>&lt; 100 g to &gt; 10 kg</td>
<td>Dependent on mechanics</td>
</tr>
<tr>
<td>Pressure Sensitivity Range</td>
<td>&lt; 1.5 psi to &gt; 150 psi (&lt; 0.1 kg/cm² to &gt; 10 kg/cm²)</td>
<td>Dependent on mechanics</td>
</tr>
<tr>
<td>Part-to-Part Force Repeatability</td>
<td>± 15% to ± 25% of established nominal resistance</td>
<td>With a repeatable actuation system</td>
</tr>
<tr>
<td>Single Part Force Repeatability</td>
<td>± 2% to ± 5% of established nominal resistance</td>
<td>With a repeatable actuation system</td>
</tr>
<tr>
<td>Force Resolution</td>
<td>Better than 0.5% full scale</td>
<td></td>
</tr>
<tr>
<td>Break Force (Turn-on Force)</td>
<td>20 g to 100 g (0.7 oz to 3.5 oz)</td>
<td>Dependent on mechanics and FSR build</td>
</tr>
<tr>
<td>Stand-Off Resistance</td>
<td>&gt; 1MΩ</td>
<td>Unloaded, unbent</td>
</tr>
<tr>
<td>Switch Characteristic</td>
<td>Essentially zero travel</td>
<td></td>
</tr>
<tr>
<td>Device Rise Time</td>
<td>1-2 msec (mechanical)</td>
<td></td>
</tr>
<tr>
<td>Lifetime</td>
<td>&gt; 10 million actuations</td>
<td></td>
</tr>
<tr>
<td>Temperature Range</td>
<td>-30°C to +70°C</td>
<td>Dependent on materials</td>
</tr>
<tr>
<td>Maximum Current</td>
<td>1 mA/cm² of applied force</td>
<td></td>
</tr>
<tr>
<td>Sensitivity to Noise/Vibration</td>
<td>Not significantly affected</td>
<td></td>
</tr>
<tr>
<td>EMI / ESD</td>
<td>Passive device</td>
<td></td>
</tr>
<tr>
<td>Lead Attachment</td>
<td>Standard flex circuit techniques</td>
<td></td>
</tr>
</tbody>
</table>
FSR Voltage Divider

For a simple force-to-voltage conversion, the FSR device is tied to a measuring resistor in a voltage divider configuration. The output is described by the equation:

\[ V_{OUT} = \frac{(V+) / [1 + RFSR/RM]} \]

In the shown configuration, the output voltage increases with increasing force. If RFSR and RM are swapped, the output swing will decrease with increasing force. These two output forms are mirror images about the line \( V_{OUT} = (V+) / 2 \).

The measuring resistor, RM, is chosen to maximize the desired force sensitivity range and to limit current. The current through the FSR should be limited to less than 1 mA/square cm of applied force. Suggested op-amps for single sided supply designs are LM358 and LM324. FET input devices such as LF355 and TL082 are also good. The low bias currents of these op-amps reduce the error due to the source impedance of the voltage divider.

A family of FORCE vs. VOUT curves is shown on the graph above for a standard FSR in a voltage divider configuration with various RM resistors. A \((V+) of +5\text{V}\) was used for these examples.
Figure D.2. Reed relay data sheets.

**GENERAL SPECIFICATIONS**

- **Coil**
  - Pull-In Voltage: 85% of nominal voltage or less
  - Drop Out Voltage: 10% of nominal voltage or more
  - Max. Voltage: 110% of nominal voltage
  - Resistance: +10% measured @ 25°C
  - Coil Power: See chart
  - Duty: Continuous

- **Contacts**
  - Contact Material: Rhodium
  - Contact Resistance: 200 megohms max
  - Contact Rating: 0.25 amp @ 100 VDC (5 VA)
  - Contact Current: 0.5 amperes max continuous carry current

- **Timing**
  - Operate-time: 1 ms or less @ nominal voltage
  - Release-time: 1 ms or less @ nominal voltage

- **Dielectric Strength**
  - Across Open Contacts: 1000 V rms
  - Between Poles: 300 V rms
  - Insulation Resistance: 1000 megohms min. @ 100 VDC
  - Capacitance: 10 pf typical coil to contact

- **Temperature**
  - Operating: -40°C to +85°C @ rated operation
  - Storage: -40°C to +105°C

- **Shock Resistance**
  - Operating: 50 gs

- **Vibration Resistance**
  - Operating: 20 gs, 40 Hz to 200 Hz

- **Life Expectancy**
  - Electrical: 50,000,000 operations @ 500mA/60mA
  - Mechanical: 60,000,000 operations low level 10V/10mA

- **Miscellaneous**
  - Operating Position: Any
  - Enclosure: Epoxy molded
  - Weight: 1 gram approx.
## DUAL IN - LINE PACKAGE REED RELAY

### SPDT, 0.25 AMP

<table>
<thead>
<tr>
<th>WIRING DIAGRAMS</th>
<th>STANDARD PART NUMBERS</th>
<th>NOMINAL INPUT VOLTAGE</th>
<th>NOMINAL RESISTANCE (OHMS)</th>
<th>NOMINAL POWER (mW)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>SPDT</strong></td>
<td>W1720IP-1</td>
<td>5</td>
<td>200 Ω</td>
<td>125</td>
</tr>
<tr>
<td></td>
<td>W1720IP-3</td>
<td>12</td>
<td>500 Ω</td>
<td>300</td>
</tr>
<tr>
<td></td>
<td>W1720IP-4</td>
<td>24</td>
<td>2200 Ω</td>
<td>270</td>
</tr>
<tr>
<td><strong>SPDT WITH CLAMPING DIODE</strong></td>
<td>W1720IP-5</td>
<td>5</td>
<td>200 Ω</td>
<td>125</td>
</tr>
<tr>
<td></td>
<td>W1720IP-7</td>
<td>12</td>
<td>500 Ω</td>
<td>300</td>
</tr>
<tr>
<td></td>
<td>W1720IP-8</td>
<td>24</td>
<td>2200 Ω</td>
<td>270</td>
</tr>
<tr>
<td><strong>SPDT</strong></td>
<td>W1720IP-31</td>
<td>5</td>
<td>200 Ω</td>
<td>125</td>
</tr>
<tr>
<td></td>
<td>W1720IP-33</td>
<td>12</td>
<td>500 Ω</td>
<td>290</td>
</tr>
<tr>
<td></td>
<td>W1720IP-34</td>
<td>24</td>
<td>2200 Ω</td>
<td>270</td>
</tr>
<tr>
<td><strong>SPDT WITH CLAMPING DIODE</strong></td>
<td>W1720IP-35</td>
<td>5</td>
<td>200 Ω</td>
<td>125</td>
</tr>
<tr>
<td></td>
<td>W1720IP-37</td>
<td>12</td>
<td>500 Ω</td>
<td>200</td>
</tr>
<tr>
<td></td>
<td>W1720IP-38</td>
<td>24</td>
<td>2200 Ω</td>
<td>270</td>
</tr>
<tr>
<td><strong>SPDT</strong></td>
<td>W1720IP-141</td>
<td>5</td>
<td>200 Ω</td>
<td>125</td>
</tr>
<tr>
<td></td>
<td>W1720IP-145</td>
<td>12</td>
<td>1000 Ω</td>
<td>144</td>
</tr>
<tr>
<td></td>
<td>W1720IP-146</td>
<td>24</td>
<td>3200 Ω</td>
<td>180</td>
</tr>
<tr>
<td><strong>SPDT WITH CLAMPING DIODE</strong></td>
<td>W1720IP-147</td>
<td>5</td>
<td>200 Ω</td>
<td>125</td>
</tr>
<tr>
<td></td>
<td>W1720IP-149</td>
<td>12</td>
<td>1000 Ω</td>
<td>144</td>
</tr>
<tr>
<td></td>
<td>W1720IP-150</td>
<td>24</td>
<td>3200 Ω</td>
<td>180</td>
</tr>
</tbody>
</table>

SEE END OF SECTION 5 FOR CROSS REFERENCE
Figure D.3. Op amp data sheets.

Ultralow Offset Voltage Operational Amplifier

**OP07**

**FEATURES**
- Low \( V_{\text{os}} \): 75 \( \mu \)V Max
- Low \( V_{\text{os}} \) Drift: 1.3 \( \mu \)V/°C Max
- Ultrastable vs. Time: 1.5 \( \mu \)V/Month Max
- Low Noise: 0.6 \( \mu \)V p-p Max
- Wide Input Voltage Range: ±14 V
- Wide Supply Voltage Range: 3 V to 18 V
- Fits 725, 108A/308A, 741, AD510 Sockets
- 125°C Temperature-Tested Device

**APPLICATIONS**
- Wireless Base Station Control Circuits
- Optical Network Control Circuits
- Instrumentation
- Sensors and Controls
- Thermocouples
- RTDs
- Strain Bridges
- Shunt Current Measurements
- Precision Filters

**GENERAL DESCRIPTION**
The OP07 has very low input offset voltage (75 \( \mu \)V max for OP07E) that is obtained by trimming at the wafer stage. These low offset voltages generally eliminate any need for external nulling. The OP07 also features low input bias current (±4 nA for the OP07E) and high open-loop gain (200 V/mV for the OP07E). The low offsets and high open-loop gain make the OP07 particularly useful for high gain instrumentation applications.

The wide input voltage range of ±13 V minimum combined with a high CMRR of 106 dB (OP07E) and high input impedance provide high accuracy in the noninverting circuit configuration. Excellent linearity and gain accuracy can be maintained even at high closed-loop gains. Stability of offsets and gain with time or variations in temperature is excellent. The accuracy and stability of the OP07, even at high gain, combined with the freedom from external nulling have made the OP07 an industry standard for instrumentation applications.

The OP07 is available in two standard performance grades. The OP07E is specified for operation over the 0°C to 70°C range, and the OP07C is specified over the −40°C to +85°C temperature range.

The OP07 is available in epoxy 8-lead PDIP and 8-lead SOIC. It is a direct replacement for 725, 108A, and OP05 amplifiers; 741 types may be directly replaced by removing the 741’s nulling potentiometer. For improved specifications, see the OP117 or OP1177. For ceramic DIP and TO-99 packages and standard micro circuit (SMD) versions, see the OP77.

**PIN CONNECTIONS**
- 8-Lead PDIP (P-Suffix)
- 8-Lead SOIC (S-Suffix)

---

**Figure 1. Simplified Schematic**

Information furnished by Analog Devices is believed to be accurate and reliable. However, no responsibility is assumed by Analog Devices for its use, nor for any infringements of patents or other rights of third parties that may result from its use. No license is granted by implication or otherwise under any patent or patent rights of Analog Devices. Trademarks and registered trademarks are the property of their respective companies.
# OP07—SPECIFICATIONS

**OP07E ELECTRICAL CHARACTERISTICS** *(V<sub>CC</sub> = ±15 V, T<sub>A</sub> = 25°C, unless otherwise noted.)*

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Symbol</th>
<th>Conditions</th>
<th>Min</th>
<th>Typ</th>
<th>Max</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>INPUT CHARACTERISTICS</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Input Offset Voltage&lt;sup&gt;1&lt;/sup&gt;</td>
<td>V&lt;sub&gt;OS&lt;/sub&gt;</td>
<td></td>
<td>30</td>
<td>75</td>
<td>μV</td>
<td></td>
</tr>
<tr>
<td>Long-Term V&lt;sub&gt;OS&lt;/sub&gt; Stability&lt;sup&gt;2&lt;/sup&gt;</td>
<td>V&lt;sub&gt;OS&lt;/sub&gt;/Time</td>
<td></td>
<td>0.3</td>
<td>1.5</td>
<td>μV/Mo</td>
<td></td>
</tr>
<tr>
<td>Input Offset Current</td>
<td>I&lt;sub&gt;OS&lt;/sub&gt;</td>
<td></td>
<td>0.5</td>
<td>3.8</td>
<td>nA</td>
<td></td>
</tr>
<tr>
<td>Input Bias Current</td>
<td>I&lt;sub&gt;B&lt;/sub&gt;</td>
<td></td>
<td>±1.2</td>
<td>±4.0</td>
<td>nA</td>
<td></td>
</tr>
<tr>
<td>Input Noise Voltage</td>
<td>e&lt;sub&gt;p-p&lt;/sub&gt;</td>
<td>0.1 Hz to 10 Hz&lt;sup&gt;3&lt;/sup&gt;</td>
<td>0.35</td>
<td>0.6</td>
<td>μV/p-p</td>
<td></td>
</tr>
<tr>
<td>Input Noise Voltage Density</td>
<td>e&lt;sub&gt;s&lt;/sub&gt;</td>
<td>f&lt;sub&gt;S&lt;/sub&gt; = 10 Hz</td>
<td>10.3</td>
<td>18.0</td>
<td>nV/Hz&lt;sup&gt;1/2&lt;/sup&gt;</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>f&lt;sub&gt;S&lt;/sub&gt; = 100 Hz&lt;sup&gt;3&lt;/sup&gt;</td>
<td>10.0</td>
<td>13.0</td>
<td>nV/Hz&lt;sup&gt;1/2&lt;/sup&gt;</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>f&lt;sub&gt;S&lt;/sub&gt; = 1 kHz</td>
<td>9.6</td>
<td>11.0</td>
<td>nV/Hz&lt;sup&gt;1/2&lt;/sup&gt;</td>
<td></td>
</tr>
<tr>
<td>Input Noise Current</td>
<td>I&lt;sub&gt;p-p&lt;/sub&gt;</td>
<td>f&lt;sub&gt;S&lt;/sub&gt; = 10 Hz</td>
<td>14</td>
<td>30</td>
<td>pA/p-p</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>f&lt;sub&gt;S&lt;/sub&gt; = 100 Hz&lt;sup&gt;3&lt;/sup&gt;</td>
<td>0.32</td>
<td>0.80</td>
<td>pA/p-p</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>f&lt;sub&gt;S&lt;/sub&gt; = 1 kHz</td>
<td>0.14</td>
<td>0.23</td>
<td>pA/p-p</td>
<td></td>
</tr>
<tr>
<td><strong>INPUT RESISTANCE</strong></td>
<td>R&lt;sub&gt;∞&lt;/sub&gt;</td>
<td></td>
<td>15</td>
<td>50</td>
<td>MΩ</td>
<td></td>
</tr>
<tr>
<td><strong>INPUT RESISTANCE—Common-Mode</strong></td>
<td>R&lt;sub&gt;INCM&lt;/sub&gt;</td>
<td></td>
<td>160</td>
<td></td>
<td>GΩ</td>
<td></td>
</tr>
<tr>
<td><strong>Input Voltage Range</strong></td>
<td>FVR</td>
<td></td>
<td>±13</td>
<td>±14</td>
<td>V</td>
<td></td>
</tr>
<tr>
<td><strong>Common-Mode Rejection Ratio</strong></td>
<td>CMRR</td>
<td>V&lt;sub&gt;CM&lt;/sub&gt; = ±13 V</td>
<td>106</td>
<td>123</td>
<td>dB</td>
<td></td>
</tr>
<tr>
<td><strong>Power Supply Rejection Ratio</strong></td>
<td>PSRR</td>
<td>V&lt;sub&gt;S&lt;/sub&gt; = ±3 V to ±18 V</td>
<td>5</td>
<td>20</td>
<td>μV/V</td>
<td></td>
</tr>
<tr>
<td><strong>Large Signal Voltage Gain</strong></td>
<td>AV&lt;sub&gt;OL&lt;/sub&gt;</td>
<td>R&lt;sub&gt;g&lt;/sub&gt; = 2 kΩ, V&lt;sub&gt;OL&lt;/sub&gt; = ±10 V</td>
<td>200</td>
<td>500</td>
<td>V/mV</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>R&lt;sub&gt;g&lt;/sub&gt; = 500 Ω, V&lt;sub&gt;OL&lt;/sub&gt; = ±0.5 V,</td>
<td>150</td>
<td>400</td>
<td>V/mV</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Symbol</th>
<th>Conditions</th>
<th>Min</th>
<th>Typ</th>
<th>Max</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>OUTPUT CHARACTERISTICS</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Output Voltage Swing</td>
<td>V&lt;sub&gt;o&lt;/sub&gt;</td>
<td>R&lt;sub&gt;g&lt;/sub&gt; = 10 kΩ</td>
<td>±12.5</td>
<td>±13.0</td>
<td>V</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>R&lt;sub&gt;g&lt;/sub&gt; = 2 kΩ</td>
<td>±12.0</td>
<td>±12.8</td>
<td>V</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>R&lt;sub&gt;g&lt;/sub&gt; = 1 kΩ</td>
<td>±10.5</td>
<td>±12.0</td>
<td>V</td>
<td></td>
</tr>
</tbody>
</table>

**DYNAMIC PERFORMANCE**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Symbol</th>
<th>Conditions</th>
<th>Min</th>
<th>Typ</th>
<th>Max</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Slope Rate</td>
<td>SR</td>
<td>R&lt;sub&gt;g&lt;/sub&gt; = 2 kΩ&lt;sup&gt;1&lt;/sup&gt;</td>
<td>0.1</td>
<td>0.3</td>
<td>V/μs</td>
<td></td>
</tr>
<tr>
<td>Closed-Loop Bandwidth</td>
<td>BW</td>
<td>A&lt;sub&gt;VO&lt;/sub&gt; = 1&lt;sup&gt;2&lt;/sup&gt;</td>
<td>0.4</td>
<td>0.6</td>
<td>MHz</td>
<td></td>
</tr>
<tr>
<td>Closed-Loop Output Resistance</td>
<td>RO</td>
<td>V&lt;sub&gt;g&lt;/sub&gt; = 0, I&lt;sub&gt;g&lt;/sub&gt; = 0</td>
<td>60</td>
<td></td>
<td>Ω</td>
<td></td>
</tr>
<tr>
<td>Power Consumption</td>
<td>PD</td>
<td>V&lt;sub&gt;g&lt;/sub&gt; = ±15 V, No Load</td>
<td>75</td>
<td>120</td>
<td>mW</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>V&lt;sub&gt;g&lt;/sub&gt; = ±3 V, No Load</td>
<td>4</td>
<td>6</td>
<td>mW</td>
<td></td>
</tr>
<tr>
<td>Offset Adjustment Range</td>
<td>RO</td>
<td>R&lt;sub&gt;g&lt;/sub&gt; = 20 kΩ</td>
<td>±4</td>
<td></td>
<td>mV</td>
<td></td>
</tr>
</tbody>
</table>

**NOTES**

<sup>1</sup>Input offset voltage measurements are performed by automated test equipment approximately 0.5 seconds after application of power.
<sup>2</sup>Long-term input offset voltage stability refers to the averaged trend of V<sub>OS</sub> vs. the time over extended periods after the first 30 days of operation. Including the initial 30 days of operation, changes in V<sub>OS</sub> during the first 30 operating days are typically 2.5 μV, refer to the typical performance characteristics. Parameter is sample tested.
<sup>3</sup>Sample tested.
<sup>4</sup>Guaranteed by design.

Specifications subject to change without notice.
Figure 2. Typical Offset Voltage Test Circuit

Figure 3. Typical Low Frequency Noise Circuit

Figure 4. Optional Offset Nulling Circuit

Figure 5. Burn-In Circuit

Figure 6. High Speed, Low $V_{os}$ Composite Amplifier

Figure 7. Adjustment-Free Precision Summing Amplifier
Figure D.4. Photoresistor data sheet.

### Plastic-coated Types (7P 10p Types)

<table>
<thead>
<tr>
<th>Type No.</th>
<th>Outline</th>
<th>Applied Voltage at 25°C (Vdc)</th>
<th>Allowable Power Dissipation at 25°C (W)</th>
<th>Ambient Temperature Ts (°C)</th>
<th>Cell Resistance A (Ω)</th>
<th>Rise Time at 10 kΩ (ms)</th>
<th>Decay Time at 10 kΩ (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>JT01</td>
<td>200</td>
<td>150</td>
<td>-30 to +75</td>
<td>3.6</td>
<td>6.4</td>
<td>50</td>
<td>20</td>
</tr>
<tr>
<td>JT02</td>
<td>200</td>
<td>150</td>
<td>-30 to +75</td>
<td>4</td>
<td>20</td>
<td>55</td>
<td>20</td>
</tr>
<tr>
<td>JT03</td>
<td>200</td>
<td>150</td>
<td>-30 to +75</td>
<td>8</td>
<td>15</td>
<td>55</td>
<td>20</td>
</tr>
<tr>
<td>JT04</td>
<td>200</td>
<td>150</td>
<td>-30 to +75</td>
<td>15</td>
<td>60</td>
<td>60</td>
<td>25</td>
</tr>
<tr>
<td>JT05</td>
<td>200</td>
<td>150</td>
<td>-30 to +75</td>
<td>50</td>
<td>150</td>
<td>60</td>
<td>25</td>
</tr>
<tr>
<td>JT06</td>
<td>350</td>
<td>400</td>
<td>-30 to +75</td>
<td>8</td>
<td>16</td>
<td>55</td>
<td>25</td>
</tr>
<tr>
<td>JT07</td>
<td>350</td>
<td>400</td>
<td>-30 to +75</td>
<td>12</td>
<td>30</td>
<td>55</td>
<td>25</td>
</tr>
<tr>
<td>JT08</td>
<td>350</td>
<td>400</td>
<td>-30 to +75</td>
<td>12</td>
<td>58</td>
<td>55</td>
<td>25</td>
</tr>
</tbody>
</table>

- **Cell resistance vs. illuminance**
  - A. Measured with the light source of a tungsten lamp operated at a color temperature of 2856K.
  - B. Measured 10 seconds after removal of incident illuminance of 10 lux.
  - C. Gamma characteristic between 10 lux and 100 lux and given by
    \[
    \log(E) = \log(R(10)) + \log(E(10))
    \]
    Where: \(R(100), R(10)\): cell resistances at 100 lux and 10 lux, respectively, \(E(100), E(10)\): illuminances of 100 lux and 10 lux respectively.
  - D. The rise time is the time required for the cell conductance to rise to 63% of the saturated level. The decay time is the time required for the cell conductance to decay from the saturated level to 37%.

- **Cell resistance vs. temperature**

- **Out-line Dimension**
Appendix E: All Questionnaire Answers

<table>
<thead>
<tr>
<th>Test User 2 (Questionnaire 2, Test Style 1)</th>
<th>IT master's</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 other things?</td>
<td>system for children.</td>
</tr>
<tr>
<td>2 annoyance?</td>
<td>8</td>
</tr>
<tr>
<td>3 good services?</td>
<td>6, bugs were being worked out</td>
</tr>
<tr>
<td>4 bog down?</td>
<td>(2nd person to use it)</td>
</tr>
<tr>
<td>5 natural?</td>
<td>10</td>
</tr>
<tr>
<td>6 surprises?</td>
<td>the alarm voice activation system</td>
</tr>
<tr>
<td>7 other things?</td>
<td>voice activation</td>
</tr>
<tr>
<td>8 Follow a routine?</td>
<td>5</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>R.M. (Q 1, T 1)</th>
<th>IT professional</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 helpful?</td>
<td>yes, it anticipated my habits</td>
</tr>
<tr>
<td>2 surprise?</td>
<td>not much</td>
</tr>
<tr>
<td>3 annoying?</td>
<td>yes, same goofy voice each time</td>
</tr>
<tr>
<td>4 follow habits?</td>
<td>part of the time</td>
</tr>
<tr>
<td>5 would it be?</td>
<td>I think it would be fun for a while and then become a pain</td>
</tr>
<tr>
<td>6 other things?</td>
<td>car (truck) and bed</td>
</tr>
<tr>
<td>7 anything else?</td>
<td>no</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Test User 5 (Q 1, T 1)</th>
<th>12/18/2003 ME undergrad</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 helpful?</td>
<td>yes</td>
</tr>
<tr>
<td>2 surprise?</td>
<td>how it learned &amp; got faster it could be annoying that loud noises cause it to ask if you need help… that could be programmed to listen for “HELP”</td>
</tr>
<tr>
<td>3 annoying?</td>
<td>for sure in the morning – probably not in the afternoon &amp; maybe in the evening</td>
</tr>
<tr>
<td>4 follow habits?</td>
<td>it’d be fun… if money weren't an issue I'd get one</td>
</tr>
<tr>
<td>5 would it be?</td>
<td>heated bed - smart office desk</td>
</tr>
<tr>
<td>6 other things?</td>
<td>open garage door when it senses your car drive up</td>
</tr>
<tr>
<td>7 anything else?</td>
<td></td>
</tr>
</tbody>
</table>

259
Test User 6 (Q 1 T 1)
1 helpful?
2 surprise?
3 annoying?
4 follow habits?
5 would it be?
6 other things?
7 anything else?

Law grad
12/19/2003 student

911 to get help, breakfast because I do everything the same
slow, better voice, ability for user to just say "yes"
yes, but sometimes habits get thrown off like during school, for example
if it were taken to the right level, ex.: for little kids to help moms
if it was user-friendly enough, it can't be too complicated, it has to be simple

Notes:
not using voice to confirm all actions (it is useful for all things, not just slow ones, people like that it confirms what they see)
open up widows in front

Test User 3 (Q 2 T 1)
1 other things?
2 annoyance?
3 good services?
4 bog down?
5 natural?
6 surprises?
7 other things?

IT undergrad

that sort of integration would be fantastic, pending that there is a "kill all" sort of option when you don't need a car to start sort of thing. Lights on and off also good, TV/VCR recording.
5 (annoyance: not used to it so annoying)

Test User 7 (Q 1.5 T 2)
1 helpful?
2 obvious?
3 surprised?
4 follow habits?
5 helpful in daily?

IT graduate

yes, remote control
yes
channel clicker, recognizing things
yeah, like email in the morning, but would need to embed
sensors into more objects
yes
6 would you want it? I break my habits
7 other things? lights would be very cool
8 anything else? did a good job for alphath testing

Test User 9 (Q 1.5, T 2) ME undergrad
for certain applications. Pretty basic, though. With lots of robotics it could set the table
1 helpful? for you
2 obvious? yeah
3 surprised? nothing
4 follow habits? yes, consistent enough not a critically needed thing, but I guess so. It would be helpful in daily?
5 helpful in daily? convenient
6 would you want it? yes, it would be
TV - to remember favorite channels. Reminders to pay bills
7 other things? n/a
8 anything else? n/a

Test User 8 (Q 1.5, T 2) IT undergrad
it was a neat concept. Could be useful, especially on "rut" mornings [moderator comment: when you're not motivated, it reminds you to get on to the next thing]
1 helpful? next thing]
2 obvious? yes
how little I pay attention to my habits [mod. Comment: user did not perform "sensible" actions, was very free-form and cues were not "suitable" to given action.
3 surprised? I have several different sets of habits depending on mood and day of the week, but I'm sure it [software] could be easily modified.
4 follow habits? modified.
5 helpful in daily? definitely
6 would you want it? definitely
7 other things? not right off
8 anything else? n/a

Test User 10 (Q 1.5, T 2) IT undergrad
1 helpful? yes
2 obvious? yes
3 surprised? the variety of actions it would
learn

4 follow habits? definitely
5 helpful in daily? yes
6 would you want it? yes
7 other things? anything that is a repeated action
8 anything else? great project

Test User 11 (Q 2 & Q 1.5, T 2)
yes, a bed that reminds me to pray, for example, would be great.

1 other things?
2 annoyance? 
3 good services? as long as it's easy to turn off functions that do get annoying it might take some time, but it would get easier each time through the computer learned more - it was interesting to see each time what the computer could do
4 bog down? Breakfast cereal that pours itself? How about a crib that changes diapers for you?
5 natural? 
6 surprises? For many routines yes; for example my morning routine is pretty consistent, and I'd always want cupboards to open if I had groceries. I'd want to be able to keep it from always prompting me to do things for other less consistent routines, though.
7 other things? Breakfast cereal that pours itself? How about a crib that changes diapers for you?
8 anything else? n/a
Test User 12 (Q 2, T 2)   Housewife

1 other things?
2 annoyance? 8

...sure, the alarm setting is great -
maybe something built into
laundry systems.

3 good services? 8
4 bog down? 8
5 natural? 8

...yes, however, I
don't think I
could make
routine to an
exact time of day
things like
making a phone
call [explain
about learning
different cues for
different times
and that phone
might not be
implemented the
best was right

6 surprises? 8
7 other things?
Follow routine
8 enuf?

...subtle things like having to sit
down as I did the first time -
how to set the routine
laundry system - maybe a
sensor at the top of a laundry
basket to let you know it's time

to wash clothes
[scale of] within 3 hours is

Test User 1 (Q 2, video
interview, T 1)   IT undergrad

1 other things?
2 annoyance? 7

...yes, it would be helpful, for cell
phone especially

3 good services? 8
4 bog down? 9
5 natural? 8

...things were
pretty self-

6 surprises? 8
7 other things? 9
Follow routine
8 enuf?

...the morning breakfast routine
was nice. I have a routine, and
this system would be perfect
for it.

showering, watching TV
<table>
<thead>
<tr>
<th>1 other things?</th>
<th>I can see how this system could help daydreamers, keep on task… reminder</th>
</tr>
</thead>
<tbody>
<tr>
<td>2 annoyance?</td>
<td>8 neat reminder/prompts for daydreamers experience shows that one idea begets another. It could simplify one thing, thus opening up time for another.</td>
</tr>
<tr>
<td>3 good services?</td>
<td>8 reminder/prompts for daydreamers experience shows that one idea begets another. It could simplify one thing, thus opening up time for another.</td>
</tr>
<tr>
<td>4 bog down?</td>
<td>n/a [reminder to buy things you usually buy]</td>
</tr>
<tr>
<td>5 natural?</td>
<td>n/a</td>
</tr>
<tr>
<td>6 surprises?</td>
<td>yes, what if I wanted to do something different one day and it automatically opened the program (can you change it?)</td>
</tr>
<tr>
<td>7 other things?</td>
<td>Follow routine</td>
</tr>
<tr>
<td>8 enuf?</td>
<td>5.5</td>
</tr>
</tbody>
</table>

---

<table>
<thead>
<tr>
<th>1 other things?</th>
<th>yes, once the system knows your schedule it would be helpful to take some annoying tasks off my hands</th>
</tr>
</thead>
<tbody>
<tr>
<td>2 annoyance?</td>
<td>when I wanted to play the game it just gave me the dice. Others were nice as well like the news -</td>
</tr>
<tr>
<td>3 good services?</td>
<td>9 automatic</td>
</tr>
<tr>
<td>4 bog down?</td>
<td>6</td>
</tr>
<tr>
<td>5 natural?</td>
<td>8</td>
</tr>
<tr>
<td>6 surprises?</td>
<td>yes, what if I wanted to do something different one day and it automatically opened the program (can you change it?)</td>
</tr>
<tr>
<td>7 other things?</td>
<td>taking a shower and playing music - turning on a fan, etc.</td>
</tr>
<tr>
<td>Follow routine</td>
<td></td>
</tr>
<tr>
<td>8 enuf?</td>
<td></td>
</tr>
<tr>
<td>Test User 17 (Q 2, T 2)</td>
<td>BLIND!</td>
</tr>
<tr>
<td>undergrad applicant, communications</td>
<td></td>
</tr>
<tr>
<td>1 other things?</td>
<td>most definitely, it would be very helpful to someone like me, who cannot see</td>
</tr>
<tr>
<td>2 annoyance?</td>
<td></td>
</tr>
<tr>
<td>3 good services?</td>
<td></td>
</tr>
<tr>
<td>4 bog down?</td>
<td></td>
</tr>
<tr>
<td>5 natural?</td>
<td></td>
</tr>
<tr>
<td>6 surprises?</td>
<td></td>
</tr>
<tr>
<td>7 other things?</td>
<td></td>
</tr>
<tr>
<td>Follow routine</td>
<td></td>
</tr>
<tr>
<td>8 enuf?</td>
<td></td>
</tr>
</tbody>
</table>

Notes:
Help for non-sighted people: picking out colors of clothing. ["You just chose a red bottom and a green top. These colors don't go together. Are you sure you want to wear them." Or "Last time you chose green…"]

| Test User 16 (Q 2, T 2) | with blind girl Housewife |
| 1 other things? | it think it would be neat, but scare me |
| 2 annoyance? | |
| 3 good services? | |
| 4 bog down? | |
| 5 natural? | |
| 6 surprises? | |
| 7 other things? | |

265
need it done.

Follow routine
8 enuf? 5

M.O (Q 2, T 2)  
yes, it was very useful not having to look for first aid info, especially if you are distraught or hurt. I really like the car idea. I prefer an alarm to go off at a certain time, maybe even an alarm that doesn't go off unless you are in bed.

ME undergrad

1 other things?
2 annoyance? 9
when it learned my routine I didn't have to make choices, I only had to

3 good services? 9 choose yes or no.
4 bog down? 6
5 natural? 9
how well it worked. It seemed extremely useful. I would look into buying one. I expected plenty of bugs. I was pleasantly surprised at how well it worked.

Test User 18 (Q 2, T 2)  
It would be nice because it would begin to recognize everything that you would

IT undergrad

6 surprises?
Washing dishes, warming the oven or stove for cooking, Heated floors, lights. Almost no limit to the possibilities. [only limited by whether or not the system has a means of recording the action and at least some of the cues.

7 other things?
Follow routine
8 enuf? 9

Notes:
My wife does a lot of things daily, the little things really start adding up (time-wise) and this could help. One could save money by having the water heater turn on or off using this, or the dishwasher. Would be good for daily tasks. If it were also voice compatible (choose by voice, maybe even say "I want to..."), so you don't have to go to the computer, it would be even more useful.

Test User 18 (Q 2, T 2)  
It would be nice because it would begin to recognize everything that you would want. An automated house
An interesting idea. It did what I wanted when I wanted. Other tasks could be more helpful, but they weren't options yet.

Once it learned my habits, I was interested to see it do my preferences automatically. Cars, certain aspects of the house.

The ability of it multitasking my morning.

The resources were useful and helpful.

Checking my mail.

[but he said his morning schedule is very repetitive]

yes, there are many examples of routine actions that would be nice (Reminder to change cell phone at night, even meeting deadlines) setting the alarm clock according to a schedule/calendar.)
the prompts were relevant to the actions and came soon enough to show items being done. (after a couple of repeats, not 5 or 6)

3 good services? 10
4 bog down? 9
5 natural? 9
I wanted to try different options or combinations though I understand this would take longer to learn.
6 Surprises?
Refrigerator that could track usage of contents, TV that could suggest possible shows at that time.
7 Other things?
Follow routine enough? 6

Test User 21 (Q 2, T 2) 1/12/2004 IT undergrad
These are great ideas! I think as long as I can still easily override the system if I want, it would be great to have it usually do my repeated tasks.
1 Other things? as if it weren't a prototype
2 Annoyance? 10
They were a great help in making my day go smoother & faster. As long as it knows that same days are different & can
3 Good services? 9
learn many tasks.
4 Bog down? 10
I wonder if I would answer the same once something like this was put in
5 Natural? 10
my home?
Yes, it would have been neat to do different tasks & see how it learns all of them & determines what I'm doing.
6 Surprises?
Mom or dad sits down & all other programs automatically start for them. It would be cool if it could automatically set the table & clean up afterwards. Integrate it with inventory of grocery needs.

7 Other things?
   Follow routine
8 enough?