

The Rationale for Infrastructure Support for Adaptive and Context-Aware Applications: A Position Paper

Nigel Davies^{1,2}, Keith Cheverst¹, Christos Efstratiou¹ and Adrian Friday¹

¹ Distributed Multimedia Research Group, Computing Department, Lancaster University,
Bailrigg, Lancaster LA1 4YR, England

{nigel, kc, efstrati, adrian}@comp.lancs.ac.uk

² Department of Computer Science, Gould-Simpson Building,
University of Arizona, AZ 85712
nigel@cs.arizona.edu

Abstract. Research has demonstrated that mobile and wireless applications benefit from a knowledge of their operating environment. Examples of context-aware or adaptive applications have been constructed and evaluated with the results being widely disseminated within the research community. However, the field is still sufficiently new that there are currently no standards for describing, disseminating or managing the necessary contextual information. Moreover, there are no standards (or even accepted best practices) for coordinating adaptation across multiple applications and systems. In this position paper we argue that the lack of standards in this area will inhibit the widespread deployment of many of the compelling context-aware mobile applications currently being designed.

1 Introduction

The benefits of context-aware and adaptive applications are well established: such systems can offer significant improvements in performance and functionality over their non-adaptive counterparts. However, to date the architectures and implementations developed to support such applications have been somewhat ad-hoc in nature. In particular, most systems focus on one specific type of adaptation or awareness, e.g. adapting to changes in network characteristics or end-system location, rather than taking an holistic approach that encompasses the many different issues that must be addressed in supporting such applications. In this paper we argue that new infrastructure services provide a means of supporting adaptive and context-aware applications and that by working to create uniformity and standardization of these services, we can accelerate the widespread development and deployment of adaptive and context-aware applications and systems.

2 Context-Aware and Adaptive Applications

For a number of years researchers have recognized that one of the defining features of mobile environments is the concept of *change*. Early work in the field of mobile computing tended to focus on change as it applied to network quality-of-service (QoS) [4], [15]. In particular, many researchers postulated that future mobile end-systems would be multi-homed with applications and system components being required to operate over different wireless networks with widely varying QoS characteristics. The solutions proposed typically involved a simple sensor-actuator feedback loop with possible adaptation strategies including adjustment of protocol parameters [7], modification of protocol semantics from synchronous to asynchronous operation [14] and code/functionality migration [14]. Such adaptive systems tend to offer significantly better performance than their non-adaptive counterparts, especially when operated over low-bandwidth networks. For example, the MOST project [4] reported a significant improvement when using an adaptive middleware platform over GSM (as compared to the non-adaptive version of the platform). Similar improvements were reported in other adaptive systems. In addition to this work on network oriented adaptation, significant work has been carried out in developing adaptive systems that attempt to modify their behavior to minimize power consumption (e.g. [16]).

In parallel with this work on adaptive systems, the early 90s saw significant activity in the field of context-aware or context-sensitive systems. Early examples include the location-aware electronic post-it note developed by Brown [2], a wide range of location-based systems from ORL as part of the Active Badge project [25] and the Cyberguide project at Georgia Tech [18]. Perhaps the most complete piece of work in this field was the context-aware system developed at PARC by Schilit [21] which proposed a framework for writing context-aware applications. Research in this field has continued with more recent projects such as the GUIDE project at Lancaster [3] that has developed a context-sensitive tour-guide system and the TEA system at Karlsruhe that uses sensors attached to mobile phones to provide context-aware functionality such as call re-direction [22].

Researchers have also begun to investigate how mobile systems can exploit services in the infrastructure as part of their contextual awareness. A prime example of this is the RADAR system [1] in which a mobile end-system can automatically locate nearby printers and then provide the user with directions on how to find the physical devices. Other examples include the work of Ishii where infrastructure components are used to supplement the capabilities of the mobile end-system [12]. Finally, systems such as [11] and [13] provide access to location-based services.

This research in the field of context-awareness and adaptation has produced a wide range of compelling scenarios and motivated the emergence of major areas of commercial activity such as the development of location-based services.

3 The Problems of Adopting an Ad-Hoc Approach

The research described in section 2 has demonstrated the advantages of adding adaptive and context-aware behavior to mobile applications. However, the current research systems adopt a largely ad-hoc approach to supporting this behavior. There are notable exceptions, such as [5], [20], [21]. However, these systems typically only take into account a small subset of likely contextual triggers and do not address issues such as, for example, service discovery.

In the following sections we highlight, using examples where appropriate, the problems of adopting an ad-hoc approach to supporting adaptation and contextual-awareness.

3.1 Adaptive Systems

The principal problem with current approaches to supporting adaptation is that they provide little support for coordinating behavior across multiple applications or system components. This problem of coordination between adaptive applications was first identified by Noble et. al. as part of their work on the Odyssey system [20]. While Odyssey provides support for sharing resource information between applications and hence helps to coordinate adaptation strategies, the primary focus is on adapting to changes in network QoS. However, it is increasingly widely recognized that mobile systems need to adapt to a wide range of attributes such as user activities, location, service availability and power levels in addition to network QoS. In current systems this is achieved using the control feedback loop described earlier: applications monitor the resources of interest and adapt to changes in the availability of these resources. However, a lack of coordination between applications can cause a number of problems [6]. A simple example is when multiple applications are running on the same end-system and are making uncoordinated accesses to the end-system's local disk. If the system is trying to conserve power by spinning down the disk when idle, then coordinating these applications in order that regular disk accesses (caused by features such as auto-save) occur within a small time-window could significantly reduce the system's overall power consumption.

When additional triggers for adaptive behavior are introduced the situation becomes complex and uncoordinated adaptation strategies may cause conflicting behavior. For example, consider a mobile end-system that is running applications that can adapt to both power and network connectivity. The end-system detects that the available power is reaching a critical level and informs the application to reduce its power consumption: the application does so by reducing its network requirements. However, this leads to a corresponding increase in the availability of network resources that might lead other applications to increase their network traffic.

Finally, most adaptation systems only monitor local resources. Hence, it is difficult to coordinate adaptation across multiple distributed components. This is an important issue when adaptation strategies involve the instantiation of network services such as proxies or filters. To make an informed decision about the use of such services the system needs to have more information than can be easily collected on the end-system itself.

3.2 Service Discovery

There is currently intense interest in the use of service location and interaction platforms such as SLP [9], Jini [23] and UpnP [19] to support ubiquitous and context-aware computing. Examples of initiatives in this space include [1]. However, as the list of technologies mentioned above suggests, there are currently a number of different service discovery and interaction technologies that can be used to support mobile applications. Further examples include HAVi [10], Salutation [24] and X.10. The majority of these platforms are specialized for a particular operating environment and it appears unlikely that convergence will occur in the near future. As a result, either gateways or new platforms will be required in the fixed infrastructure or mobile clients will need to support multiple discovery protocols. However, it is important to note that there are substantive differences in current service discovery and interaction technologies that can impact on application design and implementation. As a result, designing applications that can use multiple service interaction technologies and gateways that support interworking between different service interaction technologies are both non-trivial tasks.

The situation is further complicated by the fact that the current set of service interaction technologies are missing many of the features required to support ubiquitous applications [8]. Examples of these shortcomings include:

- Lack of scalability. Current systems (with the exception of SLP) are primarily designed to operate in small scale networks and do not scale well as the number of services and devices increases.
- Lack of state-based query support. None of the systems currently support queries for services based on dynamic state. For example, in UPnP to find a printer with paper would require a client to query the network in order to obtain a list of all printers and then access each individually to check their state. Future clients would have to repeat this process.
- Lack of support for service meta-data. It is currently impossible for third-parties to augment service descriptions. For example, users in a network may wish to augment a service description of a printer with comments about its tendency to jam at key moments.

Other examples of omissions in current service interaction technologies are given in [8].

3.3 Location Awareness

We have experienced first-hand the problems of adopting an ad-hoc approach to supporting contextual awareness during our work on the GUIDE system [3]. This system provides visitors to the city of Lancaster with an electronic city-guide that provides information to the visitor tailored to their preferences and the current context such as location. In the current system we use location beacons transmitted using a city-wide 802.11 network to provide the location context for this application. However, we would clearly like to be able to use GUIDE in cities that do not support this form of location information. Hence, a technology independent form of location information is desirable. Of course achieving this raises a large number of issues

including resolving the differences between symbolic and geographic location models (symbolic models identify locations using a naming scheme rather than absolute geographic coordinates) and addressing security and privacy concerns. An excellent treatment of these issues can be found in [17].

4 The Role of Infrastructure Support

Based on the examples in section 3 we believe that infrastructure support for adaptation and contextual awareness is required if we are to witness rapid and widespread deployment of applications that exploit such techniques. The precise nature of this support is clearly a matter for further research. However, we identify the following features that we believe this support should provide.

1. An agreed mechanism for applications to obtain and interpret contextual and adaptation information. This mechanism should support both local and remote sources of information and should adequately address issues such as privacy. Moreover, it is important that the mechanism is specified in sufficient detail to support access to independently developed information sources, yet is flexible enough to allow the integration of new sources of information as they become available.
2. An agreed set of mechanisms for discovering services. Of course this may be related to point 1 above in a number of different ways: a service discovery mechanism could be used to discover sources of contextual information or services could appear as additional pieces of contextual information.
3. An agreed set of mechanisms for coordinating applications and system responses to adaptation and contextual information. Once again, such a set of mechanisms must span both local and remote sources of information and components that are required to adapt.

Clearly this is not intended to be a complete list of requirements. For example, significant work is still needed on designing appropriate adaptation policies for devices that use contextual information.

It should also be noted that in considering the above points it is important to stress that these issues should be tackled from a mobile computing perspective. More specifically, many of the technologies currently being considered for such activities are based around heavyweight protocols/architectures such as http and SOAP and include payloads containing XML. Such approaches are clearly not optimized for transmission over low-speed links or to resource-poor mobile end-systems.

5 Concluding Remarks

In this brief position paper we have presented the rationale for developing infrastructure support for context-aware and adaptive applications. We believe that such support is necessary in order to facilitate the rapid and widespread development and deployment of applications that will bring significant benefits to end-users. While the exact nature of this infrastructure support is clearly a matter for further research, we have identified a number of core requirements that need to be considered as part of

this work. These requirements include the need for common mechanisms for obtaining and interpreting contextual information, an integrated approach to service discovery and a set of mechanisms for coordinating applications and system adaptation in response to multiple contextual triggers.

References

1. Bahl, P., Padmanabhan, V. N.: RADAR: An In-Building RF Based User Location and Tracking System. In Proc. of IEEE INFOCOM 2000, Tel-Aviv, Israel (March, 2000) 775-784
2. Brown, P.J.: The stick-e document: a framework for creating context-aware applications. In Proc. of EP'96, Palo Alto, U.S. (1996) 259-272
3. Cheverst, K., Davies, N., Friday, A., Mitchell, K.: Experiences of Developing and Deploying a Context-Aware Tourist Guide: The Lancaster GUIDE Project. Proc. of Mobicom'00, Boston, U.S. (2000)
4. Davies, N., Blair, G.S., Cheverst, K., Friday, A.: Supporting Adaptive Services in a Heterogeneous Mobile Environment. In Proc. of the Workshop on Mobile Computing Systems and Applications, Santa Cruz, CA, U.S. (1994)
5. Dey, A., Abowd, G., Salber, D.: A Context-Based Infrastructure for Smart Environments. In Proc. of 2000 Conference on Human Factors in Computing Systems. (2000)
6. Efstratiou, C., Cheverst, K., Davies, N., Friday, A.: An Architecture for the Support of Adaptive Context-Aware Applications. Proceedings of Mobile Data Management (MDM 2001), Hong Kong. (2001)
7. Friday A., Davies, N., Blair, G., Cheverst, K.: Developing Adaptive Applications: The MOST Experience. Journal of Integrated Computer-Aided Engineering, 6(2). 143- 157
8. Friday, A., Davies, N., Catterall, E.: Supporting Service Discovery, Querying and Interaction in Ubiquitous Computing Environments. Proc. of 2nd MobiDE Workshop, Santa Barbara, U.S. (2001)
9. Guttman, E., Perkins, C., Veizades, J., Day, M.: Service Location Protocol, Version 2. RFC 2608. (1999)
10. HAVi Consortium.: HAVi Specification V1.0. (2000)
11. Hodes, T.D., Katz, R.H., Servan-Schreiber, E., Rowe, L.A.: Composable ad-hoc Mobile Services for Universal Interaction. In Proc. ACM Mobicom'97, Budapest, Hungary. (1997) 1 - 12
12. Ishii, H., Ullmer, B.: Tangible Bits: Towards Seamless Interfaces between People, Bits and Atoms. In Proc. of Conference on Human factors in computing systems (CHI '97), Atlanta, ACM Press. (1997) 234-241
13. José, R., Davies, N.: Scalable and Flexible Location-Based Services for Ubiquitous Information Access. In Proc. 1st International Symposium on Handheld and Ubiquitous Computing, HUC'99, Karlsruhe, Germany. (1999)
14. Joseph A., Tauber, J., Kaashoek, F.: Mobile Computing with the Rover Toolkit. IEEE Transactions on Computers: Special issue on Mobile Computing, 43(3), (1997)
15. Katz R.: Adaptation and Mobility in Wireless Information Systems. IEEE Personal Communications, 1(1). (1994) 6-17
16. Kravets R., Krishnan, P.: Application-Driven Power Management for Mobile Communication. In Proc. 4th ACM International Conference on Mobile Computing and Networking (MOBICOM '98). (1998)
17. Leonhardt, U.: Supporting Location-Awareness in Open Distributed Systems. Ph.D. Thesis, Dept. of Computing, Imperial College London. (1998)

18. Long, S., Kooper, R., Abowd, G., Atkeson, C.: Rapid Prototyping of Mobile Context-Aware Applications: The Cyberguide Case Study. In Proc. of the 2nd ACM International Conference on Mobile Computing, Rye, NY, ACM Press (1996)
19. Microsoft Corporation.: Universal Plug and Play Device Architecture Reference Specification, Version 1.0. Microsoft Corporation. (2000)
20. Noble, B., Satyanarayanan, M., Narayanan, D., Tilton, J.E., Flinn, J., Walker, K.: Agile Application-Aware Adaptation for Mobility. In Proc. of the 16th ACM Symposium on Operating System Principles, St. Malo, France. (1997)
21. Schilit, B., Adams, N., Want, R.: Context-Aware Computing Applications. In Proc. of the Workshop on Mobile Computing Systems and Applications, Santa Cruz, CA, U.S. (1994).
22. Schmidt, A., Takaluoma, A., Mäntyjärvi, J.: Context-Aware Telephony over WAP. Personal Technologies Volume 4(4). (2000) 225-229
23. Sun Microsystems Inc.: Jini Architectural Overview – January 1999. Sun White Paper. (1999)
24. The Salutation Consortium.: Salutation Architecture Specification (Part 1) Version 2.0c. (1999)
25. Want, R., Hopper, A., Falcao, V., Gibbons, J.: The Active Badge Location System. ACM Transactions on Information Systems, Vol. 40, No. 1. (1992) 91-102