Designing collaboration in consumer products

Elyon DeKoven and David V. Keyson

Department of Industrial Design
Delft University of Technology
Jaffalaan 9, 2628 BX, Delft
The Netherlands
e.dekoven@io.tudelft.nl d.keyson@io.tudelft.nl

Abstract

Designers of consumer products usually try to address as wide a range of user needs as possible. Due to various design constraints, such as product size and a limited interface mechanism, only a portion of those needs can typically be supported. Furthermore, products are generally designed only to assist the user in terms of the product s features, or based on the last action the user performed with the product. This paper considers efforts to support the design of everyday consumer products which can collaborate with the user in terms of meeting the user s task goals. Additionally, some of the implications of embedding such capabilities in a consumer product are discussed.

Introduction

Designers trained in user-centered design methodologies typically begin their work by determining the needs of a wide range of users. This analysis is then translated into a single solution designed to accommodate as large a range of users needs as possible. However, an individual user may have goals not explicitly supported by a best-fit . In such a case, the users will either stop using the product, or may expend a lot of effort on determining how to apply the product s features towards meeting this specific need. While most products are able to explain how a particular feature works (perhaps in the manual), the product usually cannot help the user determine how to apply any given feature to achieve the goal, nor even which feature is best to use. In fact, it cannot even identify the user s goals, and the user has no way of getting them across to the product.

There appear to be a number of advantages to designing products that can explicitly support the user s goals:

- Multiple ways for achieving a certain goal can be programmed into or determined by the product. In this manner, a wider range of user approaches to accomplishing tasks can be supported.
- It might be easier for the designer to accommodate newly discovered user goals and usage strategies by updating the task models, than it would be to change the design.
- The product can possibly assist the user in determining the best way to use its features in order to reach the user s goal.

For example, consider a combination microwave-convection oven. In order for the user to describe a recipe to the oven, say to prepare a chicken, a lot of interaction would be required, such as to set the amount of time to defrost, to bake, and to grill it. Current sensor technology can augment this interaction by automatically measuring physical factors, such as how much the chicken weighs, and how cold it currently is, but sensors alone cannot determine how well done the user would like it. In contrast, a goal-based combination oven might, for example, recognize the need to take the chicken from frozen to cooked, or the user might be able to state explicitly the goal of having the chicken grilled. The oven can then suggest to the user a number of different approaches, or ask about specific preferences (such as how crispy to make the skin). The user can then react to the individual steps, or the whole plan, or particular parameters, without worrying about how the oven does it.

To achieve a more goal-directed level of interaction requires the ability to communicate at various levels, from product features to user goals. This dialogue is necessary in order to know what the user s goals are, and how best to achieve them according to the user s preferences and constraints. This requires two-way, give-and-take communication in order to reach a shared plan towards achieving the user s goal. In other words, this requires user-product collaboration.

A product which can collaborate with its user might require a large range of internal models and databases of information. Additionally, in order to interact with the user at the level of goals, new interaction metaphors and capabilities might be necessary. This includes:

- *Task models* of the product s general application domain and of the usage of that specific product, as well as databases of information pertaining to the tasks (e.g. various cooking methods such as hot air, microwave or electric grill, and combinations of these, as well as specific recipes for different types of food);
- *Models of the user*, including the user s preferences and a history of the user s previous experiences with the product (e.g. user interactions with the oven to make this particular meal, as well as a complete history of recipes and cooking experiences the user has tried and the user s subjective reactions to the results);

- *Dialogue rules*, such as how to handle communication failures and other forms of feedback, and conversational rules for determining when the product or user should take the initiative (e.g. whether the user or the oven should have the final say in how long the food should cook);
- *Interaction capabilities*, such as supporting the user in expressing both low-level preference and higher level goals, as well as a way for the product to reflect back to the user what it understands, what it is thinking about, and what it intends to do.

Within the past decade, there have been a number of explorations into various aspects of human-product communication. Some efforts have focused on improving the product s expressive capabilities (see for example the various research efforts on multimodal agents (Cassell et al. 2000)). Other studies have sought to develop an understanding of the user s desire and ability to work with a product (for example, see the work on products as teammates (Nass, Fogg, and Moon 1996)). Additionally, technology has been developed to support the human-product dialogue in general (for example, the Collagen dialogue manager (Rich and Sidner 1998)). Together these research efforts shed much needed light on designing and building a likable product that can communicate about user preferences and goals. However, these studies have been focused on design for desktop computers. It remains to be seen to what degree these findings can be transferred into everyday consumer products. Issues which need to be addressed include:

- What is the degree to which the product s ability to collaborate is dependent upon the type of product, the type of user, the task and the context in which the product is being used?
- How will users perceptions and expectations of goal-based products be affected by their previous experiences with similar products which do not have a collaborative capability?
- What are the best ways to manage and support the human-product dialogue given the physical constraints and usability requirements of consumer products, and how will this management and support affect the dialogue itself?

Current Efforts

In order to create a collaborative product the designer needs to integrate all this knowledge. This might require new or expanded design methodologies, tools to facilitate design and prototyping, and measures for the evaluation of the various aspects of the communication. The goal of the current research is to provide researchers and designers with a platform to develop these tools, technologies and techniques. This platform is being developed through iterative design, implementation and analysis of working prototypes of collaborative consumer products.

In particular, we are focusing on ways to support and evaluate goal-based human-product dialogue. For example, this dialogue can be supported visually by presenting to the user the dialogue history, or a list of suggested things to say (see (Rich and Sidner 1998) for examples of such interfaces). They can be evaluated based on linguistic models of collaborative dialogue (Grosz and Kraus 1993) or general dialogue principles (see Grice s maxims of collaborative dialogue (Grice 1989)), as well as by traditional usability measures. Overall, the human-product interaction can be designed from the beginning to support a goal-based dialogue.

As with other product design methodologies, the design of the dialogue and graphical interface should be based on empirical observations and task analyses. However, conversational consumer products are currently not readily available. Instead, we have designed a product from the ground up with a task-level ability to communicate. Part of the objective here is to obtain first-hand experience towards identifying a successful design methodology and specific needs for prototyping tools. Applying this to other products, different types of users and different usage contexts will then provide for generalizable observations about the processes and the results of designing collaborative products.

In order to facilitate the current research, a product was needed with which people are familiar, but which does not currently support task-level interaction. Based on an analysis of the user, the task and the domain, a redesign of the graphical interface could be conducted. At the same time, a task-based agent to support the user with the new visual design could be designed and implemented.

Two of the most prominent features of modern thermostats are advanced temperature programming, and the ability to optimize the energy used. For example, some of them can store the results of the heating performed, and use these to calculate optimal settings for saving energy (such as how far in advance to turn on or off the heating). Most programmable thermostats regulate the temperature in separate rooms or zones, at different times of the day or week, or in terms of activities (such as Wake , Leave , Return , and Sleep in Honeywell s T8000C Programmable Thermostat).

While the naming of the programs does approach a user-centric vision of home heating, the current design of such features is generally not flexible enough to support user-specific tasks. The user is offered an array of buttons hard-coded to specific functions (note that Honeywell s collection of what they call lifestyles are not modifiable by the user). Furthermore, current programmable thermostats are not able to present and discuss different heating options with the user. Instead, the thermostat makes hidden calculations to determine optimizations, using predetermined heuristics of human comfort. Throughout this interaction, the user and the product have limited dialogue about the user s heating goals.

Our approach to designing an intelligent thermostat (Keyson et al. 2000) has been two-fold. The initial focus was on visual design of the thermostat to support flexible temperature programming based on the user s tasks. In particular, the design established a language style, visual metaphor and interaction design suitable to the domain and the users. During this process tasks were identified which the user might want to do but which could not be successfully supported within the visual design. For example, instead of having the thermostat make assumptions relevant to saving energy, the user might want to know more about the particulars of these tradeoffs. While a what-if exploration might not be completely supported by our visual design, it could possibly be augmented with speech. Identifying and supporting these tasks is the second design focus.

The same task analysis used to support the visual design was also used to support the agent design; both the agent and the graphical interface have the same basic capabilities. For the first user test, the agent has been designed to be minimally helpful. That is, it makes no proactive suggestions, nor does it offer tutoring or intervention when the user is having difficulties. It does, however, respond as flexibly as possible to the user s questions and requests, according to its abilities and understanding of the request. This design should minimize (though not remove) the impact of the designer s notions of what help the user needs. The next agent design will be based on observations of user difficulties with the basic thermostat functionalities.

The implementation of the thermostat makes use of Collagen (Rich and Sidner, 1998) — a dialogue management system based on theories of human-human collaborative dialogue from computational linguistics. This middleware is geared to capture semantic actions and utterances from both the user and the agent, and to build those into a history structured by an application-specific task library. The agent can then use this history to determine how to react. The history can also be shown to the user, providing a task-level ability to express the desire to undo, retry or replay (Rich and Sidner 1998).

The first test of the design will use the Wizard of Oz technique where a human will play the role of the agent, strictly abiding by the dialogue design. In addition to making empirical observations of people s general heating preferences, this setup enables the collection of a body of human-product dialogues. Analysis of these dialogues should provide information about what sort of assistance the user might need and expect. This includes identifying common patterns of usage (which could be presented as a button in the subsequent visual design), as well as particular aspects of the visual design that create problems for the user (which could be supported in the subsequent dialogue design). These observations also help to verify and expand the current task analysis, towards a more accurate collection of user goals and tasks. Feeding this expanded model back into the dialogue manager and testing it will provide insight into the original questions about the efficacy of aug menting internal models versus modifying the visual design.

Conclusions and next steps

There appear to be a number of viable approaches and theories for building products which can effectively communicate with the user. In particular, attention needs to be paid to the management and support for the dialogue itself, and how these affect user expectations and needs. The efforts described in this paper have begun to address the need for further empirical product-based research into these issues. In order to have a more complete understanding of the implications of collaborative capabilities in everyday consumer products, additional examples of such products, for various types of users and usage contexts, need to be designed and implemented. Additionally, the design and implementation processes would benefit from flexible and usable prototyping tools and analysis techniques, for different types of designers at different stages of design. After designing, building and evaluating more working prototypes, then we can really start talking.

Acknowledgments

We would like to thank Jouke Verlinden and Bernard DeKoven for their assistance in an earlier version of this paper. In addition, Marc de Hoogh has contributed greatly to the design and implementation of test products and the design platform.

References

Cassell, J., Sullivan, J., Prevost, S., and Churchill, E. eds. 2000. *Embodied Conversational Agents*. Cambridge, Mass.: The MIT Press.

Grice, P., 1989. *Studies in the Way of Words*. Cambridge, Mass.: Harvard University Press.

Grosz, B. J., and Kraus, S. 1993. Collaborative plans for group activities. In *Proceedings of the 13th International Joint Conference on Artificial Intelligence*. Chamb ry, France: Morgan Kaufmann.

Keyson, D. V., de Hoogh, M. P. A. J., Freudenthal, A., and Vermeeren, A. P. O. S. 2000. The Intelligent Thermostat: A Mixed-Initiative User Interface. In *CHI 2000 Extended Abstracts*, 59-60. New York, NY: ACM Press.

Nass, C., Fogg, B. J., and Moon, Y. 1996. Can computers be teammates? *International Journal of Human-Computer Studies* 45:669-678.

Rich, C., and Sidner, C. 1998. Collagen: A Collaboration Manager for Software Interface Agents. *User Modeling and User Adapted Interaction* 8:315-350