Using Context as a Crystal Ball: Rewards and Pitfalls

Keith Cheverst, Nigel Davies, Keith Mitchell and Christos Efstratiou
Distributed Multimedia Research Group
Department of Computing
Lancaster University
Lancaster, LA14YR, U.K.
+44 (0)1524 594539
{kc, nigel, mitchelk, efstrati}@comp.lancs.ac.uk

1. Introduction

The use of context (both situated and environmental) has significant potential for simplifying the user's understanding of, and interaction with, complex interactive systems. Due to the size of devices appearing on the horizon, e.g. smart phones, the interactive systems that run on these devices are likely too share very limited input/output bandwidth at the interface between the system and the user. Therefore, techniques for simplifying interaction and reducing any input/output bottleneck is certainly desirable in order to serve the notions of ubiquitous computing [Weiser,91] and 'Information appliances' [Norman,99].

This paper discusses some of the potential rewards and pitfalls that can await designers wishing to incorporate context-awareness [Schilit,94][Brown,97] into interactive systems. Many of the issues are described in anecdotal form, based on our experiences developing and evaluating the context-aware GUIDE system [Cheverst,99][Cheverst,00].

To introduce some of the issues that arise concerning context-awareness, consider that popular interactive system: the car, and, more specifically, its braking system.

The Anti-locking Braking System (ABS) is a context-aware system that was introduced as a safety measure in order to reduce braking distance and greatly reduce the potential for a driver causing their car to skid through excessive braking. The context sensed by the antilock braking system includes:

i) Whether the driver is currently trying to brake (i.e. situated context),

ii) Whether or not the wheel is currently 'locked' under braking (i.e. environmental context).

The adaptive element of the system involves detecting when the wheel is locked and then decreasing braking force until the wheel is no longer locked. At this point more braking force is applied provided the situated context is still that of braking.

Before the advent of ABS, drivers were required to develop mental models that took into account the complex interrelationship that exists between braking force and the friction between the car's tyres and the road. The number of accidents each year involving cars skidding illustrates the fact that many drivers miscalculate the aforementioned relationship.

Consideration of the ABS system allows us to identify an agent, acting on behalf of the driver, that reduces the mental and physical demands of driving the car. In effect, the agent takes some control (or power) away from the driver and (providing the driver prefers less rather than more interaction with the car) makes the car easier to drive.

The ABS system enables the driver to form a simplified mental model regarding the cars braking system, i.e. drivers don't need to have such a detailed comprehension of the rules governing 'excessive' braking force and the resultant lack of control. However, if the car's driver is used to a conventional, manual, braking mechanism he or she might have learnt the skill of ‘pumping the brakes’ in order to prevent the car from skidding. Unfortunately, if this skill is employed by the driver of a car with ABS, the two approaches can conflict causing the braking distance to be increased. This example highlights three potential pitfalls that can arise from adapting to context, namely:

i) The problem of failing to reach a stable state [Thimbleby,90].

If both the user and the system attempt to adapt to the current context then it is unlikely that the system will manage to reach a stable state. Under such circumstances the system is likely to appear unpredictable. When designing context-aware systems, it is clearly important to consider the background/expertise of the user, i.e. are they likely to have already formed a mental model for interacting with a similar (non-adaptive) system?

ii) The trade-off between prescription and freedom.

If the driver wanted, for whatever reason, to lock the wheels of the car then the system would prevent him or her from achieving this task.

iii) The user must trust the agent performing adaptation on his or her behalf.

When ABS was first introduced, there was, not altogether surprisingly, some mistrust of the system by drivers. Indeed, the driver who knows the workings of the ABS system is required to trust both the context sensing technology and the intelligence of the agent, i.e. its infallible ability to react appropriately to the context in a failsafe manner.

It is possible to identify three main ways in which context can be used to simplify the user’s interaction with an interactive system:

i) Simplifying/reducing the task specification required from the user in order to achieve his or her desired goals, i.e. reducing the need for input/action by the user.

At one level this can simply mean filling in a required blank, such as the user’s current location, based on information that is sensed by the system. However, at a higher level, it can also involve attempting to pre-empt the user’s current goal in order to reduce his or her task specification (e.g. the ABS system).

ii) Changing the output produced by the system, i.e. the reducing quantity of information that has to be processed by the user or increasing the quality of information presented.

Once again, some reduction in output might be achievable by attempting to pre-empt what output is likely to be required/expected by the user, based on the current context.

iii) Reducing the complexity of rules constituting the user’s mental model of the system.

This is generally achieved by some form of intelligent agent that performs some proportion of the required computation on the user’s behalf.

The following section describes and analyses some of the positive and negative experiences of using context gained through our development and evaluation of the GUIDE system.

2 Experiences Developing a Context-aware Tourist GUIDE

The GUIDE system has been developed to provide visitors to the city of Lancaster with information that is tailored to his or her context. The city contains a number of strategically positioned wireless communication cells with a diameter of approximately 300 m depending on the layout of buildings. These communication cells are used for disseminating location information and tourist information to mobile GUIDE units. By carrying a GUIDE unit, visitors will receive up-to-date information about the city's attractions while following a structured tour of the city tailored to their specific requirements.

One of GUIDE's key requirements [Cheverst,00] was that of flexibility. In more detail, visitors using the system should be able to change from one kind of situated interaction to another. So, for example, the visitor should be able to change from following a tour to booking accommodation and then resume their tour. For this reason, we designed a user interface that enables the user to switch between different aspects of functionality in order to meet the demands of his or her current situation (figure 1).

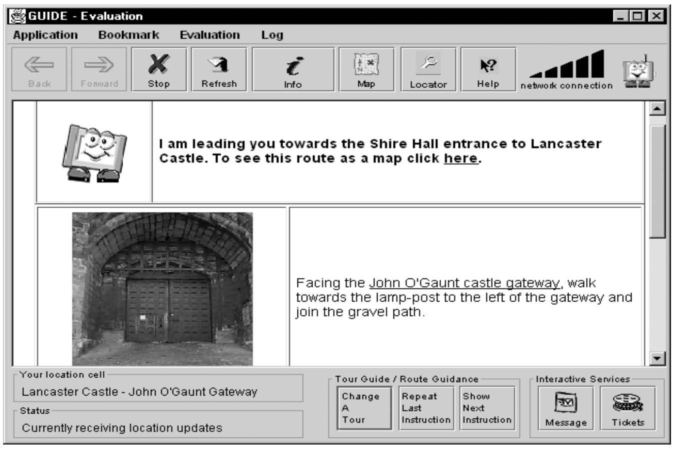

Figure 1: The user interface presented to a visitor when following a tour.

Following some initial evaluation, a compromise was made to reduce some of the flexibility afforded by the user interface in order to increase ease-of-use. The compromise involved making 'following a tour' the default mode of situated interaction, until the tour has either been completed or aborted.

In order to describe the way in which GUIDE adapts to context the three categories identified in the introduction will be used:

i) **Simplifying/reducing the task specification.**

One type of context utilized by GUIDE is that of the visitor's location and the location of attractions within the city. The system reduces the need for input by assuming that the information required by the visitor is strongly influenced by his or her current location. So, for example, a visitor standing outside Lancaster castle can request the system to 'tell me about the area I am in' as opposed to searching the contents page for 'Information on Lancaster Castle'. However, the initial GUIDE system made the mistake of only allowing visitors to obtain information regarding their current location and so the over-determination [Thimbleby,90] employed by the system to simplify the visitors task was inappropriate. The lack of flexibility in this version frustrated users and so the system was extended to enable visitors to specify their requirements more fully, e.g. by searching for information using a keyword. This last point clearly illustrates the intrinsic problem of trying to successfully pre-empt the goal of the user.

ii) **Changing the output produced by the system**.

In general, the GUIDE system attempts to constrain and tailor information presented to the visitor based on his or her current context. So, for example, when the visitor requests a list of nearby attractions, the list is constrained in such a way that those attractions that are open, and have not already been visited, are placed higher up the list. The assumption is made that the visitor is more likely to be interested in attractions that are open and that have not already been visited. An earlier version of the system constrained the output by removing all closed attractions from the presented list. However, this frustrated some visitors who were interested in visiting the attraction anyway, e.g. to view the architecture of a building. Again, this demonstrates the difficulty of pre-empting the user's goal. A future version of the system will use the visitor's stated interest in architecture to determine whether closed attractions with a clear architectural value are included in the list.

iii) **Reducing the complexity of the user's mental model.**

In GUIDE, the agent that acts on behalf of the user is designed to relieve the user of the onerous task of studying maps and guidebooks in order to devise and follow an interesting tour. In more detail, the agent calculates tours based on a variety of different contexts, such as the visitor's current location, the current time, special opening hours of attractions, the relative positioning of attractions in the city and the preferences of the visitor, e.g. an interest in historic buildings.

While evaluating the GUIDE system we experienced some difficulty capturing the visitor's location context with sufficient accuracy. Of course, the problem with obtaining inaccurate or incorrect contextual information is that the adaptation performed by the system, based on the context, will produce inappropriate results. In the case of GUIDE this meant that when presenting a list of 'nearby attractions' to the visitor, some of the attractions were not always as 'nearby' as might have been expected.

3. Strategies for Building Context-Aware Applications

A number of strategies can be identified for the design of interactive systems that utilize situated and/or environment-based context. The following strategies are based on our analysis of existing context-aware interactive systems, such as ABS, and, in addition, those concerns that were experienced during the development and evaluation of the GUIDE system:

i) When using context to constrain the presentation of information, or to simplify the specification of a task, it is crucial that the adaptation does not inappropriately over-determine the users interaction.

ii) Furthermore, designers need to carefully consider the fundamental trade-off between prescription and freedom/flexibility when deciding how to adapt to context.

iii) When considering adaptation to context, designers should be careful to bear in mind the principal of least astonishment and the need for predictability. Of course, if designed well, then adaptation to context has the potential to increase the integral predictability/consistency of the system. However, as described in the ABS example, the inappropriate transfer of skills can cause difficulties.

From a more technical perspective, the following issues need to be considered when engineering context-aware systems.

i) The sensing technology used for obtaining context needs to be dependable. This means both accurate and available in a timely manner.

ii) The intelligence of the agent responsible for adapting to context needs to be flexible in order to cope with problems obtaining context and the potential for over-riding demands by the user.

Hopefully, by considering these strategies/issues, designers of context-aware interactive systems can avoid many of the potential pitfalls.

4. Conclusion

This paper has considered some of the potential rewards and pitfalls of utilizing situated and/or environment-based context in the design of interactive systems.

In summary, adaptation to context can be used to develop interactive systems that allow users to form a simplified mental model for understanding, and interacting with, the system. In effect, context-aware systems migrate complexity away from the user to some form of intelligent agent.

Context also has the ability to reduce the complexity of a user's task specification and reduce the quantity (and/or increase the quality) of information output by the system. This ability is crucial when considering the generation of interactive systems on the horizon that will be designed to run on future mobile computing devices. Such devices are likely to share very limited input/output bandwidth at the interface between the user and the device and so maximizing the use of this bandwidth is of paramount importance.

To conclude, providing designers consider the potential hazards, context-awareness could help to bring the ubiquitous information appliance one step closer to reality.

References

[Cheverst,99] Cheverst, K., N. Davies, K. Mitchell, and G.S. Blair. "The Design of an Object Model for a Context-Sensitive Tourist Guide", Computers and Graphics, Vol. 23. No 6, December 1999, pp. 883-891.

[Cheverst,00] Cheverst, K., N. Davies, K. Mitchell, A. Friday and C. Efstratiou. "Developing a Context-aware Electronic Tourist Guide: Some Issues and Experiences, to appear in Proceedings of CHI'00 (Netherlands, 2000), ACM Press.

[Thimbleby,90] Thimbleby, H. User Interface Design. Addison-Wesley Longman Ltd, ISBN 0-201-41618-2. (1990)

[Norman,99] Norman, D.A. The Invisible Computer. MIT Press. ISBN 0-262-14065-9. (1999)

[Weiser,91] M. Weiser, "The Computer for the Twenty-First Century", Scientific American, 265 (3), September 1991, pp. 66-75.

[Schilit,94] Schilit, B., Adams, N., and Want, R. Context-Aware Computing Applications. Proc. Workshop on Mobile Computing Systems and Applications, Santa Cruz, CA, U.S. (1994).

[Brown,97] Brown, P.J., Bovey, J.D., and Chen, X. Context-aware applications: from the laboratory to the market place. IEEE Personal Communications, Vol. 4, No. 5 (1997) pp 58-64.