

# Evolution of Mobile Services: An analysis of current architectures with prospect to future

Ivar Jørstad<sup>1</sup>, Schahram Dustdar<sup>2</sup> and Do van Thanh<sup>3</sup>

<sup>1</sup> Norwegian University of Science and Technology, Dept of Telematics, O.S. Bragstads  
plass 2E N-7491 Trondheim, Norway

[ivar@ongx.org](mailto:ivar@ongx.org)

<sup>2</sup> Vienna University of Technology, Distributed Systems Group (DSG), Information Sys-  
tems Institute A-1040 Wien, Argentinierstrasse 8/184-1, Austria

[dustdar@infosys.tuwien.ac.at](mailto:dustdar@infosys.tuwien.ac.at)

<http://www.infosys.tuwien.ac.at/Staff/sd/>

<sup>3</sup> Telenor R&D, Snarøyveien 30 N-1331 Fornebu, Norway

[thanh-van.do@telenor.com](mailto:thanh-van.do@telenor.com) <http://www.item.ntnu.no/~thanhvan>

**Abstract.** With the advent of the Internet and the plurality and variety of fancy applications it brought with it, the demand for more advanced services on cellular phones is increasingly becoming urgent. Unfortunately, so far the introduction of new enabling technologies did not succeed in boosting new services. The adoption of Internet services has shown to be more difficult due to the difference between the Internet and the mobile telecommunication system. The goal of this paper is to examine the characteristics of the mobile system and to clarify the constraints that are imposed on existing mobile services. The paper will also investigate successively the enabling technologies and the improvements they brought. Most importantly, the paper will identify their limitations and capture the fundamental requirements for future mobile service architectures namely openness, separation of service logic and content, multi-domain services, personalisation, Personal Area Network (PAN)-based services and collaborative services.

## 1 Introduction

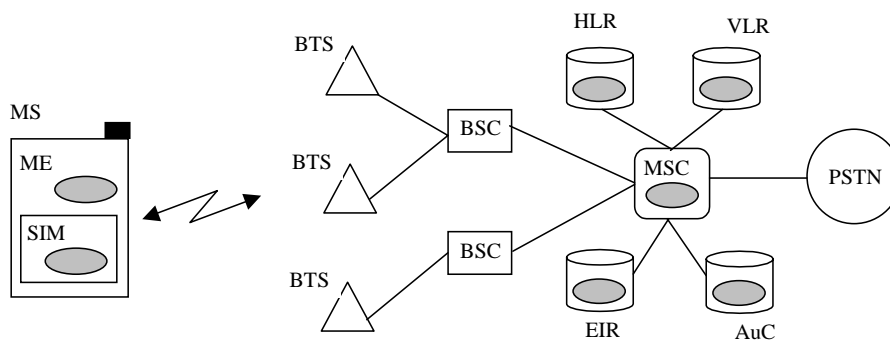
With digitalisation, the difference between telecommunication and computer networking is fading and the same technologies are used in both fields. However, the convergence does not progress as rapidly as expected. Moving applications and services from one field to the other has proven to be very difficult or in many cases impossible. The explanation is that although the technologies in use are rather similar there are crucial differences in architecture and concepts. The paper starts with a study of how mobile services are implemented in mobile telecommunication systems and an identification of their limitations.

## 2 Analysis of current mobile service architectures

### 2.1 Voice communication

As indicated by its name, the objective of mobile telecommunications systems is to provide communication between mobile distant persons. In Europe, Nordic Mobile Telephony (NMT) became available in the 1980s. In 1982, development of the GSM system started. Initially, these systems only supported direct voice communication or telephony between two participants, but supplementary services like call forwarding, barring and voice mail were added later on.

Figure 1 shows that the mobile telephony service is realised by components represented by grey ovals that are distributed both on the mobile phone, also called Mobile Station (MS), and on the mobile network. On the MS, there are components both on the Mobile Equipment (ME) and on the Subscriber Identity Module (SIM).



**Fig. 1.** Telephony service components in mobile communications system

To establish a telephone conversation the service components on the MS are collaborating with the ones on the mobile network to allocate a channel and to maintain it throughout the session even when the MS is moving and changing base stations.

Thanks to the clearly defined interfaces between them, the components can be designed and implemented by different parties. The components on the mobile phone are installed by the manufacturer while the ones on the network are delivered by network equipment suppliers.

As observed above, the telephony service is implemented in a very robust way and shall be operative 99,999% of the time. On the other hand, the architecture is also very rigid and is not favourable for the introduction of new services.

## 2.2 Supplementary services with IN

It does not take long time before there is a need for more advanced call control services like call forwarding, barring, voice mail, premium call, etc. As shown in Figure 2 an IN (Intelligent Network [1]) Service Control Point (SCP) is introduced in the mobile network to allow the implementation of supplementary services. It is worth mentioning that these services are derivatives centred around the voice communication service. Another restriction is that the SCP is implemented on equipment manufacturer proprietary technologies. The SCP is also located inside the telecom operator domain making third party service development difficult.

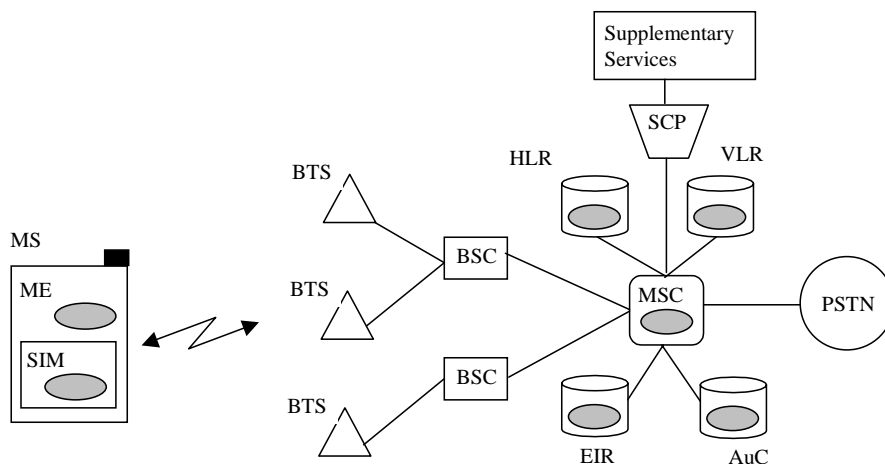


Fig. 2. The implementation of supplementary services

## 2.3 Enabling services on the SIM with SIM Application Toolkit (SAT)

The telecom operators want to have other services than telephony and its derivatives and turn to the SIM, which are their property. Unfortunately, although the SIM is a smart card having both processing and storage capabilities necessary for new services, it is useless due to the lack of interfaces with input unit (keypad, microphone) and output units (display, loudspeaker). The SIM is supposed to be the slave executing orders from its master, the ME. To remedy this, the SIM Application Toolkit (SAT) [2] is introduced to allow applications/services residing on the SIM to control the input and output units.

With SAT it is possible to develop applications on the SIM but there are many restrictions (see Figure 3). First, SAT applications should be small in size and developers must have access to SIM application development environment, which is both difficult and costly. Second, the installation of applications on the SIM is controlled by operators who are reluctant to open the access due to security. The results are that

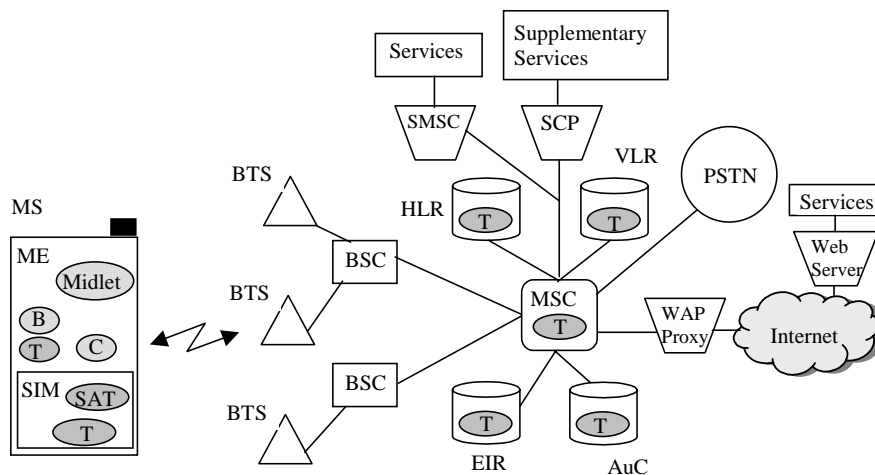
SAT applications are usually operator-owned and are typically security related since the SIM is a tamper-resistant device.

Recently, the JAVA SIM cards start to emerge and it will be very interesting to have collaboration between SIM JAVA components and JAVA components on the Mobile Equipment enabled by J2ME.

## 2.4 Text services with Short Message Service (SMS)

Although voice communication is a big success there is still a demand for sending text messages from one mobile phone to another. An SMS client was introduced in the ME and responsible for sending short messages to the Short Message Service Center (SMS-C). The SMS-C is responsible to store and forward messages to and from mobile phone (See Figure 3). In the illustration, components used for SMS are the client (C) in the ME, the SIM (for storage) and services connected to the SMS-C in the network.

SMS substantially increased the value of mobile telecommunication systems by providing an alternative, more informal, way of communication between customers.



**Fig. 3.** Current mobile services including SAT applications, SMS-based services, WAP services and J2ME applications

Not long after its introduction, SMS was seen as a suitable technology for providing a lot of other value added services by delivering specialised content requested by users. Development of SMS services can be performed by anyone with a little bit of programming experience. Most services can actually be implemented as Perl scripts or with any other programming language capable of reading data from standard input and producing some output.

For developers and potential service providers there exist tools for testing and deploying SMS services, as well as completely free open source SMS Gateways. Such

SMS Gateways are where the actual services are “plugged” in, e.g. as a Perl script as mentioned above.

Provisioning of SMS services requires installation of the above mentioned application on an SMS Gateway that either communicates directly with an SMSC using one or several protocols defined for this purpose (e.g. CIMD [3] or SMPP [4]). Another solution is for the system running the SMS Gateway to act as an SMSC itself (e.g. a PC using a radio modem through a serial port). To have direct access to an SMSC requires cooperation with the operator that owns the SMSC, which often can provide a TCP connection for sending/receiving SMS messages part of a service. The advantage of the second solution is that such cooperation with an operator enables the owner of the SMS Gateway to receive revenue from generated traffic. In Norway, such an agreement between an SMS service provider and an operator is available through a Content Provider Access (CPA) agreement.

Anyone with an SMS enabled handset (all GSM handsets today) can access services where the service access number is known (mostly a 4 digit number). In order to multiplex many services onto one such service access number a complete service request is created by appending additional service identifiers, and input information for the specific service, to the service access number. Complete service descriptors are thus created in a hierarchical way.

The problem with access to SMS services is remembering both the service access number and the additional identifiers and parameters for a specific service (the protocol). This is where the greatest limitation of SMS is; for an arbitrary service with a lot of input/output requirements, SMS is unfeasible to use and a more user-friendly approach must be taken. Nevertheless, it has been shown that a lot of high quality services can be based around simple textual content.

For the future, the interest in SMS lies in the combination of it with other, dynamic application platforms as J2ME or native applications of the OS, but also in the combination with other media; a lot of television channels have seen the revenue potential in SMS for interaction between viewers and TV shows where it is often used for voting purposes.

## **2.5 Internet access with WAP**

Wireless Application Protocol (WAP) [5] was initially an attempt to provide access to the WWW on handheld terminals. This concept is often referred to as the Mobile Internet. In concordance to the World Wide Web, a micro browser installed in the Mobile Equipment is communicating with a WAP Proxy introduced between the Internet and the mobile network to convert Internet protocols to Wireless binary protocols as shown in Figure 3. On the terminal side, a WAP browser is located in the ME (symbolised by an oval with a B inside) and services are connected to a Web server on the network side.

Similar to SMS, development of WAP services can be performed by anyone with some programming experience. Most services typically consist of some static WML content together with a CGI-script as back-end that can generate dynamic content retrieved from for example other Web sites or from a DBMS. These back-ends can

also, as with SMS, be developed in Perl or any other CGI capable language. The WAP programming model [6] is very similar to WWW, except for the restricted set of mark-up tags available. The basic markup language for WAP 2.0 is XHTMLMP, which extends the Basic profile of XHTML as defined by W3C [7]. The realisation of WAP 2.0 is radically changed from the previous standards, because a WAP proxy is no longer needed between the handset and the server; WAP 2.0 supports the standard Internet protocols, so a conversion between WAP protocols and Internet protocols is no longer needed.

There exists a lot of WAP Software Development Kits (SDK) available for developers. Both Ericsson and Nokia provide their own solutions.

The only requirement for deploying and provisioning WAP services is a HTTP server that is accessible from the Internet and that supports the content types (MIME types [8]) required by WAP (most importantly text/vnd.wap.wml and text/vnd.wap.wmlscript).

However, to receive revenue from WAP services, a model similar to commercial SMS services is often used. In Norway, an agreement called WAP CPA is used, and it makes WAP service requests pass through the operator before the service is rendered to the user. This way, the operator is able to charge the user, and the service provider receives a cut of this charge.

WAP services can most efficiently be accessed through portals. Most mobile telecom operators have their own portal which is automatically configured together with other WAP settings on the handset. For access to other services, a URL must be manually entered in the WAP browser. Most handsets allow the user to store bookmarks of URLs, thus providing easy subsequent access to services. However, subsequent access to personalised services is not easy with WAP, because the technology has had poor support for sessions. This makes services requiring user authentication/login unfeasible to use; the username/password must be entered every time the service is accessed.

One restriction of the technology is that it is not possible to access ordinary web-pages using a WAP browser. Instead, services must be designed and implemented specifically for WAP. Although the new WAP 2.0 standard can render documents written in XHTML Basic, many devices do not yet support the WAP 2.0 standard and most services on the WWW do not conform to the XHTML Basic language. Thus, providing services to WAP almost always requires a re-implementation of already existing services.

A big advantage of WAP is that anybody can provide services, although billable services are not possible to provide without a special agreement with a telecom operator. The fact that anybody can provide services is an advantage in comparison to SMS, where service development and delivery is restricted.

Disregarding WMLScript, WAP is purely based on content, and since this content will be located in the Internet it will be accessible by anyone with an Internet connection; a handheld device with WAP browser is not required. The content can for instance be accessed from Internet browsers on a PC (Opera does support WML, although limited) or a WAP emulator. From this viewpoint, WAP currently seems like the most versatile platform for development of mobile services, supporting terminal independence.

Browsing for a particular service of interest is not satisfactory way to access services on a small, handheld terminal. A more definite and intelligent approach is needed; services on mobile terminals need to be more “at-hand” than on stationary computers. This difference in requirements for accessibility follows naturally from the different contexts the two terminals are used in; mobile terminals are often used while in transit between two locations, thus minimal attention should be required for using the services. One solution might be to provide better search engines designed for WAP services, potentially exploiting emerging Semantic Web [9] technology.

## **2.6 Dynamic applications on mobile phones with J2ME (CLDC/MIDP)**

So far, the functionality of the mobile phones is defined at manufacture time and it is not possible to install new applications. Indeed, the mobile phone is a terminal, i.e. a device terminating the network system and not a computer allowing the selection and installation of applications. It is desirable that mobile phones evolve to be closer to computers but at the same time retain the reliability and robustness of terminals. With introduction of the J2ME CLDC/MIDP platform, development of unlimited types of applications has become possible. A vast amount of such applications, called MIDlets, can be found on the Internet, some for free and some to be purchased. In Figure 3, these applications are represented in the ME as an oval with the text “Midlet” inside.

J2ME CLDC/MIDP [10] is a runtime environment for small footprint Java applications, targeted at handheld terminals with limited resources (processing capacity, memory etc.). With J2ME, it is possible to develop dynamic standalone applications, but also clients that are part of a larger distributed system. This means that most of the services provided by the earlier discussed platforms, in theory, can be developed using J2ME as well, provided there exists resources for the developer to exploit; most importantly implementations of standard communication protocols (e.g. TCP, HTTP, SOAP).

A WAP client, or any other type of browser, can potentially be developed in J2ME. Also, specific services available through WAP can be developed on a per service basis, often with a more flexible user interface. These services can be developed using the HTTP protocol which is mandatory in any implementation of the J2ME CLDC/MIDP 1.0 platform.

When it comes to SMS, there are still some restrictions in J2ME and its Wireless Messaging API (WMA) [11]. Access to the standard inbox for SMS messages on a handset is not allowed, although this is actually the really interesting part with an SMS API. The WMA instead provides an API for sending specialised SMS messages between J2ME MIDlets. The lack of good SMS APIs, as well as Telephony APIs, as an integrated feature of J2ME, is a sign of the industry trying to keep control of these features within networks so they don’t risk suddenly finding themselves outside the value-chain. But it should be emphasised that opening access to these features has the potential to generate more revenue, faster, than what we have seen until now. With an SMS API providing access to the SMS inbox of a handset, it would be possible to provide services like:

- Incoming SMS filtering (spam filter)
- Incoming SMS forwarding (e.g. to an email address or other storage)
- SMS Auto reply (e.g. away messages)

The way the SMS APIs are built today they only provide SMS as a transport mechanism for other services to use. With a more open and dynamic API, it could be used to enhance SMS as a service platform itself, which operators would probably be better off with. It could have the potential for prolonging the life of SMS. Particularly, sending SMS from a non-J2ME/WMA enabled handset to a MIDlet providing a service on a J2ME & WMA enabled handset is thus not possible. This is a pity, because with this opportunity, anyone with a J2ME/WMA enabled handset could easily act as a service provider by installing such a MIDlet on the handset. From an operator perspective, this should be interesting as it has the potential to generate a lot of traffic by allowing early technology adopters to interact in services with people without the newest and most fancy handsets.

Most Integrated Development Environments (IDE) for Java allows development of J2ME applications as well. It is only required either to install Sun's J2ME Wireless Toolkit or a more specialised toolkit from a handset manufacturer (e.g. Nokia Developer's Suite 2.0 for J2ME ). The important things to install are the J2ME libraries as well as an emulator for testing services, and to ensure compatibility with most handsets it is recommended to stay with Sun's Wireless Toolkit.

Over-the-air provisioning of J2ME MIDlets was not a part of the MIDP 1.0 specification. A supplementary document describing a "Recommended practice" was released after the standard was completed. The MIDP 2.0 standard includes an updated version of this document. Most likely because of the late arrival of this OTA specification, handset manufacturers defined their own ways of performing OTA, thus making it more difficult to implement a standard J2ME OTA server supporting any J2ME MIDP 1.0 compliant device.

If we ignore the existence of different standards for J2ME MIDlet OTA provisioning, it is possible for anyone with a permanent Internet connection and an official IP address/accessible server to provision J2ME MIDlets. It is mostly an issue of configuring a standard HTTP server (e.g. Apache) and deploying descriptors of the applications in an XML document, together with the appropriate JAD and JAR file(s). One of the authors of this paper has developed a solution using Perl (as CGI) for easily deploying J2ME MIDlets through a WWW interface, and the script counts around 200 lines including DBMS interactions for keeping track of available services. In addition to OTA provisioning, owners of J2ME capable handsets can transfer applications from their PC/laptop to the handset using IrDA or other communication features available on the specific handset.

Although J2ME is a standardised technology, performed through the Java Community Process, the "write-once-run-anywhere" concept is not valid for this platform. Some terminal manufacturers have adopted the platform and added their own touch to it, especially in the presentation layer (Graphical User Interface, GUI). Motorola has developed its own library for GUI construction called the Lightweight Windowing Toolkit (LWT) which is distributed for free. Although enhancing the potential of J2ME, it ruins the concept behind, namely that the same services should be available using any J2ME CLDC/MIDP compliant device.



For personalised services requiring login, J2ME differs positively from WAP. J2ME CLDC/MIDP includes a small DBMS called the Record Management Store (RMS) which allows a database per MIDlet. This database, together with a corresponding database on the server side, can be used by services to allow long lasting sessions (by providing storage for user identifiers and authentication data).

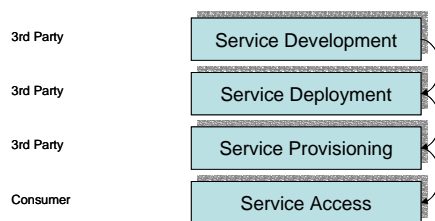
The strength of J2ME, however, will come when extended support for APIs becomes available. For example APIs telecom specific functionality (call control and SMS), extended support for communication protocols (TCP, XML Web services) and for accessing other integrated features of the handset (e.g. Bluetooth, cameras, storage etc.).

### 3 Future mobile service requirements

As discussed in earlier sections, the enabling technologies aiming at promoting data services do have specific focuses and limitations. Furthermore, they are either fragmented or overlapping and put together they are not sufficient to fill all the holes in the picture of the future mobile service arena. This section aims at identifying and elucidating these missing pieces and hence contribute to the definition of future mobile service architectures and concepts.

#### 3.1 Service openness

Many unresolved issues impacting the future of mobile services, are related to openness of specific systems. Today, the chain of activities between developer, provider and consumer are still characterized by a closed system where the terminal manufacturers and telecom operators are controlling everything and focusing on not losing control of the value-chain. A potential growth of services, as well as revenue generated by them, might be realised by allowing anyone to take on any of the roles behind the four specified activities, (see Figure 4).



**Fig. 4.** Openness in all activities

The activity, which is currently most open is the development of services. Both operators and 3rd party can today successfully create some types of new services. What is most important to further support this in the future, is that APIs of the service platforms are open, specifications of them are readily available and good Software Development Kits (SDK) are freely available.

Some service platforms do not require a developer to cooperate with operators to deploy services (e.g. WAP and J2ME) and some do (e.g. SMS). Installation of WAP and J2ME applications can be performed by anyone. SMS requires specific agreements with operators to access SMS-Cs through TCP or X.25 connections.

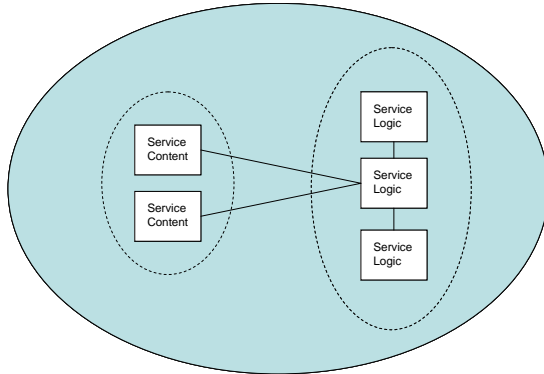
It is important that future service platforms are as open as possible for each of the mentioned activities. Still, they should include mechanisms for receiving revenue from service usage. It is difficult to reach compromise for this matter and further studies should be carried out.

### 3.2 Separation of service content and logic

Mobility is the ultimate requirement for mobile services; they should by definition be available at any time, any place using any device with communication capabilities. The mobility properties of a service are dependent on the architecture used to realise the service, and particularly on the *location* of the components making up the service. Considering a service as consisting of two components, *service logic* and *service content*, makes the analysis easier (see Figure 5).

The primary question is where the service logic should be located. In early mobile telecom services (Voice telephony and SMS), we have seen that the service logic was embedded in the dedicated hardware components of the system. This has been a hindrance for development of flexible services, but more importantly, it means that these services will by default not be accessible from outside an operator domain. Although interoperability between operators has been achieved by roaming features of the mobile telecom systems, the services are still only accessible from specialised terminals, i.e., cellular phones with a particular radio interface (e.g. GSM). Mobility means also that a service should be accessible by any device. To enhance the mobility of services, it is necessary to decouple the service logic from the system components.

Second, users will have an increasing need for accessing content that is related to a service; content that is a product of earlier service usage (e.g. content of an address book or a document created in a word processor). It is thus not enough that the service logic is accessible from any device, on any network. Content generated by the user must be accessible also. This raises the question of where this content should be located, if it should be replicated throughout service domains and networks or if it should be centralised in the home network of a user. Probably the easiest illustration of this challenge is the bookmark list a user keeps in an Internet browser. Today, the browser on each terminal keeps its own bookmark list, instead of providing access to the same content everywhere. This suggests that future service platforms must also cope with accessibility of service content and not only of the service logic. Alternatively, the service content can be incorporated in the user profile.



**Fig. 5.** Generic service composition.

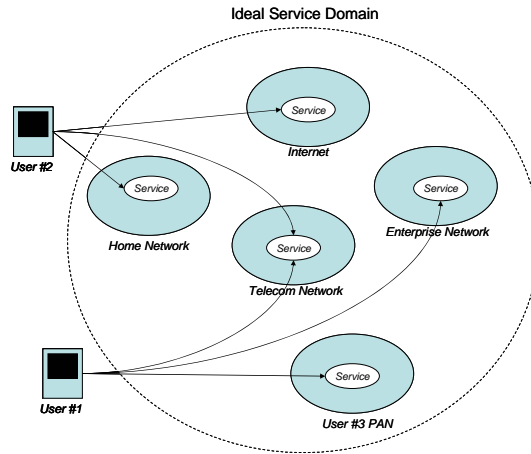
### 3.3 Multi-domain services

As mentioned earlier all services initially provided by mobile telecom operators were located in the telecom network itself, and developed by telecom operators. With SMS and WAP, many services were moved to a third party service provider, outside the operator network domain, providing access through CPA agreements or similar. Also solutions for accessing enterprise network domains and their services have been introduced (for example Microsoft Exchange with Mobile Features).

A complete service should embrace service elements belonging to all the domains as shown in Figure 6. The concept of a Virtual Home Environment proposed by 3GPP for third generation mobile systems allows a user to roam to foreign networks while keeping access to all services that they subscribe to in the home network. Unfortunately, the availability is restricted only to mobile telecom services. Taking this concept further, the ultimate goal would be to provide ubiquitous access to data services as well no matter where the service content and service logic are located.

As shown in Figure 6, one important domain is the Home Network of the user. In fact, more and more people get connection to the Internet at home through a permanent, broadband connection (xDSL, Cable or similar). Some of these have their own LAN at home. This home network constitutes an important host for personalised services. To offer a more complete service to the consumers, it is important to investigate solutions providing access to services located in the home network.

Ultimately, mobile services will be provided as distributed services where the logics residing in different places will cooperate in delivering the end-user service; i.e. composite services of disjoint service domains.



**Fig. 6.** An ideal composite service spanning over several domains

### 3.4 Personalisation of Services

Personalisation has for many years been regarded as an important feature of mobile services. However, a clear picture of what personalisation means, and of exactly what should be subject to personalisation, is lacking. A psychology study [12] defines two categories of motivation for personalisation:

- To facilitate work
- To accommodate social requirements

Until now, most personalisation efforts of mobile services have been to accommodate social requirements, whereas a big potential exists for services of the first category.

Personalisation can be performed at several levels:

1. Individual Service Personalisation
2. Personalisation of Service Portfolio
3. Personalised Service Composition

Individual service personalisation is the common way of thinking personalisation. It means changing parameters of a single service to accommodate a specific users needs. Such parameters can be either *passive* (look and feel of the service), *active* (behavior of the service) or *content related* (user added content, result of service usage). The user-defined values of the personalisation parameters are captured in the user profile that can be distributed or replicated in all the five domains: Telecom Networks, Internet, Enterprise, Home and PAN. Little work has been done on the distribution and organisation of the user profile and they will be subject to future studies.

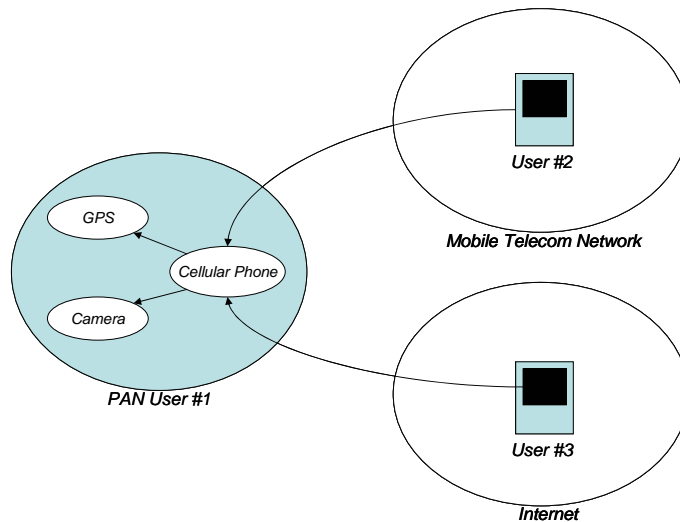
Personalisation of service portfolio means that a user makes certain services easily available through a portal or menu system. Such personalisation is done on most PCs,

where different users have different services/applications available through the desktop environment (or Quick Launch Bar in MS Windows).

Personalised service composition means that users can “create” their own services by picking discrete service components and combining them into a complete service. This is a novel concept that needs further investigation.

### 3.5 PAN-based Services

Nowadays, each individual is using several devices like mobile phones, PDAs, digital camera, GPS, etc. that are autonomous and functioning independently of each other and without any coordination. In fact they are not even aware of the presence of other devices. As the owner the user is required to handle them all and does not always succeed since as a human being he cannot perform many tasks at the same time. With the emergence of wireless short-range technologies like Bluetooth, WLAN and potentially UWB (Ultra Wide Band), Personal Area Networks can be formed to allow communications between devices. The Virtual Device [13] is a novel concept, which considers all the autonomous devices on the user's Personal Area Network as one big "Virtual Device" having multiple input and output units and providing a coherent and surround interface to the user. The Virtual Device concept paves the way for innovative and exciting services. An example is shown in Figure 7.



**Fig. 7** Providing access to resources in a Personal Area Network.

The User #1's Virtual Device consists of a cellular phone, a digital camera and a GPS receiver. Since it has multiple inputs and outputs multiple services can be executed simultaneously. The first one may supply User #1's location information to User #2 and the second one may show real-time pictures for User #3.

Every PAN becomes a service domain and potential host for service logic and service content as discussed in section 3.3.

### 3.6 Collaborative Services

The concept described in the previous section and illustrated in Figure 7 is particularly important for collaborative systems. With a multi-domain service, it will be possible for people not only to collaborate across network boundaries, but also across terminal boundaries. An employee sitting at home can for example work on the same document that other employees are working on at the office, while another employee on travel can work on the same document using his handheld terminal, be it a GSM cellular phone or a PDA connected through WLAN. This will not be restricted to documents, but to any resource that is sharable through communication technology (e.g. Bluetooth enabled devices). It should also be possible for several people to collaborate by exchanging information through several channels and devices simultaneously such as talking on the phones, showing picture on digital cameras, reading documents on PDAs, etc. In [14] we argue that Web services are a suitable technology for achieving service interoperability and suggest an architecture including service configuration points. The layered architecture depicted in Figure 8 suggests new layers of abstractions for management and configuration of services as well as their compositions.

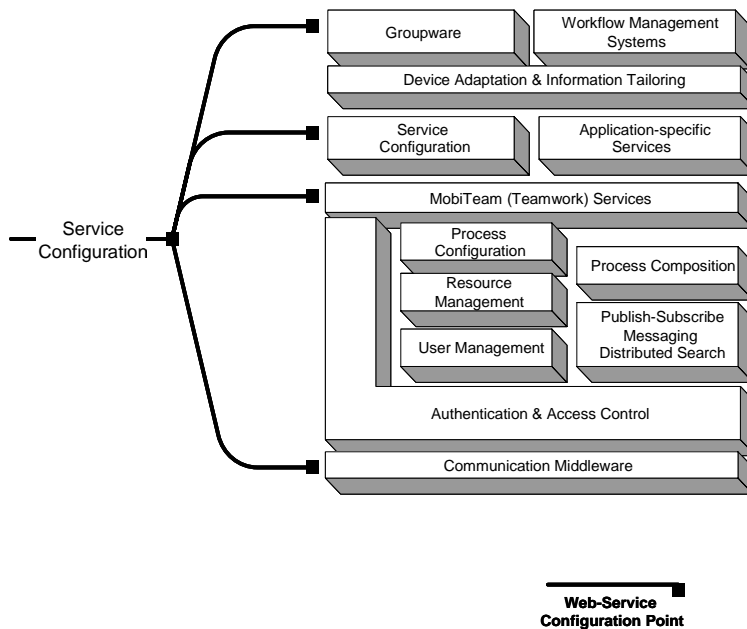


Fig. 8 Web services Management and Configuration Architecture

## 4 Conclusion

This paper presents an analysis of the evolutionary path of mobile services, from early voice communication services to prospects of future service possibilities. As a result of this analysis, some important concepts of mobile services are identified and further discussed. Activities related to mobile services, and their relative *openness* with regards to actors is discussed. It is argued that increasing this openness can help excel the future of mobile services. General mobility requirements are discussed and put in context of openness, as well as the distribution of the service logic and content. Further elaborating these issues, the various hosting domains for services are discussed. Personal networks at home and PANs are identified as two important domains for containing service logic and content for future mobile services. Employing Personal Area Networks as a mobile service domain is briefly considered in particular, since this is the most novel of the two recently identified service domains. It is recognized that the future will need generic redirection mechanisms (of input/output) between service components to enhance availability of services and exploit resources in various devices to the fullest.

Personalisation is presented as a more complex construction than hitherto acknowledged, as it initially can be divided into at least three separate levels. Each of these levels can be further elaborated and sub-divided, but only an initial attempt at this is done in this paper; the rest is left for future work. Last, the evolution of mobile services and technologies supporting them are put into the context of collaborative services.

Each of the concepts discussed around mobile services in this paper are on their own advanced and extensive fields of research and they must be further elaborated in separate studies. Thus, the discussions in this paper are preliminary and do address only the basic structures and further works will be carried out.

## 5 References

1. Gunnar Heine, *GSM Networks: Protocols, Terminology and Implementation*, ISBN 0-8900-6471-7, January 1999
2. ETSI, *Digital cellular telecommunications system (Phase 2+); Specification of the SIM Application Toolkit (SAT) for the Subscriber Identity Module - Mobile Equipment (SIM-ME) interface (3GPP TS 11.14 version 8.15.0 Release 1999)*, 2003
3. Nokia, *CIMD Interface Specification*, 2002
4. Aldiscon Ltd., *Short Message Peer to Peer (SMPP) Interface Specification*, 1996
5. Open Mobile Alliance, *WAP 2.0 Technical White Paper*, January 2002, [http://www.wapforum.org/what/WAPWhite\\_Paper1.pdf](http://www.wapforum.org/what/WAPWhite_Paper1.pdf)
6. Vineet Rajosi Sharma, Indian Institute of Technology, *Study Of Wireless Application Protocol*, 1999, <http://www.iitd.ac.in/~rajosi/mini/>
7. W3C, *XHTML Basic*, W3C Recommendation, December 2000, <http://www.w3.org/TR/xhtml-basic/xhtml-basic.pdf>
8. Internet Assigned Numbers Authority, *MIME Media Types*, <http://www.iana.org/assignments/media-types/>
9. W3C, *Semantic Web*, <http://www.w3.org/2001/sw/>

10. Java Community Process, *JSR 37: Mobile Information Device Profile 1.0 Final Release*, 2000
11. Java Community Process, *JSR 120: Wireless Messaging API Maintenance Draft Review*, January 2003
12. Jan Blom, *Why do we personalise?*, Department of Psychology, University of York,
13. Do, van Thanh, Jønvik Tore, Vanem Erik, Tran, Dao van & Audestad, J.A.: *The Device Management Service*, Proceedings of The IEEE Intelligent Network Workshop 2001 (IN2001), Boston, USA, ISBN 0-7803-7047-3, May 6-9, 2001.
14. Dustdar, S. Gall, H., Schmidt, R. (2004). *Web Services For Groupware in Distributed and Mobile Collaboration*. 12<sup>th</sup> IEEE Euromicro Conference on Parallel, Distributed and Network based Processing (PDP 2004), A Coruña - Spain, February, 11-13, 2004, IEEE Computer Society Press.