NOTIFICATION SERVICES FOR MOBILE AND WIRELESS TERMINALS

A Class of Services Appropriate for Mobile Scenarios

Michael Decker, Rebecca Bulander
Institute AIFB, University of Karlsruhe, Englerstr. 11, 76 128 Karlsruhe, Germany

Keywords: Mobile and wireless services, notification services, context-awareness.

Abstract: We introduce the concept of a notification service and show that while simple this class of services is especially suited for mobile scenarios and many useful mobile services can be considered as notification services.

1 INTRODUCTION

Many publications in the field of mobile business and mobile computing deal with platforms or frameworks for mobile services whereas often the focus is on architectural considerations or how to process context information. What is often missing is an exact description of the class of service supported by the platform. There are also many services which are just mobile version of services known from the conventional internet. Those services are not always appropriate for mobile scenarios, e.g. users don't want to browse web sites for hours when they use a mobile terminal.

In this article we will introduce the concept of a notification service (NS). The basic idea is that because of the technical limitations and the ergonomic restrictions of mobile terminals along with the typical habits of usage in mobile scenarios one cannot expect much more of users than to perform a short configuration of a service and read notification message on mobile devices. A clear description of NS is provided along with examples that create mobile value and a discussion why this class of services is very suitable for scenarios with mobile terminals.

The remainder of this article is organized as follows: in the second section we introduce the concept of a NS. We argue why this class of mobile services is appropriate for mobile scenarios in chapter three. Examples for mobile services that could be implemented as NS are listed in section four, in section six we conclude with a summary and outlook.

2 NOTIFICATION SERVICES

2.1 Definition

A service is a certain set of functionalities offered by one or more servers (backend system) to clients; in the case of mobile (and wireless) services these clients are mobile terminals (MT) like cellular phones, PDAs or smartphones and at least the first leg of the data transmission between clients and backend systems depends on standards for wireless data communication like GPRS, WiFi or UTMS.

According to our definition a notification services (NS) is a service that based on configuration and context information can send one or more push-messages to a user's MT. Push messages in this sense are messages that the recipient perceives as not being directly requested[1] (e.g. SMS/MMS, e-Mail, WAP-Push). Context information (in mobile computing) is information that is available in explicit form at the runtime of a service (or application) and deliberately used to support the user's interaction with the service (resp. application), see also Dey (2001). We distinguish personal context and public context information: personal context information is information that

[1] From the technical perspective a push message might be a pull messages (explicitly requested) indeed.

Decker M. and Bulander R. (2006).
NOTIFICATION SERVICES FOR MOBILE AND WIRELESS TERMINALS - A Class of Services Appropriate for Mobile Scenarios.
In *Proceedings of the International Conference on e-Business*, pages 151-156
DOI: 10.5220/0001427601510156

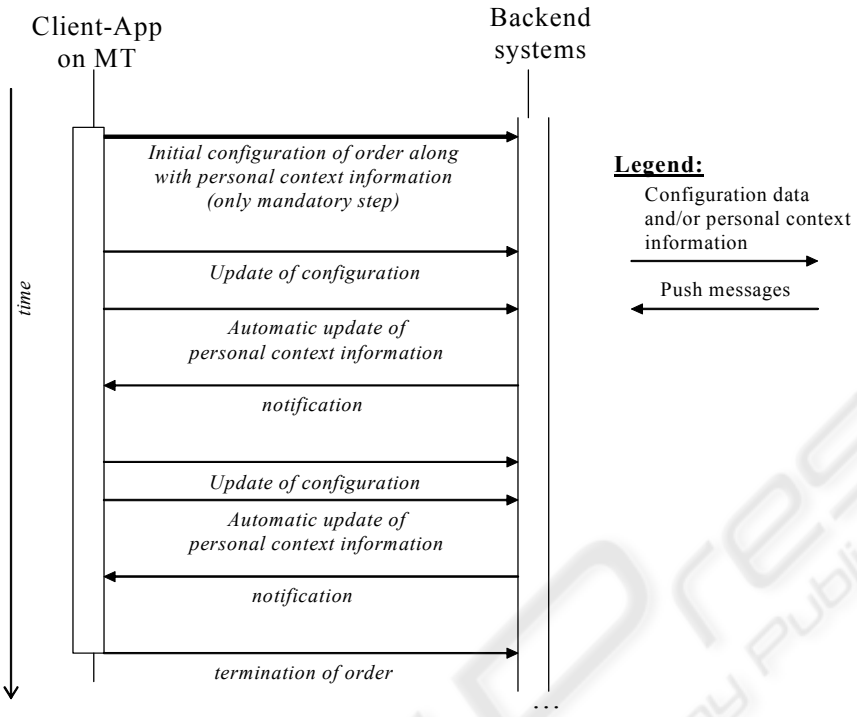

Figure 1: Sequence diagram for the lifespan of a NS instance.

contains person-related data (like current location of user, his profile); in contrast to public context information (e.g. time, weather) there are privacy concerns relating to personal context information.

The semantics of the notification messages and the configuration data and context information required depends on the type of the service. We denote a configured instance of a distinct type of NS as "order".

One type of mobile services we could implement as notification service is "location based advertising" (Kölmel & Alexakis, 2002): according to his profile and current position the user receives advertisement messages on his MT of nearby shops. We use this example to illustrate the basic principle of NS: the configuration data is a wish list of products and services the user is interested in (e.g. "clothing", "gastronomy", "entertainment"); the context information are the profile of the user (e.g. age, gender) and the current location and time. The push-messages (notifications) are the messages with the advertisement.

Figure 1 shows a generic sequence diagram for NS. Please note that the only mandatory message exchange is the initial configuration of the service (bold arrow), all other steps are optional; even the notification messages are optional (e.g. if the advertising service cannot find matching offers).

The notifications are not bound to one channel like SMS or e-Mail for a given order. It is thinkable that a NS sends messages as MMS to the user under normal conditions, but if the battery power of the user's MT is below a given level (personal context parameter) SMS are sent instead to help to save energy; or the service may be configured to send MMS instead of the usual SMS if the event detected is considered extraordinary important, e.g. a certain stock price exceeded or falls below a given value. There might be even an application-specific push-channel provided by the mobile client application.

2.2 Technical Details

NS in our sense require a special client application on the MT and a machine readable description file (preferred XML-based format) for each service type. The description file specifies

- which attributes the user has to configure, e.g. formulated using a subset of the XForms-language (Dubinko , 2003)
- which personal context parameters are of interest (e.g. current location, profile-data, battery level, …) and have to be filled in

and updated automatically by the client application.

All information needed for the configuration must be included in the description file, so no network interaction between client and server is necessary during the configuration of the NS. According to the description file the client application renders a GUI containing the necessary widgets (e.g. checkboxes for the wish list in the advertising example) so the user can configure the order.

After the reception of a notification the user may change the configuration of the service, but every interaction beyond that is out of the responsibility of the NS, e.g. a notification may include a URL which points to a WML- or cHTML-document, but the retrieval of that document is not part of the notification service itself.

NS as defined above seem to be very limited at the first glance, but we will show below that there are a lot of examples for reasonable mobile services that can be considered as NS. Furthermore they are appropriate for the technical possibilities of mobile services. Also this simplicity is beneficial for the analysis of this class of services, especially with regard to privacy concerns.

3 SPECIAL SUITABILITY OF NS FOR SCENARIOS WITH MOBILE DEVICES

Mobile services have to face several limitations (Satyanarayanan, 1996), but NS meet these limitations:

Features of wireless communication: Wireless data communication is unreliable, (yet) relatively expensive and of limited bandwidth. One has to assume connection drop outs when developing a mobile service and take the variability of the connection quality into account. Temporary connection failures should be hidden from the user. NS meet this requirement: when the user interacts with the NS (configuration and reading notifications) no network interaction is required. If during sending new configuration data to the backend system there is a connection drop out the client application simply tries to submit the data again some time later; the user won't mention this.

Limited resources of MT: MT have limited energy supply. We don't assume that there will be a leap towards significant better batteries in the next few years although we saw a considerable improvement of available cpu-power and memory for MT in the past. This implies we cannot do extensive computations on MT (even if a powerful CPU and big memory is available), because a CPU under load consumes a lot of electricity. NS meet this limitation: they don't require extensive computations on the MT since the actual logic of the services is executed on the stationary servers of the backend system.

Display quality of MT: MT have a small display and thus can only display a limited amount of information at one time, the resolution and colour depth available are also limited.

Data input: MT don't have a real keyboard or a mouse, so it is cumbersome to enter data. NS meet this limitation since after the initial configuration of the service they do require a much data input. They also make use of context information so the user has to enter less data.

Short sessions: User sessions of mobile services are usually short; typical i-Mode-sessions for examples are shorter than two minutes, but there may be 10-20 sessions per day and user (Zobel, 2001; pp. 109). When using a MT people don't want to browse for a long time (like they do when using a personal computer) to find the information they need, because the user interface of MT is limited and users may be in a surrounding where they are distracted; also the batteries wouldn't stand longish sessions. NS meet the postulation for short sessions: the notifications are delivered as push messages (the MT may play a ring tone to notice the user), so the user has just to look at his MT and read the message.

The prevalent paradigm for mobile services seems to be the delivery of www-like pull documents in special markup languages (e.g. cHTML, WML, XHTML), see for example the services in the portals of the mobile network operators. In contrast to NS the user experience of such services is notable impaired by connection drop outs and automatic updates of personal context information are not possible; also pull-mode delivery of information seems not to be adequate for many mobile scenarios.

4 EXAMPLE SCENARIOS FOR NOTIFICATION SERVICES

4.1 Classification of Examples

In this section we will give some examples for mobile services from literature as well as from

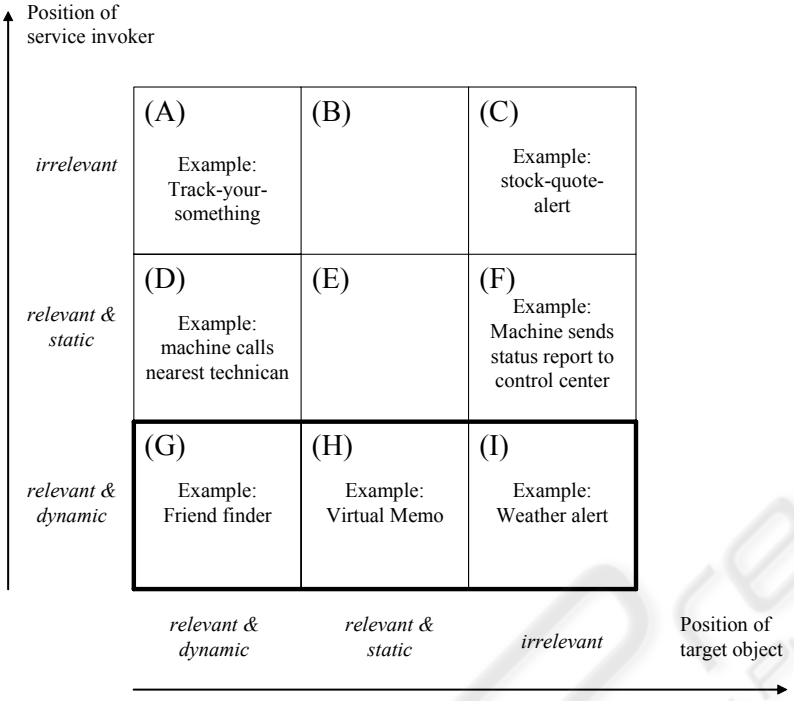

Figure 2: Classification schema for examples.

practice to demonstrate that despite their simplicity a wide range of useful services for mobile scenarios can be implemented as NS.

Since in most examples the current position of an object is relevant we use a classification schema similar to the one proposed by Turowski & Pousttchi (2004, pp. 78) to classify the example services (Figure 2). For this classification the "objects" — the user/executer who calls the service and an optional target object of the service — have to be put in one of the following three classes with regard to the relevance of their location:

- The position of an object is relevant for the service and dynamic and thus has to be determined, e.g. the current position of the user's MT with an attached GPS-receiver.
- The position of an object is relevant but doesn't change and thus can be looked up in some kind of database, e.g. the position of the nearest bus stop.
- The position of an object is irrelevant for the service or cannot be stated.

This schema results in $3 \times 3 = 9$ cases, the three cases where the position of the invoker is "relevant & dynamic" (G, H, I) are location based services. For each example service we name the required configuration data and the content of the notification. All NS-examples offer a mobility-specific value: the services provide time-critical information or cover information-needs that typically arise when the user is "on the move".

4.2 Examples

Case A: irrelevant/dynamic

The user is interested in the position of the target object, which might be a missing child or his car respective a MT used by the child or installed in the car. (Configuration: identifier of the target object; Notification: position of current location of target object.)

Case C: irrelevant/irrelevant

Neither the position of the invoker nor the target object is relevant in this case. The notifications in this case are triggered by time-critical events, e.g. significant changes of stock quotes. We call this class of services "alert services" (see Adya (2002) for other examples of mobile alert services and their popularity). (Configuration: identification code of stock quote, threshold values; Notification: identifier and current value of the stock-quote). Meanwhile there are even Business Intelligence (BI) systems that push notifications to MT if certain values (e.g.

available stock) are outside certain threshold values, see MIK (2006) for example.

Another example is the notification of performance after written exams at universities (gov.mt (2006)): some students are very eager to find out about their result as soon as possible, so they would like to use a NS that sends their result onto their MT (Configuration: the mapping of surname to matriculation number is supposed to be secret and thus can be used as authentication; Notification: course identifier and result.)

The conventional form of authentication for online banking is a password and a transaction number (TAN) for each transaction. During a phishing-attack a user enters these secrets into a web site he wrongly believes to be the website of his bank. To prevent this form of phishing a bank could implement a two factor authentication (Wüest, 2005) based on a mobile NS: after submission of the transactions details (account number & bank code of recipient, amount of money to be transferred) and TAN on a non-mobile computer the web page shows a challenge code. This challenge code has to be entered on a MT (configuration) which then will display the transaction details and a response code (=notification). To actual execute the transaction the response code has to be entered on the web page. A phisher had to compromise both — the fixed computer and the MT — to be successful which is much harder.

The basic principle of m-ticketing is to use MT for distribution and presentation of electronic tickets (Hussin et al., 2005); a tickets is something that certifies that the owner of the ticket has the right to claim a certain service (using public transportation vehicle or permission to entertainment events like cinema, concert). Electronic tickets usually have some kind of digital signature to hinder fraud (e.g. SMS-message which states the date and destination of a bus journey along with a certificate code). We can model m-ticketing-systems as notification system, if we consider the notification to contain the ticket. The configuration is the booking of the ticket (category of seat, number of zones in public transport).

Case D: static/dynamic & Case F: static/irrelevant

Since a NS is always invoked by the MT this case implies that the MT has a fixed position, which is only reasonable if it is embedded in a machine (e.g. vending-machine). We do not consider this case here although a machine calling the nearest service technician (case D) or reporting its status to a control center (case F) could be implemented as NS.

Case G: dynamic/dynamic

This service notifies a user, when another mobile user of the service classified as “friend” is away less than a certain defined distance, see the description of “FriendZone” by Burak & Sharon (2004). (Notification: Name/nickname of the “friend” that is within the defined distance; Configuration: identifier for the group of friends and own nickname.) If instead of a group identifier a profile (age, gender, fields of interest) is used for the configuration of the service we would obtain a “blind-date-finder”. (Notification: “someone who matches your profile is around”).

Other examples of NS for case G would be “Call a taxi” or “call an ambulance/police car”, the target object would be the nearest taxi or ambulance. If the NS only forwards the request to a control center where the choice which vehicle to send to the location of the user is made manually we get case I. (Configuration: type of vehicle requested; Notification: confirmation with optional estimated time to wait.)

Case H: dynamic/static

To all positions in a given area (certain city or whole country) the users of the service can assign virtual memos with a message like “don’t visit the restaurants at the corner of this street” (see also the *GeoNotes*-system by Persson et al., 2002). Notifications: If a user approaches a location with such a virtual memo he gets a notification with the message(s) of the virtual memo deposited by other users. (Configuration: range, message to deposit at current location, expiry date of message.)

Tourist guides are another example for NS in case H (see also Sampat et al., 2005): the user receives notifications with information concerning the sights and places in his surrounding as he wanders around in a district with places of interest.

The example of mobile advertising belongs also to this case and was mentioned in section 2.1; the target objects for this type of service are the shops of the advertisers.

Munson & Gupta (2002) mention the idea of a NS that notifies car drivers when they are approaching a highway section with a construction site.

Case I: dynamic/irrelevant

There are NS which send alerts to people when a thunderstorm or other menacing weather situation is supposed to approach their current location, e.g. to warn outdoor sportsmen (e.g. sailing, skiing). As target object in this case we consider the "weather"; since "weather" is a complex phenomenon, we cannot assign a position to it. A location aware weather alert service is provided by a German insurer (VBK's "Wind & Wetter"), see also Fraunhofer (2006). (Notification: warning about predicted change in the weather; Configuration: events of interest, current location.)

5 SUMMARY AND FUTURE WORK

We introduced notification services (NS) and argued that this class of services is suitable for mobile scenarios with regard to the technical restrictions and the user experience. A lot of examples showed that while simple NS cover a wide range of reasonable usecases. Using public key cryptography it is also possible to specify a simple protocol that guarantees that the mobile network operator cannot learn about the contents of the order and the actual service provider cannot learn about the identity of the user or send unsolicited messages.

The next step is the development of a software-framework for NS; a framework in this sense (Johnson & Foote, 1988) is a generic application for a given class of applications (in our case: NS). Using this framework a distinct NS could be implemented as extension of the framework. In contrast to a conventional software library a framework enforces the reuse of the whole design (not just of parts of the implementation, but also the architecture) and handles the control flow, so the service specific parts are called by the framework ("inversion of control"). Altogether frameworks are supposed to promote a quicker development of applications which are simpler to maintain.

REFERENCES

Adya, A., Bahl, P. & Qiu, L., 2002. Characterizing Alert and Browser Services for Mobile Clients. In Proceedings of the USENIX Annual Technical Conference.

Burak, A. & Sharon, T., 2004. Usage Patterns of FriendZone — Mobile Location-Based Community Services. In *Proceedings of the 3rd International Conference on Mobile and Ubiquitous Multimedia*, ACM.

Dey, A. K., 2001. Understanding and using Context. *Personal and Ubiquitous Computing Journal*, Vol. 1(5), pp. 4-7.

Dubinko, M., 2003. *XForms Essentials*. O'Reilly, Upper Saddle River, USA.

Fraunhofer, 2006. @ptus weather — Weather Information on Demand. Brochure of Fraunhofer ISST, Berlin, Germany.

Gov.met, 2006. Website of Government of Malta. http://www.mobile.gov.mt/services.asp?mb:lang=en, last accessed: 10.06.2006.

Hussin, W., Coulton, P. & Edwards, R., 2005. Mobile Ticketing System Employing TrustZone Technology. *Proceedings of the International Conference on Mobile Business (ICMB)*, Sydney, Australia, IEEE.

Johnson, R. & Foote, B., 1988. Designing Reusable Classes. *Journal of Object-Oriented Programming*, Vol. 1, pp. 23-35.

Kölmel, B. & Alexakis, S., 2002. Location Based Advertising. In *Proceedings of the 1st International Conference on Mobile Business (ICMB)*, Athens, Greece.

MIK, 2006. Information Any Time it's Needed — Mobile OLAP Reporting with MIK-BIS-Agent. Website of MIK: http://www.mik.de (30 November 2005).

Munson, J. P. & Gupta, V. K., 2002. Location-Based Notification as General-Purpose Service. In *Proceedings of the 2nd International Workshop on Mobile Commerce*, ACM, pp. 40-44.

Persson, P., Espinoza, F., Fagerberg, P., Sandin, A. & Cöster, R., 2002. *GeoNotes*: A Location-based Information System for Public Spaces. *Designing Information Spaces: The Social Navigation Approach*, Spring, pp. 151-173.

Satyanarayanan, M., 1996. Fundamental Challenges in Mobile Computing. In *Proceedings of the Symposium on the Principles of Distributed Computing*, Philadelphia, Pennsylvania, USA, ACM.

Sampat, M., Kumar, A., Prakash, A. & McCrickard, D. S., 2005. Increasing Understanding of a New Environment using Location-Based Notification Systems. In *Proceedings of the 11th International Conference on Human-Computer Interaction*, Las Vegas, NV.

Turowski, K. & Pousttchi, K., 2004. *Mobile Commerce* (in German). Springer, Berlin et al.

Wüest, C., 2005. Phishing in the Middle of the Stream — Today's Threats to Online Banking. *Proceedings of the 8th Association of Anti-Virus Asia Researchers Conference (AVAR)*, Tianjin, China.

Zobel, J., 2001. *Mobile Business und M-Commerce* (in German), Hanser Publishing, Munich, Germany.