Personal Technologies, 5, 1, 2001. (http://link.springer.de/link/service/journals/00779/)

Modeling Both the Context and the User

Anthony Jameson
Department of Computer Science, University of Saarbrücken
P.O. Box 15 11 50, D-66041 Saarbrücken, Germany
jameson@cs.uni-sb.de; http://www.cs.uni-sb.de/users/jameson

Abstract

Research into context-aware computing risks losing sight of the user. This paper discusses how different types of information about a user, ranging from information about the current context to information about the user's long-term properties, can simultaneously be relevant to a given adaptation decision. Pointers are given to two areas of research that can help with the integration of a broader range of information into context-aware systems: research on user-adaptive systems and on decision-theoretic methods.

1 Introduction

There is a great deal of excitement nowadays about the possibilities—and the importance—of modeling and taking into account the context of a user's interaction with computing systems (including portable and wearable devices). But this excitement brings with it a danger: that the focus of attention in design may switch too completely from its traditional object—the user—to the context surrounding the user.

What we need is not a shift of focus but an expansion of focus: We need to consider, simultaneously, both the user's context and all of the properties of the users themselves that designers have been learning to deal with during the past two and a half decades.

This paper will argue for this claim with reference to a typical example of a context-aware system, considering how various types of information can be taken into account.

The final section will point to two lines of research that can help designers of context-aware systems to take into account a broader range of information about the user.

2 The Relevance of Context- and User-Related Information

2.1 Using Only Information About the Environment

Let us start with a very simple view of the problem of having a system $\mathcal{S}$ adapt its behaviour to the context of the user $\mathcal{U}$ (Figure 1). The notation used here is a more abstract variant of the sort of notation commonly used for graphical causal models employed in decision-theoretic systems (to be discussed in Section 3.2). Even for the design of systems whose implementation uses no decision-theoretic methods at all, it is worthwhile to examine the underlying conceptualization in this way.

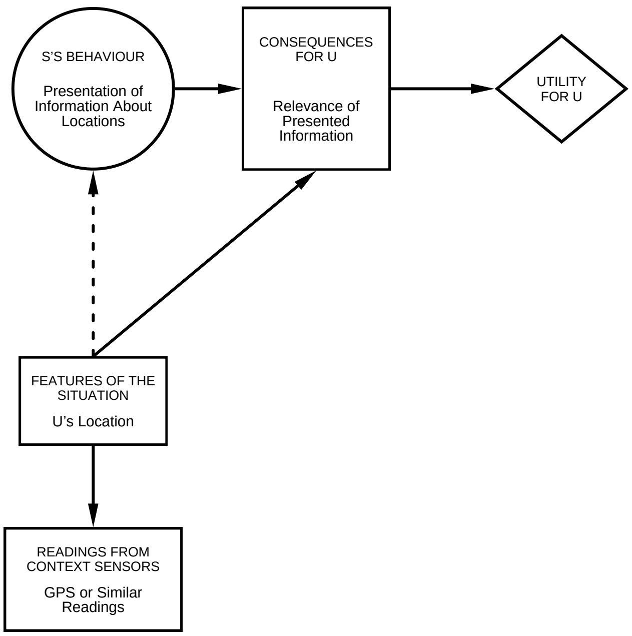

Figure 1 *Using only information about the environment.*

(The lowest box describes a group of observable variables, while the other rectangular boxes describe variables that are not in general observable to the system $\mathcal{S}$. The circle stands for the decision to be made by $\mathcal{S}$: how to adapt its behaviour to $\mathcal{U}$'s situation. The variable represented by the diamond is the overall usefulness of $\mathcal{S}$'s behaviour to $\mathcal{U}$, which is presumably supposed to be maximized by the adaptation. A solid black arrow denotes a causal influence, while a dotted arrow indicates that a variable is taken into account in $\mathcal{S}$'s adaptation decision.)

The figure indicates how a tourist guide system can presumably increase the utility of its behaviour for $\mathcal{U}$ by taking into account $\mathcal{U}$'s situation: Direct observations of $\mathcal{U}$'s context allow $\mathcal{S}$ to infer $\mathcal{U}$'s location. The arrow pointing from FEATURES OF THE SITUATION to CONSEQUENCES FOR U reflects the assumption that the relevance for $\mathcal{U}$ of the information that $\mathcal{S}$ provides—and hence its utility for $\mathcal{U}$—depends on $\mathcal{U}$'s current location. Accordingly, when deciding what information to present, $\mathcal{S}$ will tend to favor information about the current location. (In most systems, which do not make use of decision-theoretic techniques, the reasoning just summarized is done by the designer, who specifies that $\mathcal{S}$ should present information relevant to $\mathcal{U}$'s current location.)

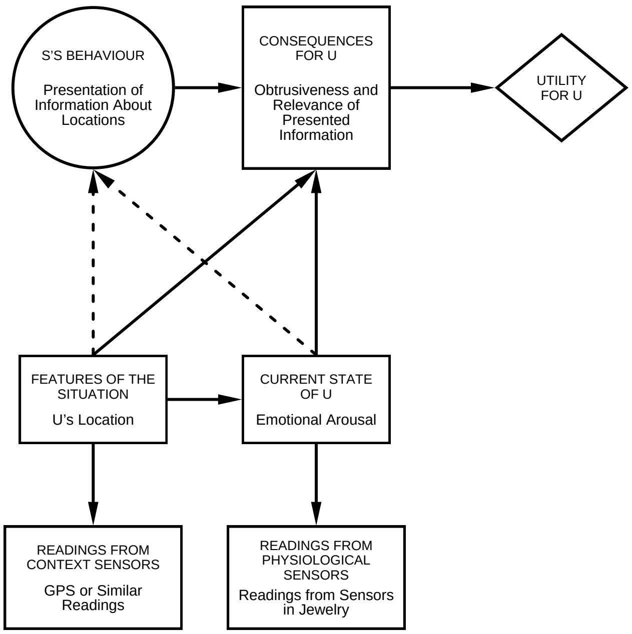

Figure 2 *Taking the user's current state into account.*

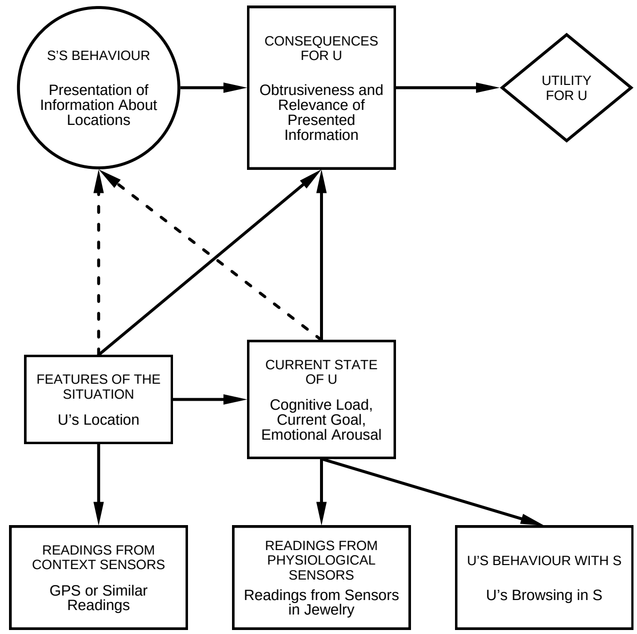

Figure 3 *Adding the user's behaviour as evidence.*

2.2 Taking the User's Current State Into Account

A more complex conceptualization, which is realized in a number of context-aware systems, is illustrated in Figure 2. In addition to $\mathcal{U}$'s context, $\mathcal{S}$ takes into account some aspects of $\mathcal{U}$'s current cognitive and/or psychological state, such as $\mathcal{U}$'s current level of emotional arousal. These short-term properties are not directly observable, but $\mathcal{S}$ can make inferences about them in either of two ways:

- by predicting the effects of FEATURES OF THE SITUATION on the CURRENT STATE OF U;
- by interpreting evidence from sensors placed on or near $\mathcal{U}$'s body.

In our example, the additional arrow pointing from CURRENT STATE OF U to CONSEQUENCES FOR U reflects the assumption that the presentation of a large amount of information can be obtrusive to a highly aroused user, even if it refers to $\mathcal{U}$'s current location.

2.3 Adding the User's Behaviour As Evidence

One further source of information, which is employed in some context-aware systems, is information about the user's behaviour with the system. As Figure 3 illustrates, this information can allow $\mathcal{S}$ to make various types of inference about $\mathcal{U}$'s current state:

1. U'S BEHAVIOUR WITH S can serve as further evidence concerning aspects of the CURRENT STATE OF U that are closely related to the current situation. For example, if $\mathcal{U}$ is experiencing high cognitive load because of distracting events that are taking place in the environment, this fact may be reflected in various types of errors in $\mathcal{U}$'s manual input [1] or speech input [2]. This additional behavioural evidence can be valuable, because (a) sensors are not always available, (b) they cannot in general capture all relevant information about the current situation, and (c) the information that they yield is often only partly reliable.

2. U'S BEHAVIOUR WITH S can also reveal aspects of the CURRENT STATE OF U which are not related to the current context at all. For instance, our example tourist, who is currently at location *A*, may be mainly concerned with planning his or her subsequent visit to some other location *B*. This fact would probably not be recognizable on the basis of information from any sensors. But it might well be reflected in $\mathcal{U}$'s behaviour with the system; for example, $\mathcal{U}$ might be looking at pages with information about location *B*.[1] For $\mathcal{S}$'s decisions about what information to present, the fact that $\mathcal{U}$ is currently interested in finding out about location *B* is presumably at least as important as the fact that $\mathcal{U}$ is now at location *A*.

2.4 Taking Longer-Term User Properties Into Account

Finally, there is often longer-term information about a user that $\mathcal{S}$ should take into account, along with the more quickly changing information discussed so far (see Figure 4). This longer-term information can be of various types, including the following:

- $\mathcal{U}$'s objective personal characteristics (e.g., profession, age, gender);
- $\mathcal{U}$'s level of knowledge of particular topics;
- $\mathcal{U}$'s level of interest in particular topics;
- $\mathcal{U}$'s perceptual and motor skills and limitations.

[1] Chapter 6 of [3] gives a useful overview of a number of context-aware information retrieval systems, including some that take into account $\mathcal{U}$'s current behaviour and other properties that are not specifically related to the current context.

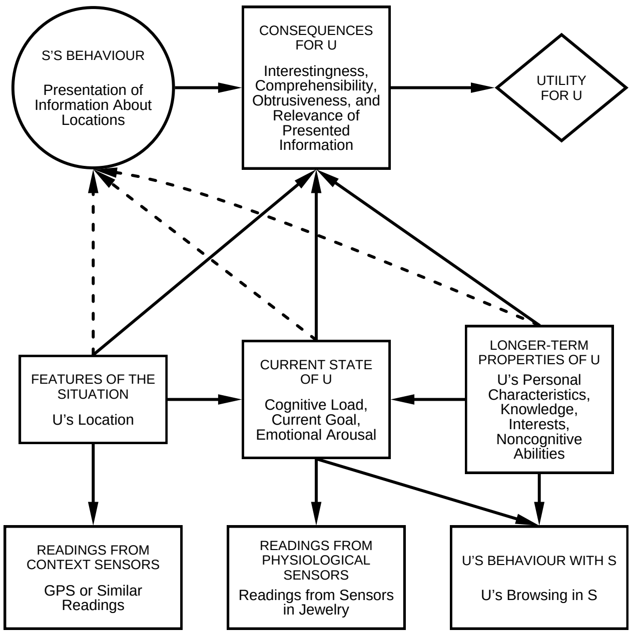

Figure 4 *Taking longer-term user properties into account.*

In our example situation, properties like these could contribute to the determination of (a) what aspects of location B the system should inform $\mathcal{U}$ about; (b) how much detail and background information $\mathcal{S}$ should include; and (c) how the information can best be displayed.

If $\mathcal{S}$ fails to take such longer-term factors into account, even the most sophisticated adaptation to $\mathcal{U}$'s current situation may be unsatisfactory. For example, $\mathcal{S}$ may present information about $\mathcal{U}$'s current (or intended) location which $\mathcal{U}$ finds uninteresting or difficult to understand.

3 Linking With Other Research Areas

It is not easy to take into account within a single system the various types of information shown in Figure 4. In particular, it is not in general straightforward to obtain reasonably accurate assessments of the LONGER-TERM PROPERTIES OF U, or to determine how they will influence the consequences of a system's behaviour. Many ideas on how to solve these problems can be found in two areas of research that have so far not received much attention in the field of context-aware computing:

3.1 Research on User-Adaptive Systems

During the past two decades, researchers in various communities, spanning several disciplines, have developed techniques that enable systems to adapt to their users in many different ways. This research into *user-adaptive systems* has been associated with a number of different labels, including *user modeling*, *student modeling*, *adaptive user interfaces*, *adaptive hypermedia*, *personal learning assistants*, and *personalization*.

Few systems developed in these areas deal with all of the different types of information about users that were mentioned in the previous section; but often several different types are taken into account. Attention to contextual factors has increased sharply during the past few years. Accordingly, some of the methods that have been used for the modeling of other aspects of the user have been extended to deal with aspects of context (see [4] for an early example and [5] and [6] for more recent examples.)

On the whole, research on user-adaptive systems has focused on variables corresponding to the categories CURRENT STATE OF U, LONGER-TERM PROPERTIES OF U, U'S BEHAVIOUR WITH S, and CONSEQUENCES FOR U in Figure 4. By contrast, the main focus in the area of context-aware computing has been on the categories READINGS FROM CONTEXT SENSORS, READINGS FROM PHYSIOLOGICAL SENSORS, and FEATURES OF THE SITUATION. As Figure 4 and the discussion in the previous section have shown, it would be arbitrary and ineffective to deal with these two sets of variables in two separate areas of research.

Publications on user-adaptive systems (under various names) can be found in many different places; the following sources offer especially concentrated coverage:

- the journal *User Modeling and User-Adapted Interaction*;
- the proceedings of the Sixth and Seventh International Conferences on User Modeling.[2]

3.2 Decision-Theoretic Methods

Figure 4 shows the potential complexity and subtlety of the inferences that need to be made. Methods for dealing with this type of inference problem have been developed under the heading of "Uncertainty in Artificial Intelligence" (see, e.g., the on-line proceedings that are available via http://www2.sis.pitt.edu/∼dsl/UAI/uai.html).

Much of the evidence that a system $\mathcal{S}$ can obtain about $\mathcal{U}$'s current situation and/or psychological state is unreliable: Often, it is only on the basis of multiple pieces of evidence that $\mathcal{S}$ can make a useful (though still uncertain) inference. Bayesian networks are a powerful technology for dealing with uncertain evidence in the context of complex causal relationships. (See [7] for the classic exposition and [8] for an introduction that includes references to many user-adaptive systems.)[3] In particular, *dynamic* Bayesian networks make it possible to model properties of the situation and the user that change over time (see, e.g., [6]).

Decisions that are made—implicitly or explicitly—by situation-aware systems need to take into account multiple factors and goals, as well as uncertainty about the relevant variables. Decision-making techniques such as *influence di-*

[2] Available on-line via http://www.cs.uni-sb.de/UM97 and http://www.cs.usask.ca/UM99, respectively). In addition, a tutorial is available via the present author's web homepage: http://www.cs.uni-sb.de/users/jameson.

[3] Easily accessible tutorial material can be found on the web pages for various software packages, including http://www.hugin.com and http://www.norsys.com.

agrams offer ways of dealing with these complications (see, e.g., [9]). Methods for *decision-theoretic planning* (see, e.g., [10]) make it possible for a system, when deciding what to do next, to consider how the next few steps in an interaction might proceed. That is, $\mathcal{S}$ can take into account not only $\mathcal{U}$'s current situation but also $\mathcal{U}$'s possible future situations. For example, when deciding how to present a route description, $\mathcal{S}$ can consider in advance how likely it is that $\mathcal{U}$ will fail to follow particular instructions successfully——and what $\mathcal{S}$ might do in these cases to get $\mathcal{U}$ back on the right track (cf. [11]).

4 Concluding Remark

Context-aware computing does indeed represent a challenging frontier for pioneering researchers. But it shouldn't give rise to an isolated colony that tries to solve all of its problems independently. Instead, lines of communication and supply with existing settlements should be maintained and exploited.

References

1. Kristin Lindmark. Interpreting symptoms of cognitive load and time pressure in manual input. Master's thesis, Department of Computer Science, Saarland University, Germany, 2000.

2. André Berthold and Anthony Jameson. Interpreting symptoms of cognitive load in speech input. In Judy Kay, editor, *UM99, User Modeling: Proceedings of the Seventh International Conference*, pages 235–244. Springer Wien New York, Vienna, New York, 1999.

3. Bradley J. Rhodes. *Just-In-Time Information Retrieval*. PhD thesis, School of Architecture and Planning, Massachusetts Institute of Technology, Cambridge, MA, 2000. Available from http://rhodes.www.media.mit.edu/people/rhodes/research/.

4. David V. Pynadath and Michael P. Wellman. Accounting for context in plan recognition, with application to traffic monitoring. In Philippe Besnard and Steve Hanks, editors, *Uncertainty in Artificial Intelligence: Proceedings of the Eleventh Conference*, pages 472–481. Morgan Kaufmann, San Francisco, 1995.

5. Eric Horvitz, Andy Jacobs, and David Hovel. Attention-sensitive alerting. In Kathryn B. Laskey and Henri Prade, editors, *Uncertainty in Artificial Intelligence: Proceedings of the Fifteenth Conference*, pages 305–313. Morgan Kaufmann, San Francisco, 1999.

6. Anthony Jameson, Ralph Schäfer, Thomas Weis, André Berthold, and Thomas Weyrath. Making systems sensitive to the user's changing resource limitations. *Knowledge-Based Systems*, 12:413–425, 1999.

7. Judea Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, San Mateo, CA, 1988.

8. Anthony Jameson. Numerical uncertainty management in user and student modeling: An overview of systems and issues. *User Modeling and User-Adapted Interaction*, 5:193–251, 1996.

9. Anthony Jameson, Barbara Großmann-Hutter, Leonie March, Ralf Rummer, Thorsten Bohnenberger, and Frank Wittig. When actions have consequences: Empirically based decision making for intelligent user interfaces. *Knowledge-Based Systems*, 13, 2000. In press.

10. Craig Boutilier, Thomas Dean, and Steve Hanks. Decision-theoretic planning: Structural assumptions and computational leverage. *Journal of Artificial Intelligence Research*, 11:1–94, 1999.

11. Thorsten Bohnenberger and Anthony Jameson. When policies are better than plans: Decision-theoretic planning of recommendation sequences. In James Lester, editor, *IUI2001: International Conference on Intelligent User Interfaces*. ACM, New York, 2001.