COMMUNICATION PROFILES AND ACTIVE ATTRIBUTES

Peter Dornbusch, Michael Fahrmair
Munich University of Technology
Boltzmannstr. 3, D-85748 Garching, Germany

Eiman Mohyeldin, Markus Dillinger
Siemens AG

ABSTRACT
To realize personalized web services and to design adaptable systems it is necessary to have a possibility for modeling the user's environment and preferences. A common approach is the usage of profiles. We use communication profiles for modeling all relevant actors in a mobile scenario. The profiles which we have proposed [2] are separated into a static and a dynamic part. In this paper we suggest a new approach in designing dynamic attributes. In order to illustrate the working of these – so-called - active attributes we introduce a positioning service for heterogeneous networks.

KEYWORDS
Adaptable System, Active Attribute, Communication Profile, Positioning

1. INTRODUCTION
Software Defined Radio (SDR) will bring to the user advantages by supporting reconfiguration of terminal. The decision unit that controls the reconfiguration process needs to know information about the current context of the entire system (i.e. information about the main system entities like user, terminal, and network). This context information is clustered into profiles. The instances in the network need to control reconfiguration processes in the terminal and before reconfiguring the terminal, the capabilities of the terminal must be known and checked regarding the target mode of operation (e.g. RAT). Further, for mode negotiations the service, network and user profiles need to be retrieved before the decision process starts.

In [2] we have already given a general overview about profiles. But a few questions are still not answered. In particular it is still not clear how we can deal with dynamic attributes. In this paper we will describe how they work. Also we want to show how an attribute can be directly connected with a service. This will be done by introducing a positioning service for heterogeneous networks.

2. COMMUNICATION PROFILES
A communication profile is the abstraction of all relevant participants in a mobile scenario, as there are the device, its network environment, the user and the available services on different levels ranging from lower layer bearer services to higher level tele- and application services. Every entity or “actor”, which is a participant in a mobile system scenario has its own profile of capabilities and features that consists of a static and a dynamic part. The dynamic part contains a set of attributes that may change frequently and the static part attributes that stay fixed for a certain period of time. So may for example be the type of display (color/bw) of a terminal fixed, but the actual available amount of free memory may vary over time. Terms concerning communication profiles are defined first:
3. ACTIVE ATTRIBUTES

Our proposed profiles are distinguished in a static and dynamic part. In addition to this we want to introduce active attributes. Active Attributes are dynamic attributes but furthermore they can be connected directly with services which calculate the attribute value during runtime. In Figure 1 (on the left side) you can see an example of a “normal” attribute request which gets the value of an attribute which is stored in the database. But how we will explain later, for values which change continuously this is not the best way to update them. For example the user’s position is a frequently changing attribute. So the value of the attribute should only be updated if necessary, i.e. the attribute is requested.

In general there are two ways thinkable how attribute value can be updated: a time-based or an event-based update mechanism. Time-based means that there is a continuous update mechanism, which calculates and sets the dynamic values. In most cases this results in increasing traffic and ineffective resource usage. Nevertheless in some cases this is the only suitable solution. A better way of updating dynamic attributes is an event-based approach. Every time a value which is listed in a profile changes an event is initiated. This can also result in increasing traffic but it is still a more effective way of updating attributes. Another way of an event-based updating could be realized by updating an attribute whenever it is used. For example an attribute which describes the present position of the terminal could always be recalculated even if it is requested by a service. We suggest the connection of a dynamic attribute with a trigger, which starts a service that calculates the corresponding value. This is similar to the above mentioned event-based approach which updates the value whenever the attribute is used. Figure 1 (on the right side) shows the way how the active attributes work in opposite to static or dynamic attributes. A service requests two different active attributes A and B which are connected both to their own service that calculates in each case the value during runtime.

4. SAMPLE SERVICE: POSITIONING

In this section we want to present a sample for an active attribute. As mentioned in the beginning we will explain how a positioning service can be realize for heterogeneous network. Our positioning service is a cell-based method comparable to the COO [4] known form mobile networks like GSM. The cell id of the present cell where the end device is connected to is used to determine the position. In order to demonstrate how an active attribute works in a heterogeneous network environment we show a solution for WLAN and GSM. Normally the position can be calculated on the server – so-called server-based positioning – or on the client – so-called client-based positioning. In our approach for WLAN and GSM we use once the client-based positioning for WLAN and once the server-based positioning for the GSM networks. On one hand this is a
A realistic way how positioning is done in each of these networks and on the other hand it is a good motivation for initiation of active attributes.

A. WLAN Positioning

The first positioning method we want briefly explain is the WLAN positioning. In [3] you can find a good introduction of WLAN positioning methods. In our case the accuracy of the method is not the most important fact. We are more interested in an integration of methods for different networks. However the usage of another positioning method for WLAN will work with our system in the same way as the COO positioning which is here used. The service which runs on the client reads out the MAC address of the access point which is used to connect the Terminal to the network. A table of the access point positions is required for calculating the position of the end device. In our architecture the service has to register itself at the profile registry whenever it is available. But the POSITION attribute should not be updated if possible. The reason for this is that we want a minimum of traffic on the air interface.

B. GSM Positioning

For realisation of the positioning for GSM network we use the Siemens Location-Enabling Server (LES) [5] which includes a simulation tool of a location server. The location server which collects the position relevant data calculates the positions and stores the position data in a database is connected by HTTP to the internet. In our case we only use the simulated location server. So the position data is faked. For our purpose this is not a big problem because we only need a realistic interface to GSM positioning data.

A service which uses position information can make a request by sending a XML message via HTTP to the server. The response is also a XML message and consists of longitude, latitude and an error value if the request was successful. As you can see the interface is very generic and is independent form a special positioning method. Actually in GSM networks is only a cell-based method available so far. Form our point of view the positioning method itself is not the fundamental aspect. Interesting is here only the usage of a server-based method which is normally difficult to combine with client-based.

C. Integration

Now we have described how the locating of a terminal in WLAN and GSM works. But how can we integrate these both approaches in our system? Therefore we use an attribute POSITION which is stored in the network profile. Obviously it is a dynamic attribute. First of all we propose that the usage of the WLAN positioning should be the preferred method because it is presumed that WLAN positioning should be the more exact method for the next years. Additionally a client based WLAN positioning can be realized as a free service without cost for the user. The WLAN Positioning service is running continuously on the client and tries to calculate the present position if possible. This means the WLAN service can only determine the location if a WLAN is available and when there is additionally a database entry for the cell id (in this case MAC address). If a position calculation is possible the service sends a message to the profile storage to activate the usage of the WLAN positioning. The GSM positioning should be always available and active but only be used if necessary. Normally the network operator locates a mobile terminal always (with cell accuracy) when the device is turned on, because it is necessary to know the position of a terminal to establish a connection. The fact where the POSITION attribute is actually stored - on the terminal or on a server – is not important because the registry handles the profile/attribute distribution. Figure 2 (on the right side) shows the architecture of the whole system for our sample service.
If a service needs now the position of the user/terminal it sends a POSITION request to the registry. The request also triggers a request to the “best” positioning service which means a request for the position with the most accurate and most recent value. In general the WLAN positioning is more accurate than the GSM method at this time. So the WLAN positioning should be selected if possible. If the WLAN is not accessible because there is no access point the GSM positioning is automatically chosen. In the case the WLAN is available but the position cannot be calculated because of a missing database entry the WLAN positioning returns an error response. To get now the proper position a GSM request is also needed.

5. CONCLUSIONS

In this paper we have given an introduction of communication profiles and their structure. We have shown the separation of profiles in a static and a dynamic part. Additionally we have shown how to deal with the connection of attributes and services. For the future work, we will try to find out, how frequent the feature changes will be tackled, that will allow for more sophisticated and efficient optimisations strategies.

REFERENCES

[4] Digital cellular telecommunications system (Phase 2+), Location Services (LCS), (Functional description) - Stage 2 (GSM 03.71 version 8.0.0 Release 1999)