

© [2005] IEEE. Reprinted, with permission, from [Peter Leijdekkers and Valerie Gay, Personalized Service and Network Adaptation for Smart Devices, Communications, 2005 Asia-Pacific Conference on, 5-5 Oct. 2005]. This material is posted here with permission of the IEEE. Such permission of the IEEE does not in any way imply IEEE endorsement of any of the University of Technology, Sydney's products or services. Internal or personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution must be obtained from the IEEE by writing to [pubs-permissions@ieee.org](mailto:pubs-permissions@ieee.org). By choosing to view this document, you agree to all provisions of the copyright laws protecting it

# Personalized Service and Network Adaptation for Smart Devices

Peter Leijdekkers and Valerie Gay

Computer Systems Department  
University of Technology, Sydney  
P.O. Box 123 Broadway, NSW 2007, Australia  
Peter.Leijdekkers@uts.edu.au, Valerie.Gay@uts.edu.au

**Abstract**– The availability of smart devices with integrated GSM/GPRS/WiFi and the rollout of public hotspots allow users to be always online at reasonable costs. Personalised and context aware applications will become available in the forthcoming years due to the wide availability of smart devices and the interest of telecom operators and service providers to provide personalised services. For the user to access his preferred network and services in a particular context we need to have some mechanisms in place and an infrastructure that reacts autonomously on behalf of the user. This paper proposes a solution based on context-aware user profiles and their associated user preferences. It describes a smart device centered solution and a prototype has been built for Microsoft Windows Mobile™ Pocket PCs to validate the ideas.

**Index Terms**– personal communications, network and service management, human interface, ubiquitous computing, WiFi, profile information.

## I. INTRODUCTION

Mobile devices equipped with WiFi (802.11b) are becoming a commodity. The latest smart phones have built-in WiFi and can be easily connected to a wireless network. Also mobile phones move into the WiFi area and several manufacturers have released smart phones with integrated GSM/GPRS and WiFi [1] [2].

On the network infrastructure side a trend is visible to move towards wireless technologies for all our communication needs. Companies in Australia such as Unwired [3] and iBurst [4] offer wireless broadband connectivity to its users and we already see an increase in mobile devices that can connect to these networks. According to the research firm TelecomView, WiMax and other new high-speed wireless technologies are likely to take market share from 3G as well as DSL. It says that WiMax will supplement and in some cases replace 3G, DSL and other wireline technologies to provide broadband services [5].

In parallel many public hotspots are available in major cities and traditional telecom operators are rolling out 802.11 access points to complement their wireless infrastructure [7] [8]. It is expected that WiFi and WiMAX [4] will be widely available in dense populated areas in the forthcoming years.

The above implies that ‘always online’ comes within reach at reasonable costs. The next step for mobile devices will be offering tailored services based on the context and

preferences of the user. Besides access to email and Internet the user will want to have easy access to specific services depending on the context. For example, if a user commutes using public transport he might want to have access to the train/bus/ferry timetable showing information related to his current location or willing to receive promotional offers from shops he passes by. Whereas at home, the mobile device should offer home related services such as a remote control functionality or access to a pizza delivery service.

These types of context aware services can be classified as passive or active. Chen and Kotz [10] define *active context-aware* services as services that change their content autonomously on the basis of sensor data whereas *passive context-aware* applications only present the updated context to the user and let the user specify how the application should change, if at all. Barkhuus and Day [9] conducted a study related to the user acceptance of context aware services that react autonomously. Their conclusion is that users are willing to accept a large degree of autonomy from applications as long as the service’s usefulness is greater than the cost of limited control [9]. This conclusion seems promising for the acceptance of context aware applications by end users.

Security and privacy are important aspects for context aware services and infrastructures supporting these services. In particular issues related to confidentiality, integrity, availability, privacy and trust put constraints on the context aware services and infrastructure. The pervasiveness and invisibility of the devices and their networking capacity allows for the possibilities to collect information from loosely coupled personal data which makes privacy a crucial success factor for the acceptance of context-aware systems [12]. One of the problems related to both privacy and security is the deployment of huge numbers of almost invisible sensors and computers implying that users can no longer easily be asked to consent to data collection or processing. An interesting idea is to bind privacy to physical proximity as suggested in [13]. For example, sensor data may freely be distributed and processed close to the location where it is collected but must adhere to privacy regulations otherwise.

This paper proposes a solution where we add a context-aware dimension to local end-user profiles and allow the end-user to specify its preferences in terms of services and networks for a particular context-aware profile. When the

user changes profile (manually or automatically triggered by sensors or third parties) we use the corresponding profile preferences to discover services and networks that the user specified for that profile. The objective is to offer a solution where the end user is able to easily manage his context-aware profiles.

In this paper we focus on technical aspects related to context-aware profiles. A prototype has been developed for Microsoft Windows Mobile™ Pocket PCs based on the architecture and ideas presented in this paper.

## II. CONTEXT-AWARE USER PROFILES

Being always connected does not mean that the user wishes to use the mobile device for the same purpose all the time and everywhere. Throughout the day a user can have different roles and different interests which should be reflected in the way the mobile device reacts and the way the user interacts with it.

Lewis and Patel highlight the need for an infrastructure to support the user acting in different roles at different times and suggest that user profiles should be further categorized [17]. They define a role-based profile being based on user roles (e.g. consumer, employee) along with their interests (e.g. sport, shopping). It also takes into account their behaviors (mood, surrounding conditions) plus time and location sensitive information (local services, time critical services).

The infrastructure presented by Lewis and Patel [17] follows a telecom-operator approach and stores/manages most of the user profile information on the server. In our approach, the context-aware user-profile is managed and stored on the local device. Storing profiles locally on a device has advantages over storing profile information on a server [11]. It allows for personalised actions even if the device is not connected to a network. Additionally, sensitive profile data can be better protected on the device than storing it on a server where the user has no control on how that data is being used. Additionally, WiFi hotspot providers often provide limited internet access (walled garden). In this case it is not guaranteed that a user can access a profile stored on a specific server. Furthermore, the user may want to change a profile depending on the location and if a server oriented approach is used every update needs to be transferred to the server which can be a serious communication overhead. For performance reasons it is cumbersome to contact a server each time to retrieve profile information. Therefore our approach is to store as much profile information as possible on the mobile device. Having said that many internet services do require a user profile to be stored on the server. This is often service specific information used by that internet service to offer personalised services. This information is independent of the profile related information. For example a Meeting service needs detailed user information in order to provide the matches. Other internet applications do not require a profile and perform their services based on user input. A bus timetable application falls in this category.

Similar to Lewis and Patel, we categorize the context-aware user profiles. However our classification is slightly different from [17] since we are dealing with a subset of profile information. That is the information that can be managed by the smart device itself as opposed to the profile information managed by a server oriented approach in [17] supporting many users and services. The user has a limited but extensible set of profiles (e.g. work, meeting, home, commuting) and is allowed to associate services and network preferences to each profile. Services can be anything from local applications such as word, contacts or email to Internet services (e.g. online banking). Networks can range from Wireless Wide Area Networks such as GPRS, UMTS and WiMAX to Personal Area Networks such as Bluetooth and UWB [18].

Each context-aware user profile stores a variety of information including:

- The networks a user wants to use in that particular profile. The latest smart devices offer the possibility to use GSM, GPRS, WiFi or Bluetooth to connect to the Internet. A user could indicate its preferred connection method for a particular profile. For example in an office environment the corporate WiFi can be the preferred option and GPRS the second option.
- The services a user wants to use in that particular profile. Services can be local applications running on the smart device or Internet services. When a profile change occurs the question arises whether applications in the activated profile should start automatically or the user should activate them. Automatic activation can be very intrusive and undesirable for certain applications. A fine-grained control should be introduced where the user can indicate whether an application can react automatically if the profile is activated.
- General settings such as security or privacy for a profile. These settings have an impact on the networks and services used in this profile. A user can indicate various levels of visibility for a profile. For example if the privacy level is set to "Do not disturb" for an office profile, applications like MSN Messenger will use this setting.

From a user perspective it should be easy to create and customise a profile. Also 3<sup>rd</sup> parties could offer a preconfigured profile which can be easily downloaded by the user (e.g. a university providing a 'Uni' profile to its employees and students). Preconfigured profiles can be offered by 3<sup>rd</sup> party providers to facilitate the use of these profiles by its users. Since users roam in many areas a standardised way of offering preconfigured profiles is desired. XML is a good candidate since it is designed for openness and is widely accepted.

## III. ARCHITECTURE

Figure 1 shows the main components that make up the context-aware profile adaptation architecture.

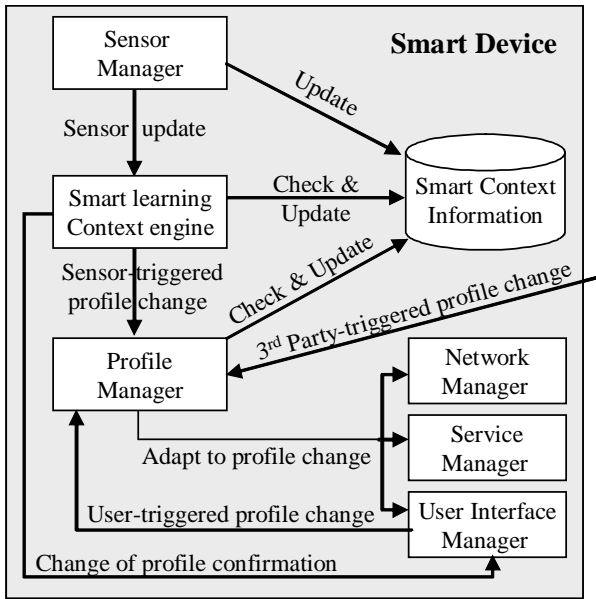


Figure 1. Architecture for context-aware profile adaptation

A profile change can be triggered by the *user*, *sensors* or a *3<sup>rd</sup> party*. Sensor triggering implies that we use intelligence and sensors on the mobile device to trigger the change of profile. An example of a 3<sup>rd</sup> party trigger can be a Meeting service that needs access to the current location but cannot retrieve this information due to the current active profile that prohibits revealing location information. A 3<sup>rd</sup> party triggered profile change can be intrusive and violating security and privacy rules.

The **Sensor Manager** manages and collects information from the sensors available on the device and updates the smart context information database. Sensors commonly available on the latest smart devices are 802.11, Cell ID, Bluetooth and GPS. Other sensors can be easily integrated in the architecture when they are available such as movement, light and temperature sensors. Also regarded as sensor is a module that regularly checks the calendar on the mobile device and triggers an event in case of an upcoming appointment.

The **Smart Learning Context Engine** uses intelligent algorithms to determine the profile. It determines whether a profile change has occurred.

If it detects a profile change but it is a new situation, it needs confirmation from the user. The user can then confirm (or not) that he has changed profile and wishes to have that change occur automatically next time. This decision is stored in the smart context information database.

The Sensor Manager prompts the Smart Learning Context Engine when new sensor data is available. When a profile change has occurred the smart engine instructs the profile manager to change profile.

The **Smart Context Information Database** stores all data that is needed by the components such as

- Sensor data that is collected by the sensor manager. Due to the variety of sensors we have one table for each type of sensor.
- Smart engine related information which includes rules to relate sensor data to a particular context which in turn relates to a profile. Historic data such as past user decisions.
- Profile information as described in section 2.

The **Profile Manager** manages the context-aware profiles. Typically a user has several profiles. The profile manager will switch to a new profile based on one of the 3 triggers (i.e. user, sensor or 3<sup>rd</sup> party). It will activate the preferred networks and services according to the settings described for this profile. Part of the profile manager is the *3<sup>rd</sup> party profile manager* that offers functionality to download profiles offered by 3<sup>rd</sup> parties. This allows preconfigured profiles to be easily installed in the device and it also allows users to share a particular profile with other users. An example would be a group of friends wanting to share their ‘surfer profile’ so that they get weather alerts and tailored advertising about surfing when this profile is active.

The profile manager initiates several actions in the mobile device when a profile change occurs.

1. If the network preference changes, the **Network Manager** tries to connect to the preferred network as specified in the new profile. If the preferred network is not available, alternatives can be proposed to the user.
2. The **User Interface Manager** will load the services as specified in the profile and adapts the user interface to show the services for that profile.
3. When a particular profile is active, the services in this profile will be aware of the current context and react accordingly. For example, the print manager application in the office profile will only show the printers that are close to the user instead of showing all available printers. Most services would only be activated by the user but for certain services the user might want them to start in the background. The **Service Manager** performs this task and the device will try to connect to the service. For other services, such as a train timetable service, receiving alerts is allowed.
4. The profile manager confirms the change of profile to the smart engine. This is important if the change was not triggered by the sensors.

The architecture described above is highly modular and can be easily extended. Based on this architecture we have developed a prototype.

#### IV. PROTOTYPE

Figure 2 shows the test bed installed at the University of Technology, Sydney (UTS) where we experiment with ubiquitous services. The faculty of IT building is covered by 802.11b Access Points (AP) which are freely accessible by students and staff (Figure 2, left bottom cloud). In addition

we have three experimental access points hooked up to a PC with a 'page redirect' software installed [19]. The experimental APs simulate public hotspot providers and when a mobile device accesses one of these APs the user will be redirected to a welcome page (see Figure 3) to download the prototype.

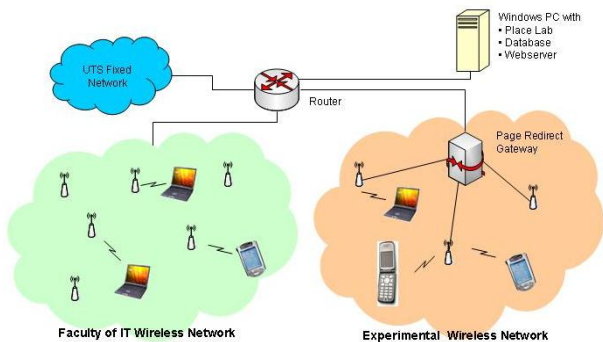


Figure 2. Test bed setup

The current version of the context aware profile application runs on Microsoft Windows Mobile Pocket PC platforms. We selected this platform due to easy access to lower level APIs which are needed for the sensor manager module. Also the tight integration with the operating system allows easier access to the applications running on the mobile device such as the calendar application. We used the .Net Compact Framework extended with OpenNETCF [15] modules to build the application.



Figure 3. WiFi Hotspot welcome page

At the time of writing we have implemented a WiFi and Bluetooth sensor that periodically send data to the Sensor Manager. The Sensor Manager decides if the Smart Learning Context Engine should be notified with the new sensor data. Additional sensor implementations are planned for the near future such as a GPS and the use of Calendar information.

The Smart Context Information component is implemented using SQL Server CE which is a compact database for mobile devices.



Figure 4. Example of a Profile

Figure 4 depicts a screenshot of an active profile. It shows the applications as URL links and when the user activates a link the application will be launched. Applications can be local PDA applications as shown on the left column in Figure 4 or internet applications (Figure 4, right column). If possible the application is started with the preferences for that profile.

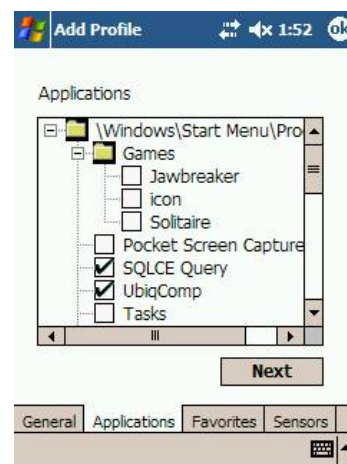


Figure 5. Profile Configuration

The user can modify the profiles and specify the applications it wants to use. Figure 5 shows a screenshot where the user can select the applications that are locally available on the device.

## V. CONCLUSIONS

This paper presented a solution to personalise services and network preferences for smart devices based on the context of the user. We believe that the user should be in control regarding information pushed to him on his smart device. Therefore, the user can personalise his smart device by identifying for each profile (e.g. office, home, commuting) his associated preferences in terms of networks and services.

A change of profile can be triggered by the smart device itself by using sensors and other information to detect a change of context that can lead to a change of profile. It can be triggered by a third party (e.g. in case of emergency showing the closest emergency exit) or by the user himself. When a change of profile occurs, the smart device adapts the services, networks and user interface to the associated user preferences for that profile.

We have implemented parts of the architecture. The user-triggered profile change is fully functional. We are improving the sensor-triggered profile change and try to make it more intelligent and less intrusive. In particular we will include the calendar information to improve the accuracy of a correct context change.

The next step is to conduct trials with end users to test the usefulness of context aware applications for mobile devices.

#### ACKNOWLEDGEMENTS

The authors would like to thank Nicolas Harris for his contribution to the prototype and valuable feedback on this work.

#### REFERENCES

- [1] Motorola, 'MOD Mobile Office Device CN620', [http://www.motorola.com/wlan/docs\\_pdf/cn62mod.pdf](http://www.motorola.com/wlan/docs_pdf/cn62mod.pdf), assessed February 2005.
- [2] Nokia, Nokia 9500 communicator features, [www.nokia.com/nokia/0,,54108,00.html](http://www.nokia.com/nokia/0,,54108,00.html), assessed February 2005.
- [3] Unwired, 'No Wires No Wait No Worries', [www.unwiredaustralia.com.au](http://www.unwiredaustralia.com.au), assessed February 2005.
- [4] iBurst 'Whenever Wherever It's Up To You', [www.iburst.com.au](http://www.iburst.com.au), assessed February 2005.
- [5] 'WiMAX to steal 3G and DSL market', [http://www.theregister.co.uk/2004/12/10/wimax\\_steal\\_market\\_share/](http://www.theregister.co.uk/2004/12/10/wimax_steal_market_share/), assessed March 2005.
- [6] Wimax trends, 'The Net's Leading Resource for WiMAX Technology & Solutions' <http://www.wimaxtrends.com/index.htm>, assessed March 2005.
- [7] Michael Sainsbury, 'Telstra plans WiFi future', <http://australianit.news.com.au/articles/0,7204,10384190%5E15306%5E%5Enbv%5E,00.htm>, assessed February 2005.
- [8] The Age, 'Telstra extends wireless coverage', <http://theage.com.au/articles/2005/03/01/1109546846250.html>, assessed March 2005.
- [9] L. Barkhuus, A. Dey, 'Is Context-Aware Computing Taking Control away from the User? Three Levels of Interactivity Examined', A.K. Dey et al. (Eds.): UbiComp 2003, LNCS 2864, pp. 149–156, 2003. Springer-Verlag Berlin Heidelberg 2003
- [10] G. Chen and D. Kotz, 'A survey of context-aware mobile computing research'. Paper TR2000-381, Department of Computer Science, Dartmouth College, November 2000.
- [11] S. Riché, G. Brebner, Mickey Gittler, 'Client-Side Profile Storage', E. Gregori et al. (Eds.): Networking 2002 Workshops, LNCS 2376, pp. 127–133, 2002. Springer-Verlag Berlin Heidelberg 2002.
- [12] Xiaodong Jiang, James A. Landay, 'Modeling Privacy Control in Context-Aware Systems', in IEEE Pervasive computing, July-September 2002.
- [13] M. Langheinrich, 'Privacy by Design - Principles of Privacy-Aware Ubiquitous Systems', In Proceedings of Ubicomp 2001. Atlanta 2001.
- [14] B. Clements, 'Future Bottlenecks in the Information Society', Study for the ITRE Committee of the European Parliament, Section 4.5, June 2001, <ftp://ftp.jrc.es/pub/EURdoc/eur19917en.pdf>, assessed March 2005.
- [15] OpenNETCF.org, 'The Premier .NET Compact Framework Shared Source Site', <http://www.opennetcf.org>, assessed March 2005.
- [16] Hull, R. Kumar, B. Lieuwen, D. Patel-Schneider, P.F.; Sahuguet, A. Varadarajan, S. Vyas, A. 'Enabling context-aware and privacy-conscious user data sharing', Proceedings of IEEE International Conference on Mobile Data Management, Pages:187 – 198, 2004.
- [17] Lewis, N.; Patel, A., 'Smart companion', First IEEE Consumer Communications and Networking Conference CCNC, Pages 605 – 610 January. 2004.
- [18] Intel, Ultra-Wideband (UWB) Technology, [http://www.intel.com/technology/comms/uwb/download/wireless\\_pb.pdf](http://www.intel.com/technology/comms/uwb/download/wireless_pb.pdf), accessed March 2005.
- [19] ControlAP, 'Share the Internet Anywhere', <http://controlap.com/>, accessed March 2005.