Editor in Chief: M. Satyanarayanan ■ CMU and Intel Research ■ satya@cs.cmu.edu

Coping with Uncertainty

M. Satyanarayanan

Pervasive computing involves many sources of uncertainty. For example, the accuracy to which a user can determine his or her location is limited by the technology being used. In some cases, the uncertainty is large enough that it is not possible to tell whether someone is inside a particular room or just outside it. The difference could be significant, because entering the room might trigger many actions.

Other aspects of a user's context, such as emotional state, comfort level, and so on, might involve even greater uncertainty. Biometric authentication techniques also involve uncertainty. When a user is angry or upset, attributes such as voice, facial features, or handwriting could change so much that the system rejects the user's authentication attempt. Of course, this might upset the user even more, further complicating authentication.

Uncertainty can also plague communication in the reverse direction, from system to user. For example, how verbose should a system be in keeping its user informed about what is going on underneath the covers? Too terse, and the system appears inscrutable; too verbose, and the large volume of communication distracts the user. How does a system strike a happy medium at all times, even when the environment or user context changes?

This is a difficult problem in the abstract but might be tractable in some specific situations. For example, Maria Ebling describes an interesting approach to striking this balance in the context of handling cache misses in the Coda File System.[1,2] Caching is attractive in distributed file systems because it is completely transparent—neither users nor applications are aware of it. Unfortunately, servicing a cache miss on a large file over a low-bandwidth wireless network takes so long that most users would rather be asked first whether they really need the file. However, a flurry of such questions can overwhelm the user. Ebling's approach uses a predictive user model to resolve this tension. On a cache miss, the system consults an internally maintained *user patience model* to predict whether the user will respond positively to a fetch request. If this appears likely, the system suppresses user interaction and transparently handles the fetch—why bother the user if his or her answer is likely to be "yes"? The patience model is static in Ebling's implementation, but we can conceive of a more sophisticated implementation in which the model dynamically adapts to user context.

How verbose should a system be in keeping its user informed about what is going on underneath?

From a broader perspective, computing has come full circle with respect to uncertainty. At the dawn of the information age, we faced a choice between analog and digital computing. Digital computing won the day because of its ability to eliminate uncertainty in state representation and transformation, even after a vast number of computing steps. It is ironic that in today's all-digital world, uncertainty reappears as a major concern at a higher level of representation.

This issue of *IEEE Pervasive Computing* focuses on building systems that deal with uncertainty. "Bayesian Filtering for Location Estimation" discusses how to reduce uncertainty in location estimates through statistical techniques. "Coping with Uncertainty in a Location-Based Game" takes a more qualitative approach. It discusses the impact of uncertainty in location sensing on user experience. It describes this impact in the context of a game and presents the authors' attempts to overcome the effects of this uncertainty. "Managing Context Information in Mobile Devices" describes a context management framework and API that simplifies the programming complexity of developing systems that handle uncertainty.

Finally, "Using Trust for Secure Collaboration in Uncertain Environments" addresses the problem of uncertainty in the context of security. This uncertainty leads to risk, and the authors describe techniques to handle this risk. P

REFERENCES

1. M. Ebling, *Translucent Cache Management for Mobile Computing*, doctoral dissertation, CMU-CS-98-116, School of Computer Science, Carnegie Mellon Univ., 1998.

2. M. Satyanarayanan, "The Evolution of Coda," *ACM Trans. Computer Systems*, vol. 20, no. 2, May 2002, pp. 85–124.

Published by the IEEE CS and IEEE ComSoc ■ 1536-1268/03/$17.00 © 2003 IEEE