Resolving the Tension Between Invisibility and Transparency

Anthony Jameson, Stephan Baldes, Mathias Bauer, and Alexander Kröner ⋆

DFKI, German Research Center for Artificial Intelligence

Abstract. After some terminological clarification, we distinguish three possible relationships between invisibility and transparency in the type of system discussed in this workshop. We present examples of the third type of relationship taken from the adaptive, context- and affect-aware personal companion SPECTER.

1 Terminology

Before even explaining what this position paper is about, we start with some terminological clarification.

1.1 How We Understand "Invisibility"

In the context of this workshop, we see *invisible* interfaces as those ambient, wearable, or portable devices that show appropriate responses to the user's behavior while requiring minimal explicit input from the user. (The term is also sometimes applied to information appliances that are so simple to use that users are able to concentrate entirely on their tasks as opposed to the interface itself, even though they have to supply explicit input.)

1.2 Two Meanings of "Transparency"

When we say that a device is *transparent*, we can mean either of two things: (a) you can look straight through the entire device (e.g., if all of its parts are made of glass); or (b) you can look through the outer covering (e.g., a glass box) to examine the inner workings of the device. Both of these senses of the word *transparent* have been used in the HCI literature, mostly in different communities. The second sense is evidently what is meant by the concept of *transparency* in the workshop title; but in the context of that title, the first sense is more strongly suggested. We therefore suggest that we use the concept *comprehensibility* instead of *transparency* in the context of this workshop.

⋆ The research described here was supported by the German Ministry of Education and Research (BMB+F) under grant 524-40001-01 IW C03 (project SPECTER).

2 Relationships Between Invisibility and Comprehensibility

When we think about invisible and comprehensible interfaces, it becomes clearer that these two properties are not in general straightforward to combine; what is invisible is not in general especially comprehensible. Roughly speaking, three relationships between the concepts are possible.

2.1 Invisibility With No Need for Comprehension

The ideal invisible interface would simply do the right thing all the time, making it unnecessary for the user to try to understand it. An example is a door that automatically opens whenever someone approaches it: The "user" does not need to understand how her presence is sensed, as long as the door's behavior is always just what is expected. As this example suggests, this ideal can currently be achieved only with especially simple systems.

2.2 Invisibility Coupled With Simple Mechanisms for Ensuring Comprehension

Even a device as simple as an automatic door will in practice show undesired behavior in some situations—for example, opening and closing repeatedly when someone is standing on the boundary of the region in which it detects a person's presence. A general design principle would be to allow users to understand and avoid such breakdowns by using straightforward means to make the system comprehensible. For example, a visible rubber mat on the floor that looks as if its purpose is to sense the approach of a walker has the advantage over invisible sensors that it makes it clear to walkers exactly how the door decides when to open. A walker is less likely to step intermittently on the mat than she is to step intermittently into the range of invisible sensors. What is involved here is the general HCI design principle of having a system's appearance suggest in a natural, unobtrusive way how the system works (see, e.g., [2]). Although this principle has often been applied to the design of physical objects, appliances, and graphical user interfaces, applying it to invisible systems such as those discussed in this workshop raises new challenges (which will presumably be addressed by some of the workshop presentations). One reason is that any effort along these lines diminishes the system's invisibility to at least some extent. (A rubber mat on the floor is less minimalistic and elegant than an invisible sensor in the ceiling.) One possible approach is to ensure that the clues to the system's workings are so unobtrusive that the user will only attend to them in the exceptional cases where it is necessary for her to understand how the system works.

Simple mechanisms such as these are most likely to work with fairly straightforward types of system behavior. The main focus of this position paper is on the cases in which a system's intelligence and adaptation are too complex to be made comprehensible in this way.

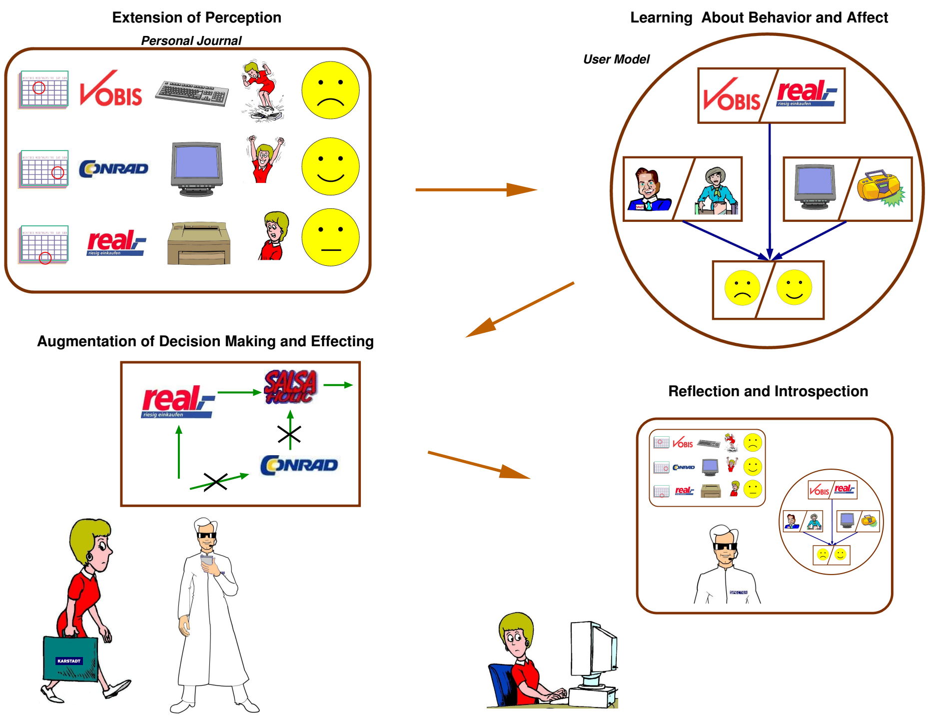

Fig. 1. *Overview of the main functions of the* SPECTER *system.*

2.3 Invisibility That Can Be Attained Only Through Significant Attention and

Effort by the User

As an example of a system where more heavy-handed methods are required for ensuring comprehensibility, consider the system SPECTER (http://www.dfki.de/specter/), of which an overview is given in Figure 1. SPECTER is an adaptive mobile personal companion. It is currently being developed and tested mainly in the context of shopping assistance, although similar functionality is applicable to many other scenarios.

As is shown in the upper left-hand part of the figure, SPECTER uses various types of sensor to record details of the user's activities and affective reactions in a *personal journal*. On the basis of this record, a *user model* is learned which helps the system to interpret and predict the user's behavior and affect. The personal journal and the user model are in turn used as a basis for the functionality that is of central interest in this workshop: supporting the user's activities as she performs tasks within an instrumented environment–in this scenario, a shopping environment.

The lower right-hand part of the figure illustrates a functionality that is uncommon in portable and wearable systems: The *reflection and introspection* component is realized on a full-size computing device such as a PC or laptop. The broad bandwidth of

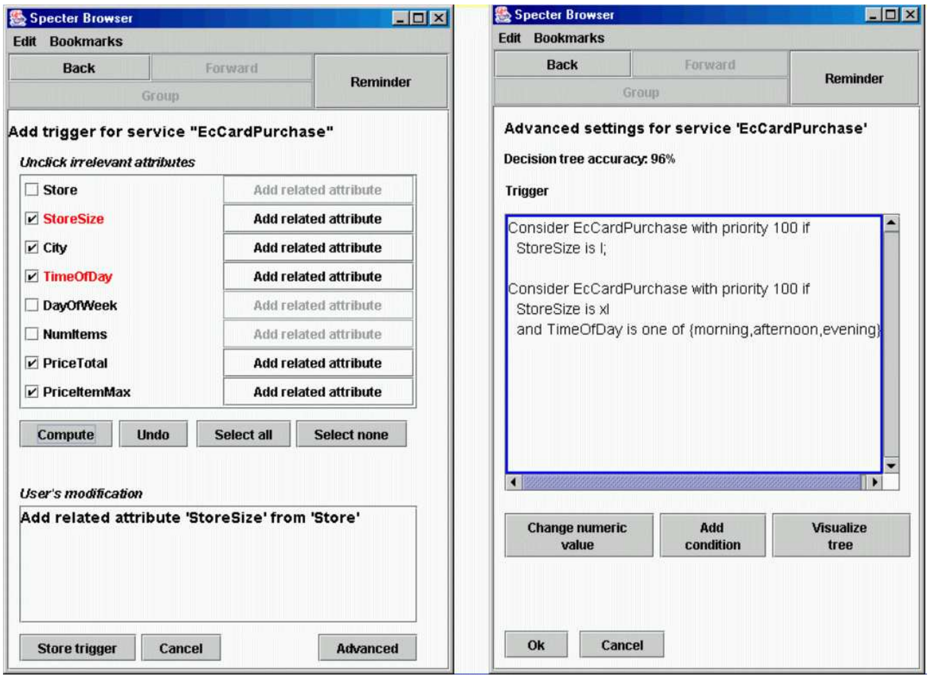

Fig. 2. *Simple (left) and more advanced (right) screens through which a user of* SPECTER *can control the learning of a prediction rule.*

such a device is exploited for the realization of rich interaction between the user and SPECTER: The system visualizes aspects of its personal journal and user model, and it explains how it arrived at them. The user can then critique these models, correcting misinterpretations and supplying additional information that will enhance the quality of the models.

Figure 2 shows an example from the currently implemented version of the SPECTER prototype.[1] . The system has offered to learn a rule for predicting when the user, while shopping, will use her EC card (similar to a credit card) to pay for her purchase. In the relatively simple screen shown at the left, SPECTER has presented a list of attributes of purchasing events which it is capable of recognizing and which might be used to predict the use of the EC card. The user has applied her common-sense knowledge to eliminate from consideration those attributes which she considers obviously irrelevant. The system then learned a decision tree on the basis of the purchasing events stored in the personal journal. The two colored attributes in the figure (*StoreSize* and *TimeofDay*) are the only ones that were found useful by the decision tree learning mechanism.

If the user does not find this amount of insight into (and control over) the learning process to be adequate, she can look at a more advanced screen (shown on the right of

[1] If it is desired, the latest version of the reflection prototype can be demonstrated at the workshop

the figure). This screen shows the exact content of the rules that have been learned so far, as well as their accuracy in covering the training data.

Although the problem of giving insight into machine learning processes has been dealt with frequently in the design of tools for machine learning researchers and those who use machine learning methods in work contexts, providing this type of comprehensibility to nontechnical users raises new design challenges. The solution illustrated in Figure 2 is an initial effort which is currently undergoing testing and refinement.

Apart from the details of this particular interface, the more general issue raised by SPECTER's reflection component is: How can users be motivated to invest effort in critiquing and improving the behavior of an adaptive, sensing-based system, so that it can operate invisibly and effectively? Our working hypothesis is that two design principles need to be applied (cf. [1]):

1. Provision of diverse, multiple benefits to the user as a reward for the effort that the user must inevitably invest. For example, there are a number of possible benefits that a user can derive from interaction with the reflection component, in addition to being able to help the system learn potentially useful prediction rules: Among other things, the system can remind the user of interesting past events; give feedback about how sensibly she has been spending her time; or help her to understand some behavioral tendencies that she may want to change. Even if none of these benefits in itself would make it worthwhile to interact with the reflection component, the sum of all of them may well do so.

2. Separation of the activities of (a) using the system in everyday environments and (b) interacting with it off-line for the purpose of reflection and critiquing. Separation of these activities is a way of avoiding the inherent contradiction between the goals of invisibility and comprehensibility: The latter activity does not have to be simple or unobtrusive, and it can benefit from the use of especially suitable interaction techniques.

3 Concluding Remarks

Aside from the presentation of specific solutions embodied in the SPECTER system, we hope that the concepts and principles introduced in this position paper will help in the construction of a synthesis of the ideas presented by the various workshop contributors.

References

1. Alexander Kröner, Stephan Baldes, Anthony Jameson, and Mathias Bauer. Using an extended episodic memory within a mobile companion. In *Proceedings of the Workshop on "Memory and Sharing of Experience" at Pervasive 2004*, Vienna, 2004. To appear.
2. Donald A. Norman. Cognitive engineering. In Donald A. Norman and Stephen W. Draper, editors, *User Centered System Design: New Perspectives on Human-Computer Interaction*, pages 31–61. Erlbaum, Hillsdale, NJ, 1986.