LoL@: designing a location based UMTS application

UMTS combines mobile communications and the Internet. After showing some fundamental issues of UMTS services, this paper presents LoL@, the Local Location Assistant. It is a prototype of a location based UMTS service, combining localization/navigation and multimedia. These components can be seen as "killer components", they can be combined to provide various services over various networks.

Keywords: mobile Internet; location based services; SIP; mobile multimedia

LoL@: Entwicklung einer positionsbezogenen UMTS-Anwendung.

UMTS fasst Mobilkommunikation und Internet zusammen. Dieser Artikel behandelt zunächst die Grundlagen für UMTS-Dienste, danach wird LoL@ vorgestellt. Lol@ ist ein prototypischer UMTS-Dienst und kombiniert Positionsbestimmung/Navigation mit Multimedia-Information. Diese Komponenten können auch für andere Dienste in verschiedenen Netzen verwendet werden.

Schlüsselwörter: mobiles Internet; positionsbezogene Dienste; SIP; mobile Multimedialinformation

H. ANEgg, H. KUNczier, E. Michlmayr, G. Pospischil, M. UmlaufT

1. Introduction

UMTS will only be successful if it provides a portfolio of attractive services. There will not be a single "killer application", people's needs vary too much to be satisfied by a single application. Therefore, UMTS standardization has defined a framework to develop a rich set of services. It consists of a concept for service access (Virtual Home Environment TS 22.121, TS 23.127), a network architecture (Open Service Access TS 22.127, TS 29.198), and a terminal architecture (Mobile Execution Environment TS 23.057).

LoL@, the Local Location Assistant, is a prototype of a location based service. It is designed for tourists who walk on a sight-seeing tour through the first district of Vienna. LoL@ guides them along the tour and provides multimedia information related to the sights.

In Sect. 2, the UMTS service architecture is shown, followed by the LoL@ architecture components in Sect. 3. The paper ends with summary and conclusions in Sect. 4.

2. UMTS service architecture

In GSM, the available services, e.g. voice, fax, data, are standardized. This ensures compatibility between different networks and terminals, but it impedes development of new services. The introduction of SAT and WAP was the first step towards an open service environment. However, both concepts are not sufficient for complex UMTS services because they are designed for very limited GSM phones and do not provide access to all relevant network elements, like the user profile server (HSS) or multimedia subsystem.

To overcome the current limitations, a flexible service environment for UMTS has been standardized. The main goals are to facilitate quick service development and convenient service access. Figure 1 shows how these goals can be achieved via the virtual home environment (VHE), open service access (OSA), and mobile execution environment (MExE).

VHE is a concept for service access with different devices (e.g. Smart Phone, PDA or Laptop) over different networks (GSM/GPRS, UMTS, wireless LAN). In all situations, the functionality and look & feel shall remain the same as far as possible. To realize VHE in mobile networks, MExE and OSA are used.
services, like data conversion, subscriber and content management, or access to content databases. Parts of the business logic are often implemented as servlets on top of a Web-Server.

In the Internet, the transport network is usually seen only as a bit-pipe. However, mobile networks offer additional functionality, e.g. call control, localization, user/device profile access. An abstraction of such a network functionality is called Service Capability Feature (SCF), provided by a Service Capability Server (SCS). OSA consists of a framework for authorization, authentication and discovery of network features and provides a standardized interface to SCFs. With OSA, network services can be included into an application in the same manner as conventional platform services by using a set of CORBA functions.

3. LoL@ service architecture

Figure 2 shows the components of LoL@. A typical service request starts in the service user interface (GUI component of the terminal, see Sect. 3.1). After an interaction with the client part of the business logic (Terminal Core), the request traverses the UMTS/GPRS network. Finally it reaches the destination server (usually the LoL@ Core), which processes the request (Sect. 3.2). It may contact other servers of an application/content provider or the mobile network operator. Finally, a response is sent to the terminal.

LoL@ is designed for a PDA-like MExE phone with a 320 x 120 pixel color LCD display and pen input. Currently, the device is simulated on a laptop with a Web-Browser and JAVA Virtual Machine, extended by a mobile termination for network access.

The terminal provides the following features:

1. Service user interface (GUI), including a JAVA applet for displaying maps, a HTML window for hypertext information, and JAVA AWT buttons for service control;
2. The client part of the business logic (Core), implemented as a (hidden) JAVA applet. In traditional Web-Applications, the business logic resides completely in the server. We decided to put a part of it into the terminal to improve response time and minimize air interface traffic;
3. SIP User Agent for packet switched voice (Voice over IP) sessions, either for human-to-human communication or human-to-machine communication (speech control);
4. Terminal LCS, providing connectivity to external localization devices, like a GPS receiver or Bluetooth transceiver, and the location server (LCS, see Sect. 3.3).

The server domain hosts the application platform and the supporting servers. Our application platform is a Web-Server with several LoL@ servlets; we do not use a specific application server. The following network related functions are implemented as Service Capability Features (SCF):

1. Call Control SCF, using a SIP Proxy for application level call control.
2. Localization SCF, used to provide Localization Services (LCS). It connects to the terminal counterpart (to retrieve e.g. GPS coordinates). This data is combined with network related information, e.g. cell site database or network based location information. Applications may use the LCS infrastructure via the OSA/Parlay Mobility Interface.
3. User Profile SCF, used for authorization of location requests and user identification. We use an LDAP database for this purpose; in an operational network this SCF is the HSS (Home Subscriber Register).

3.1 LoL@ user interface

LoL@ will be used outdoors on a PDA-like device. Therefore, environmental problems, device constraints and special user needs pose additional challenges. Consider, for example, low LCD contrast in direct sunlight or the uncomfortable and small keyboard. Additionally, people are on the road and might be nervous because they are lost in a foreign city, or pressed for time (e.g. they want to find a tourist attraction before it closes).

We consider the following four interaction scenarios to be the most important ones:

1. A user is in the hotel and wants to preview the tour or get information about sights.
2. A user is on the tour and wants information for the current point of interest (Pol).
3. A user wants to be guided to the next Pol.
4. A user is lost and needs guidance (routing).

Our application is map-centric; i.e. we use a map-metaphor with additional textual information screens. In the map we use "select, then apply command" interactions: a user clicks on a Pol, then on the "info" button to get information for this Pol. A browser-metaphor is used to navigate through hypertext information. To avoid user distraction, the maximum depth of the menu structure is restricted to 5. In addition to pen and limited keyboard input we use server-side speech recognition, based on SIP-controlled Voice-over-IP sessions. LoL@’s speech control provides a set of commands which are used as "short-cuts" to access often used menu items.

A sample usage scenario might look like this: When the user starts LoL@, user information is retrieved from the HSS and the user is logged on automatically (the user does not have to enter...