# PANEL: Towards a Better Understanding of Context and Context-Awareness

**Moderators**

Gregory D. Abowd and Anind K. Dey

Graphics, Visualization and Usability Center and College of Computing,
Georgia Tech, Atlanta, GA, USA 30332-0280
`{abowd, anind}@cc.gatech.edu`


**Panelists**

Peter J. Brown[1], Nigel Davies[2], Mark Smith[3], Pete Steggles[4]

[1] Computing Lab., The University of Kent at Canterbury, Canterbury, Kent, CT2 7NF, UK
`P.J.Brown@ukc.ac.uk`
[2] Computing Dept., Engineering Building, Lancaster University, Lancaster, LA1 4YR, UK
`nigel@comp.lancs.ac.uk`
[3] Hewlett Packard Laboratories, Palo Alto, CA, USA
`msmith@hplmts.hpl.hp.com`
[4] AT&T Laboratories Cambridge, 24a Trumpington Street, Cambridge CB2 1QA, UK
`pjs@uk.research.att.com`

## 1  Introduction

When humans talk with humans, they are able to use implicit situational information, or *context*, to increase the conversational bandwidth.  Unfortunately, this ability to convey ideas does not transfer well to humans interacting with computers. In traditional interactive computing, users have an impoverished mechanism for providing input to computers. By improving the computer's access to context, we increase the richness of communication in human-computer interaction and make it possible to produce more useful computational services.  The use of context is increasingly important in the fields of handheld and ubiquitous computing, where the user's context is changing rapidly. In this panel, we want to discuss some of the research challenges in understanding context and in developing context-aware applications.

We define context as any information that can be used to characterize the situation of an entity, where an entity can be a person, place, or physical or computational object. We define context-awareness or context-aware computing as the use of context to provide task-relevant information and/or services to a user. Three important context-awareness behaviours are the presentation of information and services to a user, automatic execution of a service, and tagging of context to information for later retrieval. Some of the main challenges in the area of context-aware computing are:

- the development of a taxonomy and uniform representation of context types;
- infrastructure to promote the design, implementation and evolution of context-aware applications; and
- a discovery of compelling context-aware applications that assist our everyday interactions with ubiquitous computational services.

## 2  Position Statements

Following are the position statements from each of the panelists.

### 2.1  Peter Brown (www.cs.ukc.ac.uk/research/infosys/mobicomp/Fieldwork/)

My view is that future research must focus more on the whole solution rather than on part solutions. There are two strands to this. The first is general to most research. When I was working with a company to create a hypertext product, I was told "People do not want hypertext: they want solutions. Hypertext is only likely to be *part* of a solution". I often quote this maxim, as one can substitute almost any research area for hypertext. Certainly it applies to context-awareness.

The second strand concerns bringing together technologies. You could get any of the following opinions from individual researchers in context-awareness:

- the key is the underlying database.
- context-awareness is a type of information retrieval.
- what we need to do is to solve the HCI issues.
- we need a solid theoretical foundation.
- if you want this to be real, you need to solve hard AI problems.
- it is a systems/software engineering problem.
- the area is waiting for the right hardware at the right price.
- context-awareness is a specialized area of distributed   systems.

All of these are at least partly true. The real key, however, is bringing the technologies together.

My theme, therefore, is that currently we researchers think too much in terms of our own pet technology. The reality is that this will only be part of context-awareness, and, if we are hoping for our research to be applied, this in turn will only be part of the solution. Certainly such success as we have had at the University of Kent at Canterbury has been the result of people with different outlooks getting together, but we, like others, still have a long way to go.

### 2.2   Nigel Davies (http:// *www.guide.lancs.ac.uk)*

Context sensitive systems and applications are those that respond to changes in their environment. Typically these responses are designed to improve a systems' performance or to make its behaviour more relevant to the situation in which it is being used. For the purposes of this panel I propose to argue that in the near future all mobile systems will be context-sensitive: after all, why would you want a system which was not context sensitive? I will provide examples to justify this claim based on the results of our work on the Lancaster GUIDE project. This project has developed and deployed a context-sensitive tour guide in the city of Lancaster which combines mobile computing hardware, context-sensitive applications and wireless communications to provide visitors to the city with context sensitive, interactive applications.

A relevant publication is: Davies, N., K. Cheverst, K. Mitchell, and A. Friday. "Caches in the Air: Disseminating Information in the Guide System" Proc. 2nd IEEE Workshop on Mobile Computing Systems and Applications (WMCSA '99), New Orleans, U.S., IEEE Press, Pages 11-19.

### 2.3   Mark Smith

Context aware computing has the potential to allow applications in emerging service based computing architectures to provide completely new functionality.  Client devices can use a wide variety of sensors to provide awareness of who the user is, and the location and environment in which the application is being used.  Environment awareness can include such things as whether or not a user is still holding a device, if the user alone or with others, stationary or in motion, hot or cold, light or dark and so forth.  Using this information, appliances and applications can be optimized and personalized in ways that provide benefit to both technology providers and users.  To realize this vision, I see six broad research areas in context aware computing.  They cut across many of the technical areas  in servers, services, transport and clients, both fixed and mobile,  that make up future service centric system architectures.

1.  Algorithms to render physical significance from sensor data.  This is most obvious in applications such as biometric based user authentication, but it can become very challenging in areas of environmental awareness.  For example, an algorithm that exploits sensor data to robustly determine something seemingly simple such as if a device is still in the possession of a user can be difficult.
2.  Technology for integrated sensors.  Some sensors such as active pixel arrays can currently be integrated onto CMOS chips without using special processing, but what about other sensors such as motion or humidity.  How can sensors be designed to allow multiple functionality, for example an imaging sensor that can be used for both biometrics and communication.  Integrated optics is another challenge both for capture and display.

3. Privacy issues. These go way beyond the security and encryption issues we have now. I would not be surprised if in the future entire conferences are given to this topic.
4. Agents that make use of sensor data. How are agents to be constructed that support dynamic persona in an application? How is the data that these agents use to be distributed? These are problems at the server and transport level.
5. Personalization of applications and appliances. One of the greatest benefits of using context aware computing is the ability to use sensor data to personalize applications and devices to conform to what the user expects. This is from the consumer's point of view the 'killer app'. These are problems at the application and client device level.
6. Capitalizing on the new spaces that context awareness enables. What customers will pay for, and what not. What partnering will be necessary across the emerging service centric computing community that is forming in order to make context aware computing real.

### 2.4  Pete Steggles (www.uk.research.att.com/~pjs)

What is needed in context-aware computing is a sufficient 'kit' of composable small services which really benefit from context-awareness. But with the right parts you can put together what I would call killer applications. We have just put together a system which combines follow-me desktops, camera tracking, text-to-speech, follow-me sound playing and 3D visualization to provide a recorded tour of our lab where we take a visitor round with a slide show following us, demo the technologies referred to in the show at the time, track the visitor with cameras and record the whole thing, giving the visitor a souvenir CD with a multimedia version of their tour with synchronized video. For us, when we have fully deployed our system, that will be a killer app!

Any description of the world that can be relevant to an application counts as a 'context type'. In our research we have attempted to create a very detailed model of the environment to address this. Inevitably, with so few real context-aware applications, there is doubt as to which facts will be useful and which won't. With respect to location, we have got a long way by attempting to model vague location-related facts using geometric containment and overlapping relationships. Really accurate location is 99% of the battle.