VTT SYMPOSIUM 183

**Keywords:** mobile computing, mobile multimedia, data transmission, multimedia applications, software architectures

# Euromicro Summer School on Mobile Computing '98

Oulu, Finland, August 20 - 21, 1998

Edited by

Tapio Seppänen
VTT Electronics

Organised by

Euromicro Office

VTT Electronics

Infotech Oulu

University of Oulu

TECHNICAL RESEARCH CENTRE OF FINLAND
ESPOO 1998

ISBN 951–38–4576–1
ISSN 0357–9387



JULKAISIJA – UTGIVARE – PUBLISHER

Valtion teknillinen tutkimuskeskus (VTT), Vuorimiehentie 5, PL 2000, 02044 VTT
puh. vaihde (09) 4561, faksi 456 4374

Statens tekniska forskningscentral (VTT), Bergsmansvägen 5, PB 2000, 02044 VTT
tel. växel (09) 4561, fax 456 4374

Technical Research Centre of Finland (VTT),
Vuorimiehentie 5, P.O.Box 2000, FIN–02044 VTT, Finland
phone internat. + 358 9 4561, fax + 358 9 456 4374

VTT Elektroniikka, Elektroniikan piirit ja järjestelmät, Kaitoväylä 1, PL 1100, 90571 OULU
puh.vaihde (08) 551 2111, faksi (08) 551 2320

VTT Elektronik, Elektroniska kretsar och system, Kaitoväylä 1, PB 1100, 90571 OULU
tel.växel (08) 551 2111, fax (08) 551 2320

VTT Electronics, Electronic Circuits and Systems,
Kaitoväylä 1, P.O. Box 1100, FIN–90571 OULU, Finland
phone internat. +358 8 551 2111, fax +358 8 551 2320

Technical editing Leena Ukskoski

Oy Edita Ab, ESPOO 1998

# Preface

Mobile computing is considered as one of the key enabling technologies in the development of future services for mobile users. The potential scope of this new technology is remarkable, ranging from new data transmission techniques for fast fading channels, through software architectures, to next generation multimedia applications. Despite the importance of this topic, the number of international conferences focusing on this topic is rather small at the present. Euromicro, therefore, decided to arrange a summer school as forum for bringing together graduate students, researchers and industrial professionals dealing with various technologies, engineering disciplines and applications involved in mobile computing.

The first Euromicro Summer School on Mobile Computing (SSMC'98) is a joint educational, academic and professional forum aimed at covering a wide range of state-of-the-art research and development in mobile computing. The authors of the presentations come from eight European countries, representing a diversity of companies, research institutes and universities.

The event is jointly organised by the Euromicro office and two Finnish research institutes, VTT Electronics and Infotech Oulu of the University of Oulu. The meeting is arranged at the premises of VTT Electronics in the city of Oulu in Finland.

As the Program Chair, I would like to thank all of the dear colleagues who have made their contribution to this joint effort. Thanks are, first of all, due to the authors, for coming to present and discuss their research results. I am also greatly obliged to Professors Nello Scarabottolo, Veikko Seppänen, Matti Pietikäinen and Olli Silven for their invaluable help in the preparation of the event. The researchers at VTT Electronics and Infotech Oulu who have prepared the impressive demonstrations for our participants to enjoy are also acknowledged. The help of Ms. Kiia Kaulanen in the final preparation of the proceedings and other arrangements during the summer school is thankfully acknowledged. Special thanks to Mrs. Katja Salmela for her great work in preparing and managing all the practical arrangements of SSMC'98, as well as

to Mrs. Chiquita Snippe-Marlisa, the Euromicro manager. Finally, I would like to express my gratitude to our sponsors, Euromicro, VTT Electronics and Infotech Oulu for providing support for the summer school.

Tapio Seppänen

Program Chair of SSMC'98

# Contents

# Keynote Session 1:
# "Software-radio technology"

## Speaker

Mika Kasslin

*Finland*

# Software Radio - a Set of Challenges

Mika Kasslin
Nokia Research Center
P.O. Box 45, 00211 Helsinki, FINLAND

## Abstract

Software radio (SWR) is considered a possible solution to the problem of the ever-increasing number of different wireless standards. You might have one single radio for almost all systems, but the SW would be the one totally defining its operational mode. But will this be ever possible? To say the least, there is a great number of challenges to contend with.

## 1. Introduction

The concept of software radio (SWR) is not new, but the definition has varied depending on the application area. From Nokia's point of view SWR is a transceiver with software-definable frequency band and channel bandwidth (figure 1). Additional software-definable features are the channel coding and modulation scheme, mobility and radio resource management. Last, but not least, an essential feature of the SWR is the possibility to change also certain user functions by downloading.

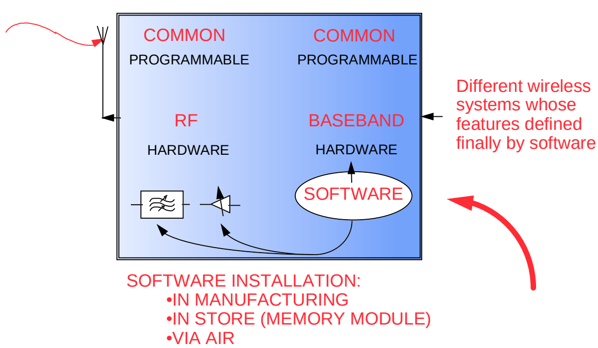



*Figure 1. The software radio concept.*

Additional system flexibility results from the possibility to change the origin of the adaptations. First of all, the parameters or the operations listed may be changed or adapted by the end user. Other possible origins of the change are the network operator and the service provider. In some applications and situations, however, no external initiation is needed and the adaptation or change may take place automatically.

## 2. Possibilities of Software Radio

If SWR is such an old idea, why hasn't it already been used? It actually has, but so far only in military communications. Motorola's SPEAKeasy is an example of a military radio capable of handling several waveforms; basic structure consisting of separate HW units for each of the waveforms. This kind of duplication has been possible since the size and the cost have never been the first design criteria in the military business.

The product cases of base stations and the terminals are much more critical. Not only do size and cost restrict the design, but so does power consumption.

Basic semiconductor technology has now developed to the stage that for the first time allows us to apply the idea of SWR to terminals and base stations. Memory prices have been dropping down while the MIPS count of microprocessors has been increasing extremely fast. New tuneable and configurable building blocks may be developed. The tasks usually done by a fixed HW may be replaced by configurable HW blocks, or they may be done totally by SW. As a result, we will have a more flexible radio offering a wider range of systems for the end user and a more cost efficient solution for the manufacturer.

## 3. Challenges in software radio

The rapid development of semiconductors is only an enabler, not a solution. There are several challenges to face on the road to the ultimate goal of SWR having a common baseband (BB) and RF HW for several different systems. The basic requirements are a higher integration level, parameterized BB modules and configurable RF architecture.

The challenges may be divided into two main categories: the architecture level and the module level. The main challenges are listed on figures 2 and 3.

*Table 1. Architecture challenges.*

| |
|---|
| Partitioning between analog and digital |
| Partitioning between digital processing units (e.g. DSP, RISC, FPGA) |
| Modular SW architecture |
| Dynamic processing capacity allocation |
| Required duplication of frequency-dependent parts |

*Table 2. Module level challenges.*

| **Analog domain** | **Digital domain** |
|---|---|
| Antennas: diversity, different bands etc. | High performance DSP: flexible architecture, configurable digital logic, vector processing etc. |
| RF/IF filtering: different bands and bandwidths | Embedded memory technology |
| PA linearity, power efficiency etc. | Algorithms |
| Wide operation range synthesizers and VCOs | Bus technologies |
| Fast A/D & D/A converters | |

Only a few problems are described here to give an idea of the level of problems, but all of the problems must be solved before SWR is implementable.

## 3.1 A.  Where to sample?

The most obvious question when considering the architecture of a transceiver is where to draw the line between the analog and digital world.

The system specifications set operational limits and requirements like output power and selectivity that the SWR transceiver has to fulfill. It must be decided which tasks will be done in the digital domain and which have to be performed before the A/D-conversion. The digital and SW solutions are more stable and flexible than the analog ones. On the other hand, the high power consumption of the digital solutions limit their use, especially in terminals. Some of the single performance figures and flexibility must be compromised to gain the total requirements.

## 3.2 B.  Modularity requirements

More challenges arise when going further in the architecture selection. In the digital domain you have to find out the most efficient way to perform the required tasks. The partitioning between processors like DSPs and RISCs, HW accelerators or configurable logic is a very challenging problem. In the digital front-end the requirements for the processing speed are hardly met without accelerators or configurable logic. The processor cores, instead, provide both configurability and future proof. It is a challenge to develop such a modular BB unit that provides both flexibility and enough processing power.

The modularity requirement is not restricted only to HW. Also the SW has to be designed and implemented to be modular. First of all, we have to find out commonalties in different systems. Only this way we may build up common re-usable SW modules that can be used in all systems. On the other hand, the SW has to be divided into bigger units performed by the real HW blocks. This is done by mapping the individual modules to the HW architecture. A well designed and realised mapping enables the efficient and dynamic use of HW resources.

## 3.3 C. Fast and configurable samplers

The location of the D/A and A/D-converters influences numerous individual modules in the SWR transceiver, directly reflecting the allocation of complexity between the modules. The biggest challenges in the module level are the A/D conversion itself and the RF/IF filtering. First, you have to note that both of them are tightly coupled with selection of the total transceiver architecture. The higher the frequency at which the sampling is performed, the tighter the requirements set for the A/D converter. Both the sampling frequency and the resolution of the sampler are normally increased when sampling at higher frequencies. As an extreme example, the sampling straight at RF in GSM would require a sampler with 16 or even 17 bits without automatic gain control (AGC) before sampling. Additionally, the jitter of the sampling clock must not exceed 2 ps in that case. It must also be taken into account that the sampling pulse width has to be adjustable to the carrier frequency; optimal when equal to the half of the carrier pulse.

## 3.4 D. Configurable analog filters

Just like with sampling, there is a straight connection between the total architecture selection and the number of filtering stages. The lower the frequency, the easier the filter design and the cheaper the solutions. On the other hand, it must be decided whether to do all the channel filtering before sampling or to leave some of the channel filtering to be done digitally. In conclusion, the problems of filtering and architecture can not be solved independently. At least you have to find the upper performance limits of both the analog and digital filtering before you can draw a line between the analog and digital domains.

Another challenge in channel filtering is minimising the number of physical filters. Both the bands and bandwidths vary from system to another. To provide such configurable filters, new filter structures have to be studied and designed. It is not enough to simply duplicate all the filters. Especially for the terminals, this kind of solution is too expensive and voluminous.

## 4. Multimode transceiver by SWR?

Solutions to the previous few problems are not enough; we must also tackle many other similar problems. These include antennas, PAs and oscillators for different bands as well as new processor techniques, just to mention a few.

Summarising, it is hardly ever possible to implement a true SWR transceiver that can be reconfigured to any existing and forthcoming system. However, facing the challenges and solving them for a limited set of systems results in an SWR concept that may help us to build multimode transceivers efficiently.

# Session 1:

# "Terminals"

# $3^{rd}$ Generation Mobile Terminals

Tom Leskinen
Nokia Mobile Phones
P.O. Box 68, 33721Tampere, Finland

## Abstract

This paper presents a possible view of future mobile terminals. The focus is on the terminal requirements for $3^{rd}$ generation future mobile communications systems, such as Universal Mobile Telecommunications System (UMTS) and International Mobile Telecommunications 2000 (IMT2000). Brief introduction to system technology development is given as well as to trends affecting mobile terminals in the future. The goals and objectives of $3^{rd}$ generation mobile systems from the terminal perspective is briefly presented.

## 1. Introduction

During recent years mobile terminals have become important items for business people as well as for other users. Technology development enabled the integration of the mobile terminal into a small and convenient package. At the same time deregulation of the telecommunications market fueled the price and service competition to enable better quality services with affordable prices. These drivers enabled the huge growth seen during recent years. Wide variety of new and different type of services will also require support from the mobile terminals, which will increase terminal functionality remarkably.

Mobile communications started from the speech service on the analog systems. Digital 2nd generation systems such as GSM (Global System for Mobile Communications) offered in addition to analog systems large variety of supplementary services and good quality data service. Future systems enhance the capabilities of these existing services and add possibilities for new services, such as multimedia service delivery (figure 1) [1].

Wireless multimedia services are the most challenging future addition to the mobile service range [2]. 3rd generation mobile systems as well as enhancements to 2nd generation will bring mobile multimedia service capabilities available for the end users. Introduction of these services will be gradual. Starting from the images, then video clips and finally moving to more complex multimedia streams. Both system capabilities and implementation technologies will require gradual stepwise introduction of these services. For example new Nokia 9110 Communicator is able to read images from digital cameras and further transmit them over GSM 14.4kbit/s wireless data channel.

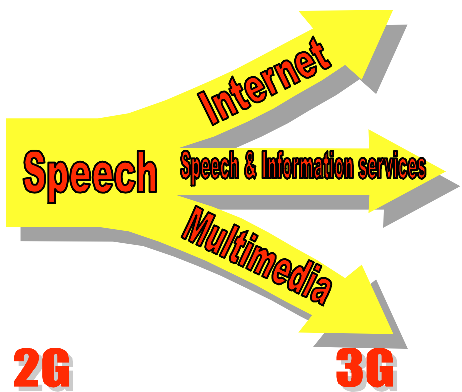



*Figure 1. Usage trends of mobiles*

## 2. Mobile System Development

The recent development of the digital mobile communication systems has been extremely rapid. After the introduction of GSM there have been several activities to develop these systems to cope with the requirements seen in the near future. This development has concentrated on adding new features, enhancing the data transmission capabilities and interworking towards other networks, among others. Major development items in GSM have been GPRS (General Packet Radio Service), HSCSD (High Speed Circuit Switched Data) and EDGE (Enhanced Data rates for GSM Evolution) [3].

| | GPRS | HSCSD | EDGE |
|---|---|---|---|
| Service Type | Packet data | Circuit data | Circuit + packet data |
| Applications | WWW, Internet, Information delivery, etc. | WWW, file transfer, real-time video, etc. | WWW, file transfer, Internet, video, etc. |
| Max bit rate | 171.2 kbit/s | 64 kbit/s | Appr. 384 kbit/s |
| Estimated market entry | 1999/2000 | 1999 | 2000/2001 |

UMTS development and standardization is still ongoing. Main focus is to optimize UMTS for realistic wide area high bit rates and to support data and multimedia services (figure 2). UMTS is targeted to offer 2Mbit/s data rates in indoor as well as in short range outdoor environments. In longer range outdoor and vehicle environments realistic data rates are between 128-384 kbit/s. An important fact for the UMTS development is that the actual multimedia services to be offered are still unknown and therefore won’t be a part of the UMTS standard. By having flexible and dynamic way of bringing new services for the end users, UMTS will be future proof and able to answer rapidly to the changing market needs. Therefore UMTS will be developed to be flexible in service introduction, service provision as well as negotiation of different underlying functionality.

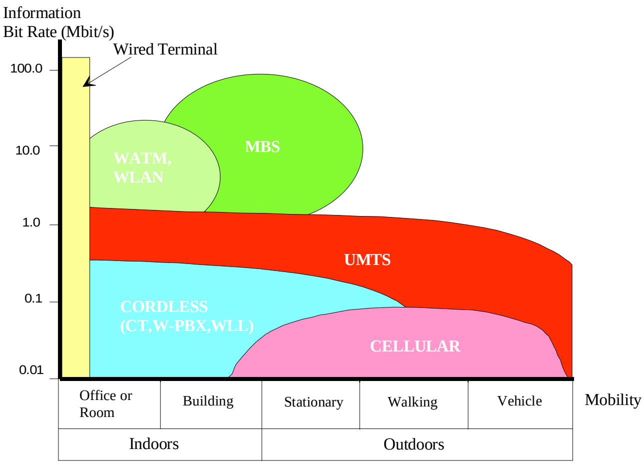



*Figure 2. UMTS support for bit rates*

Wireless Application Protocol (WAP) is an initiative to produce simple service framework and protocol stack for offering Internet-type of services for the users of mobile terminals ranging from basic data terminals to PDA type of terminals. WAP enables provision of information services with relatively small amounts of information. Internet on the other hand is being developed to support guaranteed delivery times as well as guaranteed data rates being able to support new QoS requirements for emerging video, audio and multimedia applications. These network protocols will form the basis of 3G mobile terminal connectivity for information services.

# 3. Terminal development trends

Currently there exists already many different type of terminals. Speech only terminals, data terminals (e.g. PCMCIA radio cards), terminals with speech and data capabilities, and communicator type of terminals with added functionality like Internet access, calendar, note editor, calculator, etc. However, in future it is seen that the range of terminals will even grow to include different types of multimedia terminals and Information access terminals.

Terminal development trends faced in today's terminals are mainly towards higher integration levels resulting into smaller size. The goal of four 100's has been a "rule of thumb" target for handsets, i.e., 100 hour standby, 100 cc size, 100 gram weight and also 100 MIPS performance. Size targets have been already achieved and the need for smaller terminals is questionable. The achieved size target can also be seen as an enabling factor for integration of extra functions and features into these small devices. The other target parameters have no minimum limitations. On the other hand, we can see the following further trends for near future terminals:

- Application specific terminals (smart traffic, vending machine radio, etc.).
- Increased number of value adding features (graphics, messaging, PC connectivity, PC compatibility,).
- Support for higher number of source codecs (several speech codecs).
- Multiband terminals (e.g., GSM and DCS1800).
- Multimode terminals (e.g., GSM/UMTS dualmode).
- Dynamic SW configurability.

These trends are more than likely to continue in the future. Multiband and multimode terminals with high integration levels would be preferred by some users. Technological development of these terminals rely on new packaging and interconnection technologies, as well as technological steps like SW-radio [4]. The concept trends of mobile handhelds can be considered to divert from simple speech terminals towards a variety of different types, e.g., communicators, wearable phones, data terminals, etc. The dominant role of speech terminals will be challenged in the future by these new data- and multimedia-oriented terminals.

Terminal implementation technologies, such as digitalisation providing programmability and terminal configurability, VLSI, and display technologies, have developed a lot recently and are a subject of further development in the future. Processing power, implementation architectures, IC and passive integration, and memory technologies are developing rapidly enabling increase of functionality. This development enables higher integration of terminals, as well as integration of more functionality into smaller terminals.

In UMTS standards, terminals are not restricted in any ways. Therefore any kinds of terminals could exist. The most critical issues for 3rd generation mobile terminals will be their capability to adapt to new applications and services. Moreover, many of these new services will be based on utilizing multimedia and therefore the requirements are even tougher to meet. To fulfill these requirements many activities exist to specify the needed functionality and API's (Application Programming Interfaces). Possible Java based API's and Mobile Application Execution Environments (MexE) will be specified for UMTS as well as functionality to support software downloading to terminals [5].


*Figure 3. Example of future multimedia terminal.*

## 4. Conclusions

In UMTS time frame the system development will increase the possibilities for introduction of wireless multimedia services. The actual services are provided and innovated by the Service Providers, increasing the available range of different multimedia services. Terminal implementations will become highly complex because of the unawareness of the provided services, as well as the processing requirements set by the complex multimedia algorithms involved. Terminal types are not restricted in any ways, and therefore a wide variety of terminals, ranging from simple speech only terminals to complex multimedia terminals, is expected to evolve.

## References

1. UMTS Forum, Draft report v.1.0.0, A Regulatory Framework for UMTS, 1997.

2. T. Leskinen, Multimedia Service Aspects of Future Wireless Systems, Thesis for the degree of Licentiate in Technology, Tampere University of Technology, June 1997.

3. J. Hämäläinen, Design of GSM High Speed Data Services, Doctoral Thesis, 104 p., Tampere University of Technology, October 1996.

4. J. Mitola, The Software Radio Architecture, IEEE Communications Magazine, May 1995.

5. UMTS ETS 22.01, Universal Mobile Communications System (UMTS), Service Aspects; Service Principles, v. 3.0.1, February 1997.

# The Pocket Companion's Architecture

Paul J. M. Havinga, Gerard J. M. Smit
University of Twente,
P.O. Box 217, 500 AE Enschede,
The Netherlands

## Abstract

The Pocket Companion is a small personal portable computer with wireless communication facilities. The typical use of the Pocket Companion induces a number of requirements concerning security, performance, energy consumption, communication and size. The energy consumption due to the increasing demand for performance and functionality will be the limiting factor for its capabilities. Therefore reducing energy consumption plays a crucial role in the architecture. Communication, and particularly wireless communication, is essential for the system to support electronic transactions. Such a system requires a good security infrastructure not only for safeguarding personal data, but also to allow safe transactions.

## 1. Introduction

The Moby Dick project (Esprit Long Term Research 20422) [8] develops and defines the architecture of a new generation of hand-held computers, the so-called Pocket Companion. It is a small personal portable computer and wireless communications device that can replace cash, cheque book, passport, keys, diary, phone, pager, maps and possibly briefcases as well. Typical applications of a Pocket Companion are diary, e-mail, note-taking and electronic payments.

The Pocket Companion is resource-poor, i.e. small amount of memory, limited battery life, low processing power, and connected with the environment via a (wireless) network with variable connectivity. The Pocket Companion is more than just a small machine to be used by one person at a time like the traditional organizers and desktop assistants. The Pocket Companion extends the no tion of a ‘desktop companion’. The Pocket Companion will run applications typically found in desktop companions, but it will run other applications using external public services as well.

### A. Problem statement

The typical use of the Pocket Companion impels a number of requirements on the system architecture of the Pocket Companion concerning security, performance, energy consumption and communication. In current architectures these items have been dealt often as separate or add-on items. Although these issues could be treated as separate problems, we believe that they are interrelated and can only be solved optimal when they are integrated in one single architecture.

- Performance and energy consumption

To support multimedia functionality for the intended applications of the Pocket Companion the system needs to have real-time properties. The increasing levels of performance and integration that is required will be accompanied by increasing levels of energy consumption. Without significant energy reduction techniques and energy saving architectures, battery life constraints will limit the capabilities of a Pocket Companion. More extensive and continuous use of network services will only aggravate this problem since communication consumes relatively much energy [7,10].

- Communication

Communication, and particularly wireless communication, is essential for the system to allow electronic transactions. Wireless links differ from wired networks in many aspects that have impact on the system design level. These include high and variable latency, low bandwidth, incomplete spatial coverage, energy consumption and communication cost.

- Security

---

University of Twente, dept. Computer Science, P.O. Box 217, 7500 AE Enschede, the Netherlands; {havinga, smit }@cs.utwente.nl

A Pocket Companion that interacts with foreign services - under full user control and according to the owner's notion of trust and security - requires a good security infrastructure. The privacy of the owner's personal sensitive data has to be guaranteed and the integrity and the authenticity of transactions with the environment has to be ensured. In the Moby Dick architecture security is an integrated part of the system architecture.
An important requirement for a Pocket Companion is flexibility and programmability. In our vision, Pocket Companions will need to provide the flexibility to handle a variety of (multimedia) services and standards (like different video decompression schemes and security mechanisms) and the adaptability to accommodate to its current environment for the changing conditions in communication connectivity, required level of security, and available resources.

## 2. The Pocket Companion's system architecture

Over the past two decades the semiconductor technology has been continuously improved and has lead to ever smaller dimensions of transistors, higher packaging density, faster circuits, and lower power dissipation. Such advances provide an effective area increase of about an order of magnitude that can be used for several goals, e.g. to increase performance, to add functionality, but also to reduce energy consumption.

The goal of the Pocket Companion's architecture is to optimize the overall energy-performance of the system, and not performance alone. The technology is used to decrease energy consumption and to increase functionality to provide services such as compression and decompression, network access, multimedia, and security functions.

### *A. Architecture*

The difficulty in achieving all requirements into one architecture stems from the inherent trade-offs between flexibility and energy consumption, and also between performance and energy consumption. Flexibility requires generalized computation and communication structures, that can be used to implement different kinds of algorithms. While conventional architectures (like used in current laptops) can be programmed to perform virtually any computational task, they achieve this at the cost of high energy consumption. The Pocket Companion has a rather unconventional architecture that saves energy by using system decomposition at different levels of the architecture and exploiting locality of reference with dedicated, optimized modules. Security incurs another trade-off: in current systems a high security level can only be reached at high costs and require much performance (or patience).

Our approach is based on dedicated functionality and the extensive use of power reduction techniques at all levels of system design [3]. The system has a number of premises:

- An architecture with a general purpose processor surrounded by a set of heterogeneous programmable modules, each providing an energy efficient implementation of dedicated tasks [1].
- A reconfigurable internal communication network that exploits locality of reference and eliminates wasteful data copies.
- A system design that avoids wasteful activity: e.g. by the use of autonomous modules that can be powered down individually and are data driven.
- A wireless communication system designed for low energy consumption by the use of intelligent network interfaces that deal efficiently with a mobile environment, by using a power aware network protocol, and by using a energy aware MAC protocol.
- An architecture for signing arbitrary contracts. The architecture can divide the system in a general computing processor and a trusted computing base (TCB) that is not user-programmable.

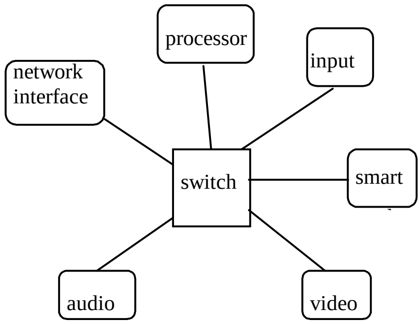



*Fig. 1: Pocket Companion system architecture*

Figure 1 gives a schematic overview of the Pocket Companion architecture. In it, we distinguish a switch with a security module surrounded by several modules. The switch interconnects the modules and provides a reliable path for communication between modules. As in switching networks, the use of a multi-path topology will enable parallel data flows between different pairs of modules and thus will increase the performance. Modules are autonomous and can communicate without involvement of the main processor. Each module has its own functionality and responsibility. The modules are optimized for their function in terms of performance and energy consumption and have their own local power management.

## A. Energy consumption and performance

The design flow of a system constitutes of various levels of abstraction. When a system is designed with the emphasis on power optimization as a performance goal, then the design must embody optimization at all levels of the design flow. In general there are three levels on which energy reduction can be incorporated. The system level, the architecture level, and the technological level. As the most effective design decisions stemm from the architectural and system level, a careful design at these levels can reduce the power consumption considerable. For multimedia application in particular there is a substantial reduction in energy consumption possible as the computational complexity is high, they have a regular and spatially local computation, and the communication between modules is significant. Improving the energy efficiency by exploiting locality of reference and using efficient application-specific modules therefore has a substantial impact on a system like the Pocket Companion.

## *B. Networking*

The wireless network interface consumes a significant fraction of the total power [11]. Therefore we put considerable effort in reducing the energy consumption of the network interface.

- MAC protocol

We use a TDMA protocol to coordinate delivery of data to receivers [6]. The basic objective is that the protocol tries to minimize all actions of the network interface, i.e. minimize 'on-time' of the transmitter as well as the receiver. A base station is responsible for traffic scheduling. Mobiles with scheduled traffic are indicated in a list, which allows mobiles without traffic to reduce power rapidly. As switching between states (i.e. off, idle, receiving, transmitting) consumes time and energy, the number of state-transitions is minimized. By scheduling bulk data transfers, an inactive terminal is allowed to doze and power off the receiver.

Measurements have shown [11] that on typical applications like a web-browser or e-mail, the energy consumed when the interface is on and idle is more than the cost of transceiving packets. This is because the interface is generally longer idle that actually communicating. Therefore we also consider to use a very low power transceiver for the signalling only. This transceiver will be used for the MAC protocol and operates in parallel with the actual data-stream with another tranceiver on the same interface. This data-stream transceiver has more bandwidth and consumes more energy, but will be turned on only when there is actually data to be transmitted, and is not used for ‘useless’ signalling.

- Error control

Wireless networks have a much higher error rate than the normal wired networks. In the presence of a high packet error rate and periods of intermittent connectivity characteristics of wireless links, some network protocols (such as TCP) may overreact to packet losses, mistaking them for congestion. The limitations of TCP can be overcome by a more adequate congestion control during packet errors [2]. In Moby Dick several methods will be examined and compared. Forward Error Correction (FEC) will be used to improve the performance and to reduce energy consumption, not only at the data link level, but also in higher levels of the protocol stack [9].

- Decomposition

In normal systems much of the LAN protocol stack is implemented on the main processor. Thus, the network interface and the main processor must always be 'on' for the LAN to be active. Decomposition of the network protocol stack and a careful analysis the data flow in the system can reduce the energy consumption. For example, the network module can handle most of the lower levels of the protocol stack and can move the data directly to its destination, thereby allowing the main processor to sleep for extended periods of time without affecting system performance or functionality. Furthermore, part of the network protocol can be handled on another machine, e.g. the base station. For example, the mobile units can use a private lightweight protocol rather than a protocol like TCP/IP resulting in a net savings of energy.

## *A. Security*

The heart of the security system is the Trusted Computing Base (TCB). This is the part of the system the owner of a Pocket Companion has to trust. The TCB consists of an input device (e.g. keyboard), the display, a stable storage to log transactions, a smart card, and the switch that connects these modules. A crucial property of this architecture is that the TCB is not user-programmable and built by a reliable manufacturer. If it does not have this property, then if the application on the main processor is corrupted, it can for example falsify data on the display, it can squirrel away the password or PIN code, and can sign other unauthorized messages as well.

A typical example of using the security function is the signing of a contract during some transaction [4]. Normally - that is when no contract signing is going on -, display and input device are available to the local processor as its input/output device. But, when a contract has to be signed, the security module and switch disconnects display and keyboard from the main processor and takes them over for the secure signing purpose. The main processor can no longer modify the display or read the keyboard. A smart card contains, protects and applies a signature key. The card also stores the certificates that authenticates the key. When the contract is signed the main processor regains the ownership of the keyboard and display.

## 3. Current status

Currently we are designing and building a testbed for the Pocket Companion using existing hardware components and subsystems. The hardware basis of the Pocket Companion is an interconnection switch. All data in the system is based on the size of an ATM cell (48 bytes data). This not only allows us to easily interconnect with an ATM environment, but the size is also adequate: it is small enough to buffer several cells in the switch and have small latency, and large enough to keep the overhead small.

The prototype of the switch is built using a Xilinx FPGA and six small microcontrollers. The switch is capable to connect six modules. It is a tiny ATM switch as it is able to transfer ATM cells to a destination according to its VCI. The basic functions of the microcontrollers is to perform routing, to establish a connection and to interface with the connected modules. The design methodology used is data-driven and based on asynchronous methods, combined with synchronous parts. Each part of the switch that is not used at some time does not receive any data and clock changes, thus minimizing energy consumption.

The attached modules can be similar to today's PDAs, notebook computers or `handheld PCs' augmented with a security module and one or several wireless network devices. Currently we are using the WaveLAN modem as network interface to exeperiment with MAC protocols and error correction. With this testbed we can easily evaluate (parts of) the architecture concerning energy consumption, security, Quality of Service and communication. We gradually improve the architecture to the final architecture.

We also experiment with system software, as hardware architecture and system software are related. The operating system being used is Inferno from Lucent Technologies [5]. This system is quite well suited for experimenting with systems like the Pocket Companion.

## References


1. Abnous A, Rabaey J.: "Ultra-Low-Power Domain-Specific Multimedia Processors," Proceedings of the IEEE VLSI Signal Processing Workshop, San Francisco, October 1996.

2. Balakrishnan H., et al.: "A comparison of mechanisms for improving TCP performance over wireless links", Proc. ACM SIGCOMM'96, Stanford, CA, USA, August 1996.

3. Havinga P.J.M., Smit G.J.M.: "Low power system design techniques for mobile computers", CTIT technical report 97-32, 1997

4. Helme A., Mullender S.J.: "An architecture you can trust", internal report University of Twente, March 1997.

5. Information on Inferno is available on http://cruel.com/vanni/

6. Linnenbank, G.R.J. et al.: "A request-TDMA multiple-access scheme for wireless multimedia networks", Proceedings Third Workshop on Mobile Multimedia Communications (MoMuC-3), 1996.

7. W. Mangione-Smith, et al.: "A low power architecture for wireless multimedia systems: lessons learned from building a power hog", Proceedings of the international symposium on low power electronics and design, August 1996

8. Mullender S.J., Corsini P., Hartvigsen G. "Moby Dick - The Mobile Digital Companion", LTR 20422, Annex I - Project Programme, December 1995 (see also http://www.cs.utwente.nl/~havinga/pp.html).

9. Rizzo L.: "On the feasibility of software FEC", internal report University of Pisa, 1997, available at: http://www.iet.unipi.it/~luigi/fec.html.

10. Sheng S., Chandrakasan A., Brodersen R.W.: "A Portable Multimedia Terminal", IEEE Communications Magazine, pp. 64-75, vol. 30, no. 12, Dec., 1992.

11. M. Stemm, et al. "Reducing power consumption of network interfaces in hand-held devices", Proceedings MoMuC-3, 1996.

# CyPhone - Mobile Telepresence and Augmented Reality for 3rd Generation Cellular Phone

Petri Pulli, Tino Pyssysalo
VTT Electronics and University of Oulu Infotech Research Center
P.O. Box 1100, 90571 Oulu, Finland

Jukka-Pekka Metsävainio, Olli Komulainen
J-P Metsävainio Design Oy
Kirkkokatu 8A7, 90100 Oulu, Finland

## Abstract

Advances in multimedia, virtual reality, and immersive environments have expanded human-computer interaction beyond text and vision to include touch, gestures, voice and 3D sound. Although there exist well-developed single modalities for communication, we do not really understand the general problem of designing integrated multimodal systems. Recent advances in mobile communication based on picocellular technologies allow the transmission of high bandwidth of data over personal surrounding networks. In this paper we analyse the sources of real-time constraints in telepresence and augmented reality applications. We offer an approach to adding aspects of mobility and augmented reality to real-time mobile telepresence, discuss the technology and potential future product concept vision, the CyPhone, and depict the general architecture and integration framework briefly. Finally, a survey of relevant telecooperation services are introduced.

# 1. Introduction

Recent visionary research [1] strongly suggests future information society going towards virtualisation. Examples of proposed virtual telepresence based services are: virtual meetings, electronic shopping, games and entertainment, guiding and tourist services, virtual village, virtual family, personal memory support systems.

Our work is based on the emerging understanding that there is a major trend towards personal advanced telecommunication services. “Personal” means that these services are mobile and conveniently available whenever and wherever we want/need to participate in communication activities. We are carrying out a long-term scientific study for how are we going to bring future increasing communication bandwidth and computing performance to the personal vicinity and personal use of individual human beings, meaning everyone of us. We focus on scientific study of user-interaction layer of the future broadband personal telecommunication products and of services. Through better understanding of user interaction issues of future telecommunication services we believe that, besides scientific contributions, we increase the potential for the telecommunication product and service industry to grow in the future.

Scientifically we are utilising two major approaches to extend the use of computing and communication resources: "ubiquitous computing" and "augmented reality". "Ubiquitous computing" is a term coined by Weiser [2] to mean a situation, where small computational devices are embedded into our everyday environment in a way that allows them to be operated seamlessly and transparently. These devices are suggested to be active and aware of their surroundings so that they can react and emit information when needed. One implementation of ubiquitous computing are active badges, that can trigger automatic doors and give information about the location of a person. Weiser's team and others at Xerox have experimented the idea by using several types of devices, like small pager-sized "Tabs", notebook-sized "Pads" and whiteboard-sized "Boards" [3].

"Augmented reality" [4, 5, 6, 7] is a research approach that attempts to integrate some form of computer media with the real world. When in ubiquitous computing there are many different active devices, in many cases each of them having their own display and interaction devices, the augmented reality approach usually uses much fever devices and aims at a seamless integration between real and digital. The integration may be between paper and electronic documents, like in DigiDesk [4], or even more commonly overlaying digital information (as a non-immersive virtual reality) on real world images [5]. The overlaying of images may take place in several ways, like by using video projection [4, 8], by the means of small, hand-held video screens or palmtop computers [6], or by mixing surrounding reality with non-immersive VR by using head-mounted see-through displays [4].

The core idea of our approach is that by using a very short distance radio communication network it is possible to mix these two approaches. Thus we suggest "ubiquitous computing" in the form where different devices in spaces and places we move around are computationally active and can recognize our presence and identity. But instead of a multitude of different displays and interaction devices we suggest that the interaction with all devices would take place in an "augmented reality", for example by using a head-mounted see-through display and a mobile phone/remote controller.

## 2. Mobile Virtual Reality Architecture

Several networked virtual reality environments [11, 12, 13, 14] exist today, but none of them supports mobility of users. Most of the networked environments are based on Internet, which could be easily replaced by a mobile multimedia wireless network like SWAN [15] providing mobile connections. However, typical currently available applications and their interfaces, based on immersive virtual reality and heavy desk-top computers, would still restrict the user's ability to move and access services in a natural and convenient way. In the Nara Institute of Science and Technology in Japan [16] an experimental mobile virtual reality system is being developed. This system like ours [17] is based on augmented reality merging both real and virtual environments to provide totally new services and interfaces to mobile observers.

The backbone of the mobile virtual reality is a wireless picocellular network, such as Bluetooth [18]. The benefits of using very small cells in mobile virtual reality are obvious. The smaller the cell size, the higher the throughput, because there are fewer users in each cell and higher transmission frequencies can be used. Usually very high frequencies are not used in mobile networks, because of a quick signal attenuation, but if the transmission range is just a few meters, the effect of attenuation is almost negligible. In addition, smaller cell size enables greater frequency reuse. The diameter of a PSN cell in our system is going to be some three meters, which enables the construction of small very low-powered hand held terminals still capable of transmitting high bandwidth multimedia data required by virtual reality applications.

Conventional mobile networks consisting of very small cells have two serious drawbacks: The number of base stations and handovers will be enormous. In our system adjacent PSNs can change information directly without using a fixed base station. This does not only enable wireless communication between user's personal terminals but makes it possible for two users to transmit data to each other directly too. In fact each user's personal surrounding network constitutes a mobile base station, which can forward traffic packets between a fixed base station and some user outside the cell around the base station. In this way the number of expensive base stations needed can be greatly reduced.

## 3. Real-Time Aspects of Telepresence

We distinct different sources of real-time constraints in teleprense and augmented reality applications

- Peer-to-peer object detection time
- Public network access and transmission delay
- Annotation fetch cycle
- Annotation syncronization cycle
- Local tracking.

Peer-to-peer object detection time determines how quickly the receiver can gain telepresence in the view or place desired. This is ultimate Quality of Service attribute, bacause if the telepresence is too slow to direct what the receiver is interested to experience, the user satisfaction will become low. A recent study proposes use of real-time view-dependent image generation from omnidirectional video streams [19].

Public network access and transmission delay determines the setup and reset delays  for telepresence connections. They also determine the jitter quality of the video + audio stream passed through the network. It would be best to have constant delay for the stream of packets. A recent study proposes rate-based transport protocols [20]. Otherwise the receiver has to maintain a long packet reorganisation queue in order to provide a fixed sample rate characteristics.

Annotation fetch cycle determines how quicly the terminal can access virtual world-model server containg the intelligence for generating the annotations. Annotations are generated based on information requested, intelligent objects referenced, place of observer, time, and what class of service the customer has subscribed.

Annotation syncronization cycle determines how quickly the augmented reality annotations can follow the receiver viewing direction movements, i-e. how well they keep syncronized with the real-world.

Local tracking detemines how quicly and accurately user movements and movements of intelligent products are recognized by the tracking system. It is a function of tracking accuracy, sampling rate and filtering, as well as the distance how far the tracking reaches and relative speeds of user and objects.

## 4. A Future Product Concept Vision - CyPhone

Our current workis demonstrated by imaginary product concept "CyPhone" in Figures 1 - 6. Please note that CyberPhone does not exist yet, it is just a vision of the future, and we expect that it  can be real around 2003 - 2008.

CyPhone can serve a s a product platform for many potential value-added services. We have considered services falling in to following categories:

- Telepresence services (tourism, teletaching, nursery)
- Annotation services (guidance, electronic commerce)
- Monitoring & maintenance services (real-estate and property maintenance and alarm systems)
- Home services (child and seniour citized day-care)

- Entertainment services (group games, athletics, training)
- Personal services (pets, tamagotchis, virtual family, virtual friends)

*Figure 1. CyPhone has a 1:1 user interface because a touch-pad and a display screen have same size. The shape of left side is designed for bringing the mini displays near eyes.*

*Figure 2. A person is transmitting with CyPhone as watching through a binocular. The receiver can be anywhere.*

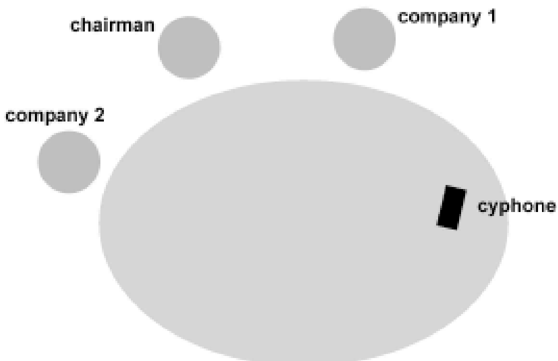



*Figure 3. A logical view of a telepresence meeting with a chairman and two company representatives physically present.*

*Figure 4. Physical representatives of the telepresence meeting.*

*Figure 5. Chairman's view of the telepresence meeting through augmented reality eyeglasses.*

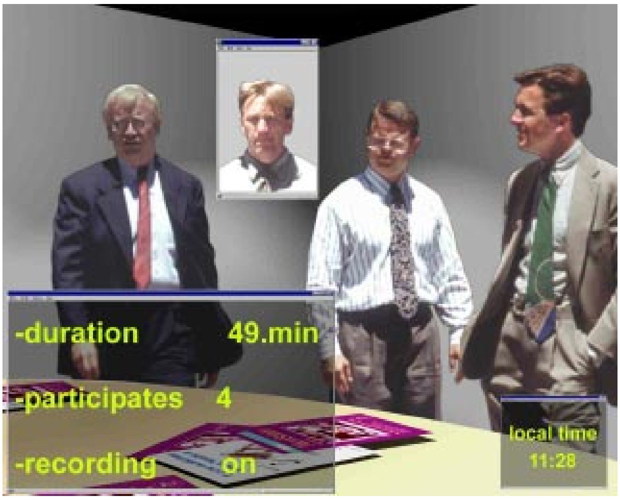



*Figure 6. Remote participant's view of the telepresence meeting in a desktop environment.*

## 5. Discussion and Conclusions

Currently we are building research environment and experimenting with a multimodal personal augmented reality user interface for CyPhone product concept, which is a small-sized combination of a stereo digital camera, notepad computer and a cellular phone. Beside the current target we believe that our approach has a much wider utilisation reserve.

Understanding the real-time requirements is very important in our research. We need better understanding in order to specify the requirements for the underlying communication network, where to locate the intelligence in the network (data and computing), and where to do the computing most efficiently in terms of real-time response.

Another important area of work is telepresence and augmented reality based telecommunication value-added service research and development, experimentation and field trials. In this area we are currently actively pursuing a regional inititiative Mobile City Oulu especially for advanced mobile services

We are open for cooperation in the areas of research indicated in this paper and look possibilities for cooperation in the EU 5th Framework Program.

## 6. Acknowledgements

This research has been carried out in PIHVI project (Picocellular based personal virtual products and services) co-funded by Technology Development Centre of Finland (TEKES), CCC Software Professionals, Elektrobit, Nokia Mobile Phones, Polar Electro, and Telecom Finland.

## References

[1] Thalmann D. (rapporteur) From Multimedia to Telepresence. In: Konidaris, S. (coordinator): Communications for Society, Visionary Research. European Commision DG XIII/B. Brussels. February 1997. Pp. 173 - 195.

[2] Weiser, M. (1991) The Computer for the 21st Century. Scientific American, 265 (3), pp. 94-104.

[3] Buxton, W. (in press) Living in Augmented Reality. Ubiquitous Media and Reactive Environments. To appear in Finn, Sellen & Wilber (Eds.) Video Mediated Communication. Hillsdale, N.J.: Erlbaum.

[4] Wellner, P., Mackay, W., and Gould, R. (1993) Computer Augmented Environments: Back to the Real World. Introduction to the special issue. Commun. ACM, Vol 36 No. 7 (July 1993), pp. 24-26.

[5] Feiner, S., Macintyre, B., and Seligmann, D. (1993) Knowledge-based augmented reality. Commun. ACM, Vol 36 No. 7 (July 1993), pp. 52-62.

[6] Fitzmaurice, G.W. (1993) Situated Information Spaces and Spatially Aware Palmtop Computers. Commun. ACM, Vol 36 No. 7 (July 1993), pp. 38-49.

[7] Cooperstock, J. R., Tanikoshi, K., Beirne, G., Narine, T., and Buxton, W. (1995) Evolution of a reactive Environment. Proceedings of CHI'95, New York: ACM Press, pp. 170-177.

[8] Stafford-Fraser, Q. (1996) BrightBoard: A Video-Augmented Environment. Proceedings of CHI'96, New York:ACM Press, pp. 134-141.

[9] MacIntyre B., Feiner S. (1996) Future multimedia user interfaces. Multimedia Systems (1996) 4: 250-268.

[10] Burdea, G.C. (1996) Force and Touch Feedback for Virtual Reality. Wiley, New York. 339p.

[11] Carlsson, C. and Hagsand, O.: DIVE-a Multi-User Virtual Reality System. In Proceedings of the IEEE Virtual Reality Annual Symposium (VRAIS'93), September 18-22, 1993, Seattle, Washington, USA, pp. 394-400.

[12] Shaw, C., Green, M., Liand, J., and Sun, Y.: Decoupled Simulation in Virtual Reality with The MR Toolkit. ACM Transactions on Information Systems 11(3), 1993, pp. 287-317.

[13] Snowdon, D. and West, A.: The AVIARY VR-system. A Prototype Implementation. In Proceedings of the 6th ECRIM Workshop, June, 1994, Stockholm, Sweden.

[14] Tarr, R. and Jacobs, J.: Distributed Interactive Simulation (DIS). In Proceedings of the 1994 Summer Computer Simulation Conference, July 18-20, La Jolla, California, USA.

[15] Agrawal, P., Hyden, E., Krzyzanowski, P., Mishra, P., Srivastava, M., Trotter, J.: SWAN: A Mobile Multimedia Wireless Network. IEEE Personal Communications, 3(2), 1996, pp. 18-33.

[16] http://isw3.aist-nara.ac.jp/IS/Chihara-lab/

[17] Pyssysalo, T., Kinnunen, U., and Pulli, P.: Real-Time Distributed Virtual Reality. In Proceedings of the 2nd International Workshop on Mechatronical

Computer Systems for Perception and Action, February 10-12, 1997, Pisa, Italy, pp. 3-11.

[18] Ghai, R. and Singh, S.: An Architecture and Communication Protocol for Picocellular Networks. IEEE Personal Communications, 1(3), 1994, pp. 36-46.

[19] http://www.bluetooth.com.

[20] Onoe, Y., Yamazawa, K., Takemura, H., Yokoya, N.: Teleprecence by real-time view-dependent image generation for omnidirectional video strams. Technical Report NAIST-IS-TR97015, Nara Institute of Science and Technology. Nara, Japan. 20 p.

[21] Shionozaki, A., Yamashita, K., Utsumi, S., Cho, K.: Integrating Resource Reservation with Rate-Based Transport Protocols in AMInet. LNCS 1368. Springer. Pp. 206-221.

# Wireless Diagnostics and Control for Industrial Plants


Pekka Ala-Siuru, Esa Tuulari, Juha Takalo, Jyrki Leskelä,
Tapio Seppänen
VTT Electronics
P.O.Box 1100
Kaitovayla 1
FIN-90571 Oulu
FINLAND


## Abstract


In the future, many hardwired networks in contemporary industrial plants will be replaced by wireless networks. Portable terminals will be used in many tasks ranging from plant monitoring and repair jobs to personnel training and marketing. Our multiyear project examines concepts, application scenarios, and technology platforms for reliable and cost-effective implementation of these forthcoming wireless systems. Potential applications areas include wood, metal, paper, and construction industries. In this short paper we present an experimental small-scale infrastructure which has been under construction at VTT Electronics. This environment includes both wireless and wirelined networks, servers and monitoring stations with on-line diagnostic functions. We also present a virtual reality (VR) user interface for a diagnostic application which has been implemented and tested in this environment.

## 1. Introduction

In many industrial areas there is an urgent need to have better communication environments for more efficient handling of complex product information in real-time. The information in the most modern systems is typically multimodal and consists of multimedia documentation and complex system control information. In many industrial environments it is very expensive to build hardwired information networks. Moreover the need to solve production problems on-site in the plant environment raises the need for environments for fast mobile, ubiquitos computing.

For the purpose of testing modern high-speed wireless environments to meet the industrial needs we have constructed an experimental platform for embedded mobile multimedia (EMMI) communications. During 1997 we tested the network with a virtual environment KATE [1], which was built for a diagnostic application. The main work has been done in the EMMI project and in 1998 the environment is further developed.

## 2. The wireless experimental network

The experimental network consist of several multimedia servers, including powerful computers which control databases storing information of, e.g., repair histories, time tables, process control, product configuration, and technical manuals. Portable terminals enable personnel interactively utilize and control any information in the network (Figure 1).

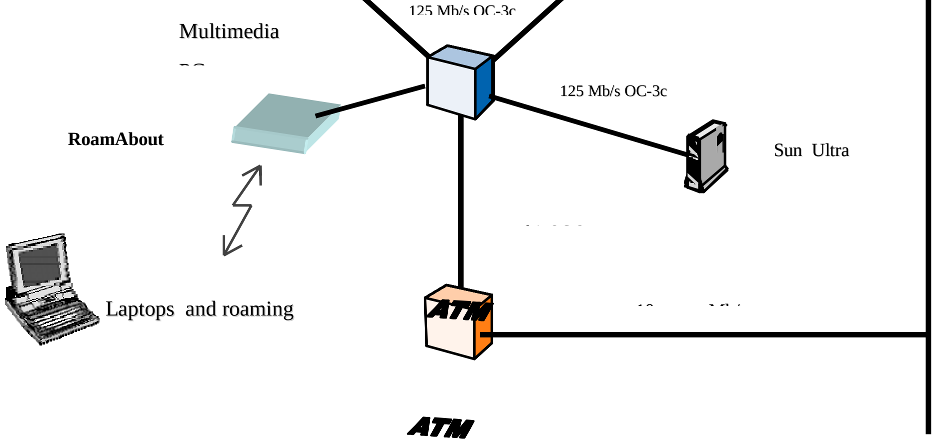



*Figure 1. Structure of the EMMI experimental network..*

In this environment nomadic users connect their terminals to the server through a spread-spectrum radio link. The server can also be contacted by a fixed workstation through an Ethernet local area network. A Java applet is downloaded to the terminal by a WWW browser. Execution of this applet enables the user to access suitable information in the server. The user can easily set up a session where he monitors in real-time the data being stored in the server.

## 3. The mobile virtual reality application

The application case is a maintenance serviceman system prototype for a steel rolling mill fault diagnostic application (KATE) which combines virtual reality (VR) modeling and multimedia documents. This system aims to provide more visual information to the user and enhance the education of maintenance and service personnel. The system also explains specific problems encountered in a process to the user. The wireless technique enables a nomadic user to be quickly and automatically notified of an alarming situation and to find explanations for complex faults. The general context of the application environment is depicted in figure 2.

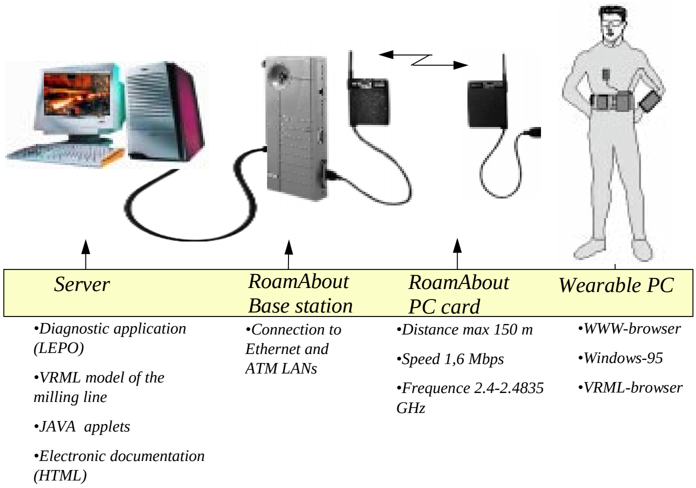



*Figure 2. The body wearable computer setup.*

The virtual model data and the other document material (part lists, technical drawings) are stored in the multimedia server and can also be used for teaching purposes, as exemplified by the FAST [5] system constructed in GTRI.[1]

Figure 3 depicts part of the virtual environment (motor room) combined with a sheet of specific signals concerning milling engine nr. 6 seen in the virtual environment (VE). The text area is a link where the explainer application can be started. The diagnostic explainer application LEPO is described in detail in [4].

[1] GTRI = Georgia Tech Research Institute

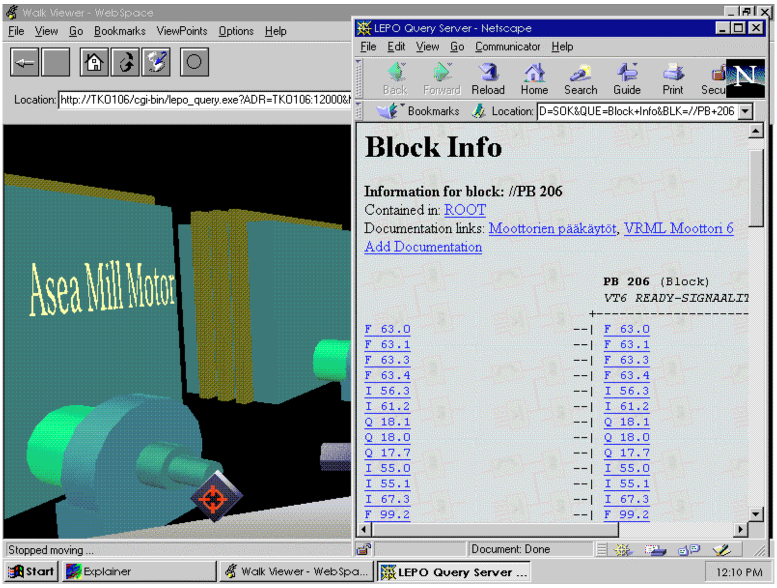



*Figure 3. VRML user interface for the diagnostic logic explainer application (LEPO)*

The virtual model user interface is written in the VRML modeling language. We integrated the constructed model with hypertext service manuals and part lists. The hypertext part is written in HTML and the interface between the virtual model and the diagnostic application is implemented in the Java language.

The service personnel can make queries to the server from the client machine which was in our first experiment a laptop 150 Mhz Pentium PC. Communication is implemented via wireless LAN (2,4 Ghz). This wide-spectrum link enables transactions up to 150 m. In different office and industrial environments the achieved working distance is about half of the theoretical maximum distance. For the transmission of this kind of complex information (VR model) the speed of the link (1.6 Mbit/s) was sufficient.

## 4. Implementation of the body wearable computer

The terminal used in the prototype application is based on a self-made body wearable computer (BWC) (Fig. 4). The implementation is composed of PC104 modules often used in embedded systems. A miniature TFT-LCD color display of 6,4" large and with VGA graphics capabilities (having a maximum resolution of 800*600 pixels) is attached to the users arm. This enables hands-free visual contact to the data. A trackball on the hip is used for controlling the menus. In addition, a speech-controlled interface is under development which will further enhance hands-free operation of the computer. The terminal is connected to the local network and servers via a spread-spectrum radio link. The wearable computer is belt-worn and it contains a 486DX100 processor with 8 MB DRAM. The computer consists of 3 PC/104 cards: CPU-card, I/O-card and PC-card. An 800 Mbyte harddisk is also included. The I/O-board has two RS232-ports and a parallel-port (like a normal PC).

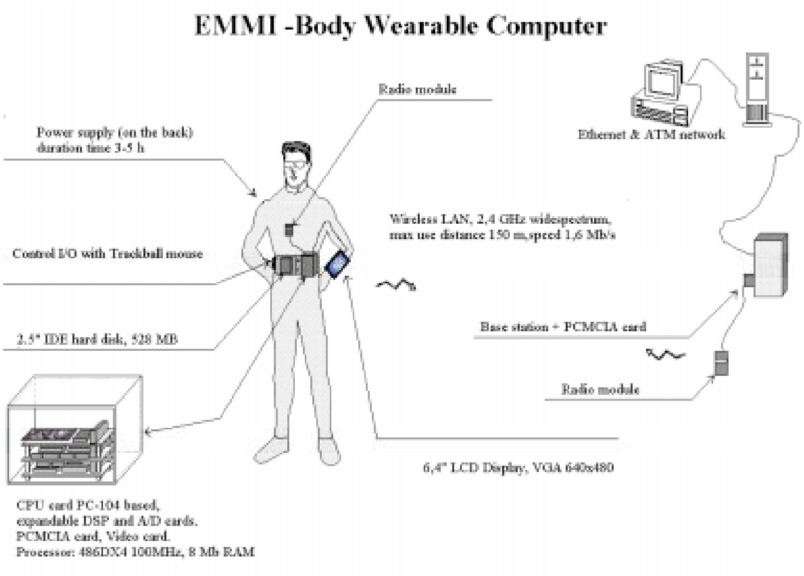



*Figure 4. The EMMI body wearable computer.*

The rechargeable battery consists of  8 NiCd-cells. Two operation voltages, 5 and 12 volts, are provided to the computer and display part. The 12 volt is needed for the LCD's backlight inverter. The capacity of the batteries is 7 A, which gives 5 to 8 theoretical hours of operation to the wearable computer depending on the use of the radiomodule. The battery package is worn in the back.

Windows 95 is used as the computers operating system. This made it possible to run 32-bit Java-applets via the Netscape WWW-browser. For browsing the VRML model we used the WebSpace browser.

## 5. Conclusions and future work

The expected benefits of using the wireless mobile environment in industrial plants were: 1) service person can move freely in the plant while monitoring the process, 2) the hands free operation of the computer frees the service person's hands for service or repair tasks, 3) running a web-browser in the wearable computer gives the service person wireless access to any information in the Internet or the plant's intranet.

The system described here has been tested in our wireless research environment. We intend to perform the tests also in the industrial, real environment. The industrial tests will not include the ATM components, but the wearable is connected to the steel mill production and office networks.  We also intend to test the wireless communication (at VTT) with more effective devices to achieve true mobile VR.  In our current prototype the distance between client and server was limited to 25-30 meters. We have ported the VE from the laptop environment  to our wearable computer. For frequent use the EMMI BWC is too heavy (3 pounds + battery 6 pounds), and the display location (on arm) is not optimal for use in industrial environments. The display may break when the user is moving around in cramped plant rooms. Another problem concerned the use time of the BWC is that the actual use time between loading was 3-5 hours. This should be increased by at least 2 hours for longer service periods in the field.

See-through glasses will be evaluated for the KATE service terminal system. This will expand the application area to an augmented system, such as described in [2,3], where document data can be overlayed with a view of the real plant environment. As an alternate input device we are going to test the Twiddler [6] chorded keyboard which is a pocket-sized mouse pointer plus a full-function keyboard in a single unit that fits neatly in either right or left hand. Voice input comprises another input method to be considered; we have selected the Dragon (Dragon Systems Inc.) system for our test environment.

## References


[1] P. Ala-Siuru, J. Takalo, J. Ensomaa, J. Plomp ”A hierarchical virtual environment for a machine fault diagnostic application”. To be published in Future Generation Computer Systems, VR Euro'97 Special Issue 1998.

[2] S. Feiner, B. MacIntyre, and D. Seligmann,” Knowledge-based augmented reality”. Communications of the ACM, 36(7), July 1993.

[3] S. Kakez, Vania Conan and Pascal Bisson, ”Virtually documented environments”, Proceedings of the 1st International Symposium on Wearable Computers (ISWC '97), IEEE, October 13-14, 1997,Cambridge, Massachusetts, USA; http://computer.org/conferen/proceed/8192/8192toc.htm

[4] J. Plomp, P. Huuskonen and E. Malm, ”Operator and Maintenance Support Through PLC Logic Analysis and Hypertext Documentation”, COMADEM '97. 10th International Congress and Exhibition on Condition Monitoring and Diagnostic Engineering Management. Espoo, FI, 9 - 11 June, 1997. Jantunen, Erkki (ed.). (1) (1997), 354 - 363.

[5] C. Thompson, Jennifer J. Ockermann, Lawrence J. Najjar, and Erika Rogers, ”Factory Automation Support Technology (FAST): A New Paradigm of Continuous Learning and Support Using a Wearable”, Proceedings of the 1st International Symposium on Wearable Computers (ISWC '97), IEEE, October 13-14, 1997, Cambridge, Massachusetts, USA, http://computer.org/conferen/proceed/8192/8192toc.htm

[6] http://www.handykey.com/over.html

# Session 2:

# "Architectures"

# An Architecture to Access Data Services Through Cellular Phone Networks

J.M. Rivadeneyra Sicilia and J. Miguel-Alonso
Universidad del Pais Vasco/Euskal Herrico Unibertsitatea
P.O. Box 649, 20800 San Sebastian, Spain

## Abstract

In this position paper, we present our research work focused on the design of a communication architecture to access TCP/IP networks (mainly Internet) through GSM. We exploit the existing and widely used TCP/IP communication architecture, but also take into account the specific features of wireless links and GSM networks. Our proposal is based in the indirect model of client/server interaction, where a intermediate element acts as a bridge between GSM and the rest of the data network. This interceptor provides enhanced functionality, improving the performance of standard TCP/IP applications in a mobile environment, maintaining the protocols in server machines, and requiring minimum changes in those running in the mobile clients. GSM operators are not required to change their network infrastructure in order to implement the architecture.

# 1. Internet access from cellular telephone networks

There is an increasing interest in using the cellular phone networks for data access, as it presents some advantages with respect to other options for data access:

1. Those networks already exist, and are increasingly familiar to the user; data transmission becomes just another facility of the mobile phone.
2. They are able to provide ubiquitous access to information.
3. Mobility issues are managed by the network.

On the other hand, the characteristics of cellular telephone networks are quite different from those of wired links. New problems arise for existing applications designed to run in fast and reliable connections. Currently we are working in the design and implementation of a framework to provide the functionality required to run standard Internet applications using a cellular system as the access network, with an acceptable performance levels. We will focus on the GSM [1] (Global System for Mobile Communications) technology for the wireless access network, as it currently forms one of the most comprehensive mobile phone systems, thus being able to provide access to data services to a wide range of nomadic population.

Departamento de Arquitectura y Tecnología de Computadores. Universidad del País Vasco / Euskal Herriko Unibertsitatea.
P.O. Box 649, 20080 San Sebastián, SPAIN. Tel.: +34 43 448000 E-mail: {pepe, miguel]@si.ehu.es

## A. Internet access using GSM

Nowadays, GSM supports circuit-switched connections, with a single error-correcting link per handset. The nominal transfer rate is 9.6 Kb/s, far from the 33.6 Kb/s of the wired telephone network (in fair conditions). Besides, actual sustainable transfer rates handled are even lower: slightly over 7 Kb/s for bulk data transfer over a good quality GSM link [2]. The use of all-digital connection from GSM to ISDN and data compression makes that rate to go up, but for some considerable time most traffic will continue to be handled by the PSTN.

Figure 1 gives an overview of the use of a GSM network to access Internet. Three different networks are traversed in the way from mobile client to fixed server (and vice-versa): GSM (digital, circuit switched), PSTN (analog, circuit switched) and Internet (digital, packet switched).

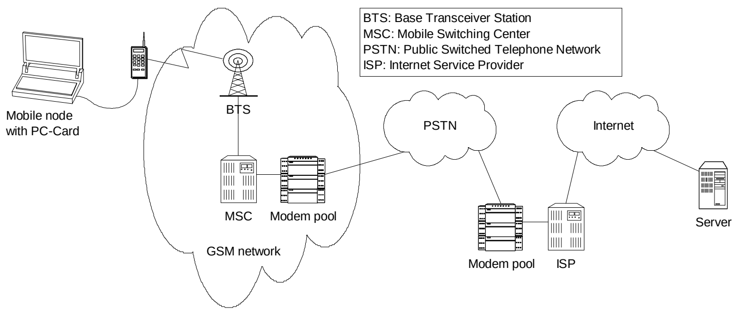



*Fig. 1. Conventional access to Internet through GSM networks.*

In this arrangement, the only "new" step is the first one. GSM specifies two modes of asynchronous bearer service: transparent and non transparent. In the first one the bit error rate on the radio link, within normal coverage, is assumed not to exceed the level of $10^{-3}$ at the line speed of 9.600 b/s and after mere forward-error correction. The non-transparent is implemented using the Radio Link Protocol (RLP), a member of the HDLC family. RLP includes error correction mechanisms, using selective retransmissions, which reduce the average bit error rate down to $10^{-8}$ [3]. The price to pay is an increase in latency while throughput and transmission delays become variable, remarkably so under poor conditions of the air link.

Figure 2 shows the protocol architecture used of this access. It is assumed, as it is common practice, that the non-transparent GSM data service is chosen.

Location updating, and subsequent call routing, use the MSC and two location registers: the Home Location Register (HLR) and the Visitor Location Register (VLR). The HLR contains the actualized position of every mobile user in the network. The VLR contains information about every mobile station present in a particular location area. Typically there is a VLR per MSC: the group of cells under control of a MSC defines a VLR location area. When a mobile station moves to a new location area, it must announce itself to the network by means of a location update message sent to the new MSC. The mobile is registered in the VLR and a message is sent to the subscriber's HLR to update the location information.

## B. TCP/IP applications over GSM

Several researchers have identified transport protocol problems that arise in wireless networks [4][5]. The problem relies on some design premises of TCP: it was intended for relatively fast and reliable fixed networks, and not for unstable, wireless networks. For this reason, some of the algorithms used in TCP to deal with congestion and error situations are inappropriate in a wireless environment. In fact, these mechanisms result in unnecessary waste of capacity and, at the same time, inefficient use of the (scarce) available bandwidth.

Another problem of classic TCP over cellular networks is the large amount of redundant control information in the segment headers, which wastes a good deal of the available bandwidth. Some techniques are already available to reduce the amount of control information in TCP headers sent through SLIP (CSLIP) or PPP.

## C. The indirect model

This model [6] allows to improve the performance of standard TCP/IP implementations in a mobile environment, maintaining the protocols in server machines, and with minimum changes in those running in the mobile clients. The main idea of the indirect model is simple: whenever two different environments, such as wired and unwired, are involved in a client/server connection, split the connection in two parts, one per environment. An intermediate system is located between the two connections, participates in both of them, and manages each one in a different way, suitable for its characteristics. The intermediate element looks as a server to the mobile client, and as a client to the fixed server, so the classical client/server model becomes a client/intermediary/server model. A datagram traveling from client to server (or vice versa) will traverse some layers in the protocol stack of the intermediary before being forwarded to its destination. The number of layers above the network layer traversed at the intermediary depends on the nature of the application. The applicability of the model can be exploited at any layer, but its benefits are specially appreciated at the transport [7] and application layers.

## 2. Proposed communications architecture

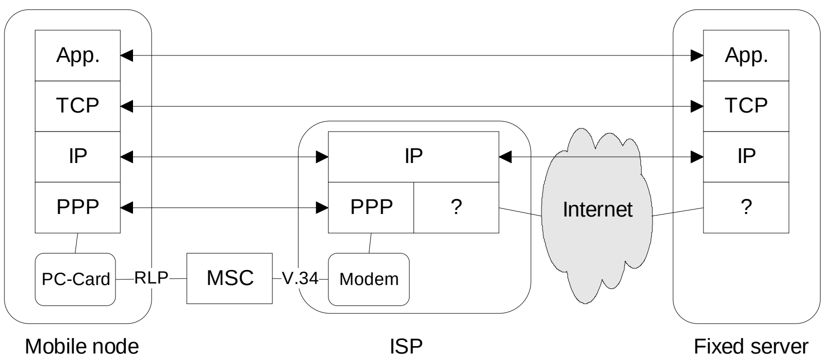
The proposed design has the following targets:

- Compatibility with TCP/IP protocols, making client mobility transparent to the common TCP/IP user.
- To establish a framework for the development of new applications, or the adaptation of existing ones, in mobile environments.
- To improve the quality of the data service of the network as it is perceived by the user, improving the throughput of network protocols. The quality of service should be close to that achievable from wired telephone networks.
- Scalability to afford new services and the increment of demand that accompany the spreading of GSM networks.
- GSM operators are not required to change their network infrastructure in order to implement the architecture, even when the new GSM standards for high speed data service (GPRS and HSCSD [8][9]) are available.

Our proposal consists of an indirect architecture, with client-server communications divided in two parts: the first one for the cellular network, controlled by special network protocols adapted for the wireless environment, and the other one for the rest of the connection, managed by conventional TCP/IP protocols. In the frontier of both sides is the intermediary, known as GSN (Gateway Support Node) using GPRS conventions. There will be one or more GSNs in the cellular network, and the user will be connected by a circuit switched connection to one of them (it will be this way until GPRS is available). The rest of the communication, from the GSN to the server, uses packet switching. Therefore, in this proposal the intermediary is not located in the border between the wireless and wired environments (as it is usual in indirect proposals for mobile computing) but in the border between circuit and packet switching environments (when GSM data services are circuit switched) or, more generally, between the GSM network and the Internet. Next we will describe, using a bottom-up approach, the different layers in the architecture for all the constituting entities, shown in Figure 3.

## *A. Network access*

Taking Figure 2 as the reference point, no changes are proposed at the lowest layers: PPP and RLP protocols are used to access the GSN. The main difference is that in this arrangement, the GSN also acts as an Internet Service Provider, thus simplifying the user access to the Internet and increasing the value added by the GSM operator. However, this is just an option. Other ones are to locate the GSN and ISP in the MSC (avoiding the use of the PSTN), or to distribute all the functions (MSC, GSN and ISP) along separate elements, with the last one out of the GSM network. Which configuration to use is a question to decide by the operator; one of our tasks is to evaluate their relative advantages and problems.




B.

## Mobility management

*Fig. 2. Protocol architecture of a conventional connection to Internet through GSM.*

The mobility management procedures described in Section 2 are enough to cope with the common scenario of a mobile client who establishes a connection with a fixed server. First, the client will connect with its ISP and will get a dynamic IP address. Then, IP datagrams will flow through the circuit switched connection (or will be packet switched using GPRS) to and from the mobile client. The maintenance of that connection is guaranteed by the GSM handover mechanism.

In the case of a user who wants to access the Internet with a fixed IP address, the above solution is not appropriate. In such situations incoming datagrams cannot be correctly routed through the appropriate GSN, as the destination address is not belonging to the set of addresses managed by the GSN. Those datagrams will be routed to the home network of the mobile station. For those situations is necessary to use Mobile IP [10]. The main difference with HLR and VLR functionality is that these GSM registers are used for location updating and call routing into the GSM network, and mobile IP is used for dynamic routing in the other side of the GSN, this is, the Internet.

## C. Transport layer

In the GSM part of the connection, a new, simplified version of TCP replaces the standard TCP (to be implemented; the proposed name is STP, Simplified Transport Protocol). The basic challenge for STP is to maximize the throughput in the connection, efficiently using all the available bandwidth. The main limitation is the modest performance and irregular behavior of the wireless link, so the protocol overhead must be minimized, and the flow control mechanism must be tailored to the characteristics of the link. STP will provide communication services to client applications in the mobile node and to proxy agents which control the execution of filters in the GSN.

The design objectives of the STP protocol are high performance, efficient recovering from expected and unexpected disconnections, and tolerance to delay fluctuations. The indirect model allows us to implement the required functionality as a simple, optimized point to point protocol. The combination of STP and TCP in both sides of the GSN constitutes an implementation of indirect TCP.

### D. Proxy layer

The use of the indirect model at the application layer means to interpose application specific programs, called filters, in the middle of client-server connections, running in the GSN. These filters, and its management, is what we call the proxy layer.

There can be general purpose filters (those that are supposed to perform well for any application) and specific filters (specifically designed for a particular application). In fact, the transport entity running at the GSN can be seen as a general purpose filter; it is just a question of naming. Actions that filters might take include selective dropping of superfluous header information (e.g., in e-mail), delay data for later pickup (e.g., POP mail), re-segmentation of data, reducing "frivolous" traffic addressed to the mobile host (e.g., replying ICMP echo requests). All of these advantages can be achieved without the use of intermediaries, but in most cases client, server or both would have to be changed. In an indirect architecture like this, changes are few and just in the client, not in the server.

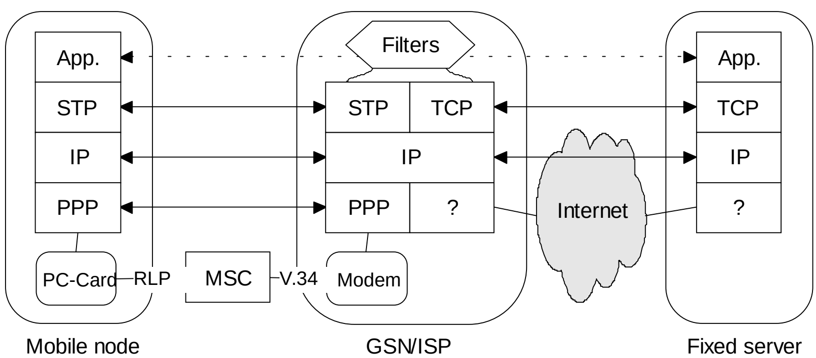



*Fig. 3. Protocol architecture with an intermediary located at the GSN.*

## E. Adapted protocols

The action of filters is not the only manifestation of the indirect model at the application layer of the architecture. It can support the option of mobility awareness to applications (i.e. non-transparency), allowing the use of modified application protocols in the client-proxy connection, always maintaining standard protocols in the proxy-server side. The WWW is an application where this approach suits particularly well: several efforts are devoted to the development of simplified versions of HTTP protocol [11] which would be used by specific clients (for example, a personal digital assistant) when connecting to specific proxy servers running in the GSN.

# 3. Current work

Our current efforts are mainly devoted to the design and implementation of the Simplified Transport Protocol for the mobile-intermediate part of the connection. A main objective of that work is to guarantee an easy integration in the most commonly used environments for Internet access, not requiring the installation of a new version of the operating system in the client machine.

Other areas to be investigated are:

- The recent evolution of GPRS and HSCDS standards, and its future integration with our architecture, with special attention to possible changes in the bottom layers of it.
- The number of GSN to install in a GSM network, and the optimal ubication of them.
- The integration of mobile-IP home and foreign agent functionality in the GSN.
- The strategy for filter managment, considering the interaction between client and proxy. It supposes the design of an application layer protocol for it.

Acknowledgements

This work has been supported by Gipuzkoako Foru Aldundia under grant number OF 134/1997.

## References

1. M. Rahnema. *Overview of the GSM System and Protocol Architecture*. IEEE Communications Magazine, 31(4) pp. 92-100, April 1993.

2. T. Alanko, M. Kojo, H. Laamanen, M. Liljeberg, M. Moilanen, K. Raatikainen. *Measured Performance of data transmission over cellular telephone networks.* Computer Communications Review, 24(1994)5.

3. ETSI. *Quality of service*, GSM specification 02.08, Version 3.0.0., March 1990.

4. H. Balakrishnan, S.Seshan, R.H. Katz. *A comparison of mechanism for improving TCP performance overwireless links* . SIGCOMM 96, August1996

5. R. Caceres, L. Iftode. *Improving the performance or reliable transport protocols in mobile computing environments.* IEEE JSAC Special Issue on Mobile Computing Networks, 1994.

6. B.R. Badrinath, A. Bakre, T. Imielinski, R. Marantz. *Handling mobile clients: A case for indirect interaction*. In 4th Workshop on Workstations Operating Systems, 1993.

7. A. Bakre, B.R. Badrinath. *I-TCP: Indirect TCP for mobile hosts*. Rutgers University DCS-TR-314, 1994.

8. Italian Cellular Home Page-GPRS.
http://www.venus.it/cellular/dati/gprs/index.html

9. Ericsson Radio Systems AB. *The mobile multi-media kit*. 1996. http://www.ericsson.se/systems/gsm/future.html

10. M. Kojo, T. Alanko, M. Liljeberg, K. Raatkainen. *Enhanced communications services for mobile TCP/IP networking.* University of Helsinki, Department of Computer Science, Series of publications C, No C-1995-15, Apri1995.

11. M. Liljeberg, H. Helin, M. Kojo, K. Raatkainen. *Enhanced services for WWW in mobile WAN environment*. Proceedings of the IEEE Global Internet 1996 Conference, London, England, November 20-21, 1996.

# Utilizing the Web to Deliver Application-Level Functionality

P. P. Sage, P. Milligan
The Queen's University of Belfast,
Belfast BT7 1NN, Northern Ireland,
United Kingdom

## Abstract

In recent years the Web and associated technologies have emerged as a viable means of publishing document-related material for mass consumption. The subsequent increase in the use of this medium has encouraged access to remote services and functionality from any point on a network. This paper examines the technologies behind mobile computing and presents alternative solutions to the provision of remote access to an existing MIS (Management Information System) application.

## 1. Introduction

The past number of years have seen a dramatic increase in the popularity of the Web as a mechanism for the delivery of document-based material, and, more recently, application-level functionality. The latter has, in many cases, been realized using a combination of the HTML forms construct and CGI programs which facilitate a simple forms-based level of Web page interaction, generally used for data entry and database query tasks. However, the emergence of the Java programming language from Sun Microsystems[1] has initiated a great deal of interest in the provision of more intricate Web-based application-level functionality, in the form of Java Applets[2].

The “Java Revolution” has indeed, been taken seriously by many software providers as a number of familiar packages are about to make their Java debuts. This paper describes a number of approaches investigated in providing web access to a commercially available database-oriented Management Information System or MIS product. The requirement to provide such functionality was largely generated from within the product user base.

Although the product provides specific MIS functions such as timetable generation, attendance recording and resource management, the techniques discussed apply directly to any database-oriented application.

## 2. System Structure and access requirements

The system, written originally for Microsoft Windows, is designed to provide a range of Management functions, such as timetable generation, attendance recording and room booking, based around a centrally managed data set. In normal usage the software is manually installed on client machines with network access to the data source. The provision of remote access to more intricate services and functionality would require modifications to the structure of the system, in particular the introduction of a client-server architecture facilitating multi-user access.

Prior to the emergence of Web technologies the development of client applications has largely been governed by platform-specific issues with different versions appearing for each targeted architecture. Distribution and installation of such software is usually a manual activity. Platform-dependent code is either installed on a machine-to-machine basis or is accessed via a distributed file system, with subsequent updates and general maintenance forming an iterative process.

The development of Java technologies, in conjunction with the Web, has provided a mechanism for the automatic and efficient distribution of fully functional, Graphical User Interface (GUI) based programs in the form of Java Applets. In addition, Java support for socket programming makes the development of network-dependent programs relatively straightforward. This, together with the ease of Web-based distribution, makes Java an attractive option for the development of client/server distributed applications. Client tools written in Java can be centrally maintained on a Web server and accessed through use of a Web browser such as Netscape which supports a Java runtime environment.

The usage of Applet technology in providing remote access to functionality such as timetable generation, requires a minimum number of components :

- Client Software (Java Applets) responsible (in general) for managing data requests from the user, obtaining data from the remote source, and visualization of results.
- Data Server responsible for managing requests from client programs, and interaction with MIS functionality.
- The database (MIS) application, providing functionality and data.
- A Web Server and associated web pages which provide the entry points to exported services.

Hence, general functional requirements for remote access can be summarized as follows :

Access to remote functionality should achieved simply by using a web browser to obtain the appropriate service page which downloads a service Applet. Once active, the service Applet responds to user data requests by connecting to the data server and issuing requests through a recognized protocol, subsequently displaying the response. The next stage is to determine, if any, tools which support this approach.

## 3. A Tool-Based approach

In addition to the standard Java Development Kit (JDK) support from Sun Microsystems, a number of Java development environments have appeared on the market, Visual J++ from Microsoft and Visual Café Pro from Symantec [3], among others. These tools are aimed at emulating the success of such tools as Visual C++ and Visual Basic in providing a complete environment for Java application and Applet development. In addition, Symantec's Visual Café Pro (VCP) is promoted as offering a solution to accessing remote data sets via Applets. Consequently, this development environment was selected to develop the remote access functionality required of the database application in question.

Included with VCP is a data server package called dBAnywhere designed to provide remote access to database tables. Through suitable Applets, SQL queries may be constructed and transmitted to a dBAnywhere server which performs required selections, returning any results generated.

This approach only partially fulfills our access requirements, as not only is it necessary to access system data, but also its functionality. For example, in the timetable generation service provided by the MIS application, timetables are not statically stored but must be generated dynamically from user input against raw data in the database tables. Therefore, the ability to extract data from a remote data source must be supplemented by the application of a timetable generation algorithm (in this case), implemented within an Applet.

In this schema, desired functionality must form part of the Applet code. This fact has a significant impact on the structure of an Applet providing access to a service. In addition to managing user requests for information and its presentation, the Applet must be responsible for the manipulation of the data received from the server into the desired form. In the case of timetable generation, this can place an unacceptable load on an Applet. Therefore, computationally intensive algorithms are best avoided in Java as bytecode interpretation is an order of magnitude slower than native code, even when JIT compilers are employed.

This method of access when implemented and tested, proved to be unbearably slow in the response to the user. The response times for a simple query resulting in a small amount of data (2 rows of five columns) is summarized by the following table :

| Browser | dBAnywhere Response | Timetable Generation. |
|---|---|---|
| Netscape 3.0 | 38.0 Seconds | 24 Seconds |
| Explorer | 38.0 Seconds | 20 Seconds |

The software was tested on a isolated, local network where the server was a Gateway Pentium Pro with a Fast SCSI Disk, 64 MB Ram, and Fast Ethernet connection; and the client was a Gateway Pentium 166 with Fast SCSI Disk, 32 MB Ram, and Fast Ethernet connection.

## 4. Back to the drawing board

The unsatisfactory experiences with a third party data server solution resulted in the development of a client-server architecture aimed at restricting Applet activity to user input and data visualization with computationally intensive functionality retained at the server side. This would enable access from fairly lightweight client machines with limited specifications. The main server code, implemented in Java makes use of the concurrent features (threads) of the language to maximize response time. The server is composed of a main server thread with the responsibility for managing new client requests. When a client Applet requests data the main thread creates a “Service” thread which subsequently handles all communication with the Applet for the duration of the connection. In this way Applet requests can be responded to almost immediately, and numerous queries can be handled concurrently. To overcome Java’s performance problem, any computationally intensive code, like timetable generation, is implemented in native C++ code called from Java. Furthermore, the success of this approach resulted in the creation of a native API which can be used to directly access the functionality of the original MIS application. This has the following advantages :

- Using existing native code, while increasing efficiency, minimizes the scope for error and conflicting results from the standard and web-based versions.
- Since the client Applet does not manipulate database tables directly, but requests information from a server through a protocol, a high level of security can be maintained over potentially sensitive data.

Test results obtained (for the same data set and query used in the previous tests) have shown a marked improvement over the previous method :

| Browser | Time to Request and Display a Timetable |
|---|---|
| Netscape 3.0 | < 2 Seconds |
| Explorer | < 2 Seconds |

The system is currently undergoing further tests on a full network.

## 5. Conclusion : The suitability of Java

From its initial launch in May 1995 Java has increased in popularity within both the academic and industrial communities both as a teaching language and a viable development platform. However, rapid changes in the language specification and subsequent re-releases of the Java Development Kit have cast a small shadow over Sun Microsystems " … write once, run anywhere .." ideology.

Web Browsers tend to "lag behind" in their support for Java technologies; most only support Java 1.0 programs, with JDK1.2 available as the most recent development kit. Netscape have, however, released support for Java1.1. So, having more constructs available in the language specification does not always translate into Applet functionality. With each subsequent release of Java, the amount of code in the support libraries increases dramatically :

JDK 1.0 : 8 packages, 211 classes, 1753 methods
JDK 1.1 : 22 packages, 477 classes, 4424 methods
JDK 1.2(beta): 60 packages, 1546 classes, 14622 methods

This increase in size has a knock-on effect for Java support within browsers. For Applets written in Java1.1 to run effectively within a browser, the machine on which it executes should have at least 32 MB of RAM. An increase of some 38 packages between release 1.1 and 1.2 will hardly help the situation, as the price of browser support will require machines with a more significant specification.

Java is an exciting and challenging development in information technology, though implementing a software product using the current technology will ultimately require a rewrite or revision. Perhaps the best version of the JDK to implement in is 1.0 as it is the most stable, and has good browser support. JDK 1.1 is not as stable as some would like, while JDK1.2 is too new and too large to be concerned with at the moment.

## References

1. Sun Microsystems, Inc., 901 San Antonio Road, Palo Alto, CA 94303 USA

2. The Java Language Specification James Gosling, Bill Joy and Guy Steele Published by Addison Wesley

3. Symantec Corporation 10201 Torre Avenue, Cupertino, CA 95014

# Design of An Applications Programming Interface for Mobile Applications

Rogelio Valencia, Jesús Villadangos, and J. Ramón González de Mendivil
Universidad Pública deNavarra
Campus Arrosadía, 31006 Pamplona (Navarra)
Spain

## Abstract

This paper describes the detailed design of the MAP Applications Programming Interface for mobile applications. This API has been designed to aid the developer to achieve a rational use of the bandwidth available to the mobile, to let the application make use of context related information that could be useful to modify its behavior depending on location and on network quality of service. The API implements these goals through the use of a distributed object environment, which separates the application in two different parts, one resident on a server in the wired part of the communications link and the other embedded on the mobile device. Hand-off and fault-tolerance are achieved through the use of a communication link between the different servers on the wired link.

## 6. I. INTRODUCTION

Due to the current advance in mobile computing technology, new exciting possibilities lie ahead for the computer applications developer to support and enhance the adaptation of his/her designs to user's mobility.

In parallel to these possibilities, some drawbacks are present in these new user environments due to the own characteristics of current mobile communications infrastructures.

In order to help the applications developer take full advantage of these new possibilities and to cope with some of its drawbacks, we are developing an API to facilitate the effort of adapting to a mobile environment [Valencia, 97a, 97b].

Other research groups are developing some work on this subject. Some provide support embedded directly into the operating system [Noble, 95], and some others provide it by means of an API [Kaashoek, 96]. For a very good summary of other projects related to this research area see [Popek, 95].

## 7. II. System's Design

MAP was designed to work under the constraints offered by the environments depicted in figure 1:

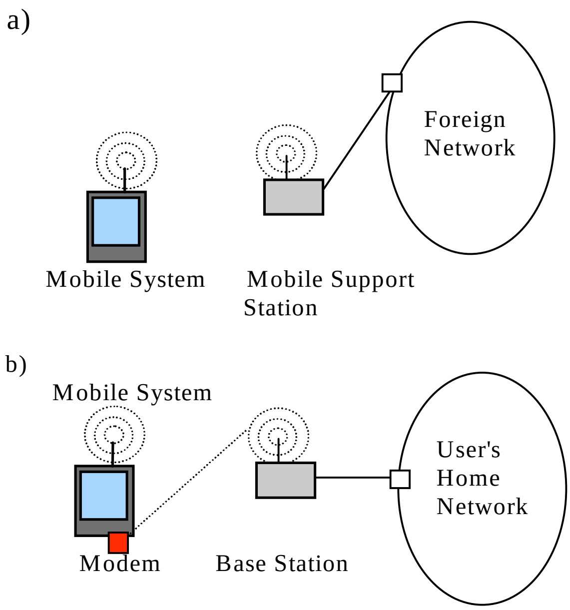



*Figure 1. Operation Environments*

These constraints consist of:

- A scarce bandwidth in the wireless interface, that suffers from hazardous disruptions.
- There are some limited resources on the foreign network available to support the connection of mobile systems.
- User's mobility makes the connection change its context constantly.

The main goal besides bandwidth saving, is the provision of means to let the applications be context aware, in order to fully exploit the possibilities offered to them by the user's constant change of location (new information available, new resources offered, etc.).

The design decisions behind these goals are:

In order to use bandwidth properly, the number and size of the messages used by the API's communication protocol are minimized.

To reduce the impact of the API support by foreign computing facilities, the use of "light weight objects" is enforced into the own API implementation and on the user's API use.

The integration of context awareness into API developed applications is provided by the use of the programming primitives for mobile computing introduced by [Wilcox, 96].

Information about the constant changes reflected on the system's connectivity is incorporated through the provision of a quality of service primitive, that could be used by the new applications to manage disconnection properly [Satyanarayanan, 90].

All these restrictions and features are combined within the API's architecture to conform a solid infrastructure for the development of mobile computing applications

## III. API Architecture

To achieve all these goals, the API defines the means to permit the applications developed by its users be divided in two parts, one located on the mobile system and the other located at a supporting entity on the foreign network being visited by the mobile.

The purpose behind this division, is the best utilization of the communication resources available to the mobile. The API transmits part of the application to the cooperating entity in order to exploit the better bandwidth available to it. The developer of the application through some API’s procedures specially designates the part of the application transmitted to the supporting agent. After being accepted by the supporting entity, the rest of the application, resident on the mobile, ask for service to the software part of the application installed on the wired part of the communications link, optimizing in this way its access to the rest of the network.

In order to permit a better control on the software parts that must be distributed to the supporting infrastructure and to facilitate a later communication with them, the system is designed for an object oriented programming paradigm.

As has been indicated , the API consists of two very well defined parts:

- A software entity located on the mobile that controls the objects that must be distributed to the available supporting agents. This is called API Server on the design.
- Another software entity that receives the objects distributed by the API Server, called the Content Server. This server accepts the objects sent by the mobile and invokes the methods called on them

These parts are shown in the next figure:

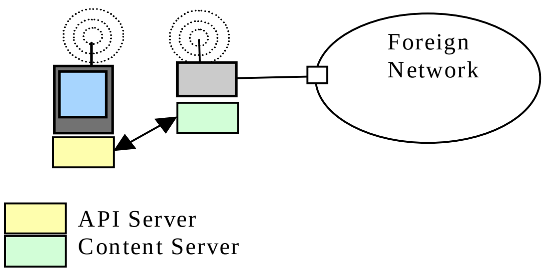



Figure 2. System's Architecture

We now describe the different parts of each of the two types of servers that constitute the MAP API.

1) API server

The parts that constitute this server are:

- The object distribution, and control subsystem.

This element is the one that preserves the state of the objects delivered by each application. It maintains the state of all the request being done to each object. This module is also in charge of the distribution of the objects whenever the system gets in contact with a content server. After doing so, it keeps track of all the request done, of those already fulfilled and of those being served.

- Inter-server communications subsystem.

This module is the one in charge of those actions related to the localization of the content server and of those related to the hand-off procedure.

- Context awareness subsystem.

This is the place were all the information relative to the quality of service of the wireless link, the current localization of the mobile and the discovery of the services available locally to the system are managed and converted to the specific needs of the objects requiring them.

2) Content server

Besides the those modules described above, the content server also includes:

- Object activation subsystem.

This module receives the request from the object's control and distribution subsystem and sets up a new thread of execution for each one, permitting in this way the management of concurrent transactions.

- Session management module.

These module coordinates with the inter-server communications subsystem to establish the environment for each new session based on the current availability of resources at the server.

The issue of security is not really addressed here because our first targets for the deployment of the API are the Mobile IP protocols developed by the IETF. This internetworking protocols manage a whole set of services to identify and authenticate each of the messages interchanged by the mobile and its foreign agent (the receiving entity at the new network the mobile is visiting).

Probably some security mechanism might have to be provided for the mobile to use other types of internetworking structures. But keeping in mind the end-to-end principles of communication [Saltzer, 84], no implementation is provided.

## IV. API Operation

The basic operation of the whole system consists of the following development steps:

1. The user constructs its program using an object oriented programming language (the current API context is being developed using JAVA, because the language provides some basic capabilities that make the API design easier, like the presence of threading, object distribution, reflection, serialization, etc.).

The application is composed of various objects and some of them are built in a client-server fashion. These client-server or distributed objects are built in such way in order to leave the client part of the object on the mobile system, and transmit the server part to the content agent where it will perform the functions solicited by the client left behind.

After program design, the whole code is compiled and tested. During run time, the program will run with the API server support.

2. At the moment of the program startup, the API server receives the server part of the distributed objects and waits for a content server to appear. Meanwhile, all request are treated in a local way by the API server. The processing done by the API server help the application believe it is connected because each request is accepted whether the mobile system is connected to the network or not. The server will perform the specified operation if network connectivity is available, or hold the request until it finds a mobility server to get access to the network.

3. Whenever the mobile arrives to a new network it connects to some form of communication server. It then waits for a content server announcement of availability and establishes an application session.

4. The mobile establishes a new connection to the content server. This process validates the mobile that is soliciting new support services and sets up the different options available for the management of the session, like sessions lifetime, etc.

5. After session establishment, the server part of the distributed objects is transmitted to the server where the content server controls it in order to preserve all the state related to each object. The client part of the distributed objects is now capable of sending messages to the server part, now located in a wired zone of the communication chain.

6. Whenever information is available for the clients, it is sent back through the API server communications channel.

7. If the mobile keeps on moving, its connection to the content server will be lost. In these cases, the content server will wait for the mobile to reconnect to it, or to receive a notification specifying that the mobile has reconnected to another content server. In this last case, the previous content server will send the current state of all the objects, avoiding in this way the requirement that the mobile sends this same information to its new server, preserving its available bandwidth.

*A. User Interface*

The API defines very simple operations to permit the user access its objects distribution capabilities.

1) Api commands

The first set of commands includes those that permit the application define distributed objects and access their methods later on.

// Object definition

MAP.MobileObject (Object object, MAPHandle handle);

// Method activation

handle.method (parameter1, parameter2, ... );

To make our application context-aware the API implements the following methods.

// Mobility primitives according to [Wilcox, 96]

boolean MAP.atPlace (Place place );

boolean MAP.atTime (Time time);

// QoS and Services primitives

MAP.getQOS (QOSObject);

MAP.getServices (service ObjectList);

The first two methods inform the application if it has reached a determined position or time mark. A specific action may be executed at a certain place, and/or time by using these to service primitives.

The third method just provides a measure of the quality of the wireless link, as perceived by the server, or by a combination of the measures taken by the server and the mobile.

This helps the application to determine the amount of bandwidth available to fulfill the requests of the user. If the user is requesting a video service through the wireless link, and the application detects a diminishing performance of the communications link, it may decide to request less frames per minute, turn to a black and white broadcasting mode or put another kind of output in the user's viewing window like a still image, or a local animation.

This parameter is provided as a percentage of the maximum rate transfer detected by the server and is complemented by the delay measures detected during the transmission. A 100% QOS implies that the application is dealing with a channel in very good conditions, and a 0% would imply a total disconnection from the network.

The last method provides the application with a list of the services available to the mobile in the current network. It will implement the service announcement protocol being developed by the nomadic technology group associated with the IETF Mobile IP Working Group.

## 8. V. Conclusions

The current design of the MAP API is based on a previous much more simple prototype that helped to find out the most important features of the inter-server communications subsystem [Valencia, 97a].

Even though the MAP API is in the middle of its development process, the key ideas behind its current design seem solid enough to guarantee the preservation of the scarce bandwidth available at the wireless link of the communications structure for mobile systems.

MAP also tries to prevent the API support servers located in foreign networks from expending lots of resources in this supporting activity, by using light-weight objects and a session limited by the specified parameters under the server's control.

The key component of the whole design is the inter-server communication mechanisms, which prevent the mobile from performing the transmission of information that is already on the network. And by specifying a period of time where previous servers may still service requests for mobile not directly under their control, avoiding the overload of application servers located throughout the network.

## References


1. Joseph, D., Kaashoek, M. F., Building Reliable Mobile-Aware Applications Using the Rover Toolkit. In Proceedings of the 2nd ACM International Conference on Mobile Computig and Networking. [November 1996].

2. Noble, B. D., Price, M., Satyanarayanan, M. A Programming Interface for Application-Aware Adaption in Mobile Computing. Proceedings of the 2nd Usenix Symposium on Mobile and Location-Independent Computing. Ann Arbor, Michigan April 10-11, 1995, 57-66.

3. Popek, G. J., Mobile Applications Support.. In Proceedings of the 1994 Mobile Information Systems Applications Workshop. Tysons Corner, VA, 7-8 (Dec. 1995).

4. Satyanarayanan, M., et al., Coda: A Highly Available File System for a Distributed Workstation Environment. IEEE Transactions on Computers 39(4) (Apr. 1990).

5. Saltzer, J. H., et al., End-To-End Arguments in System-Design. ACM Transactions on Computer Systems 2(4) (Nov. 1984).

6. Schilit, B., Adams N., Want R., Context Aware Computing Applications. In Proceedings of the 1994 Workshop on Mobile Computing Systems and Applications. Santa Barbara, CA, 8-10 (Dec. 1995).

7. Valencia, R. R., Building an Environment for the Development of Mobile Applications. In Student Poster Competition of the ACM's SIGCSE'97 Conference. San Jose, CA, (Mar. 1997a).

8. Valencia, R. R., González de Mendivil, J. R., MAP an Applications Programming Interface for Mobile Applications. In Proceedings of IASTED Conference on Parallel and Distributed Computing and Networks. Singapore, (Aug. 1997b).

9. Wilcox, D., Gruia-Catalin, R., Reasoning About Places, Times, and Actions in the presence of Mobility. IEEE Transactions on Software Engineering, Vol. 22, No. 4 (Apr. 1995).

# Aspects of Embedded Web Service for Remote Control and Monitoring

Emelianov M.M., Sheinin Y.E., Ignatiev M.B.
St.-Petersburg State University of Aerospace,
67, Bolshaya Morskaya, 190000 St.-Petersburg, Russia

## Abstract

Wireless communications and unified resource access are the main principles of today networking. There are many areas of remote control and monitoring implementations. We consider remote access to embedded systems that control industrial or aerospace objects. Web technologies may give us not only a unified flexible interface for full and clear representation of information on embedded systems and controlled objects, but means for remote system monitoring and management. Mobile computing is the largest area that demands not only protocols and interconnection tools but host situated software and hardware enabling mobile access for control and monitoring. Global networking for remote access to embedded Web servers is considered.

St.-Petersburg State University of Aerospace Instrumentation
67, Bolshaya Morskaya, 190000, St.-Petersburg, Russia
Phone: +7 (812) 210-7044 Fax: +7 (812) 315-7778
E-mail: mark@mark.stud.saai.ru, kira@topol.stu.neva.ru

Microprocessing Technologies
67, Bolshaya Morskaya, 190000, St.-Petersburg, Russia
PO box 84, 190000, St.-Petersburg, Russia
Phone: +7 (812) 210-7094
E-mail: ysh@mt.spb.su

## 1. Introduction

The embedded systems are systems, which are not programmed directly by the user [1]. Embedded systems are used in various fields - from power plant to control systems for a car, house systems. Important applications for embedded systems are industrial systems, control systems for complex industrial objects. Embedded systems usually are specialized hardware-software complexes, with built-in operational environment, closed architecture, specific interfaces and complex organization of an external and internal data flow. For such industrial objects as power turbine they are complex multiprocessor distributed systems [2]. The basic function of an industrial embedded system is to pick up, accumulate data about the state of the object, process them and to control object operation. Embedded system should also present an object's state information to operators and bring in instructions from an operator. The user interface is necessary for embedded systems to display information and to interact with the operational personnel. The complexity of user interfaces development for embedded systems makes it rather labor consuming. Hardware-dependent components, specialized units are often used in developments of such interfaces. There are also problems with the distribution of the software to client workplaces. As the number of embedded systems in the world is growing very fast it requires new approaches in embedded systems operation, especially in distributed environment. Also we consider that remote control and monitoring for mobile users is required.

## 2. Embedded Systems And The Web Technology

### *A. Areas of implementation*

Nowadays one of the most popular and general technologies for information resources access and information representation is the World Wide Web technology (Web technology) which is widespread in the global computer network Internet [3]. The main characteristics of the Web technology are: client stations platform independence, precisely specified algorithm of the information interchange (HTTP) [4], precisely specified format of data representation

(HTML) [5], server part low expenses on data granting. The Web-technology allows not only to represent information, but to carry out interaction with client also. The standard methods of Web-navigation can be used for the control and monitoring of embedded systems and controlled entities.

The Web technology is an efficient and cost-effective technology for embedded systems interface development and remote object monitoring and management. It gives various standardized means for user interface organization: display of the text (including formatting), various types of tables, lists etc., display graphic images (for example, of a block of the controlled entity). Hypertext references facilitate easy navigation on pages, data input forms and protocol of data sending to server give a way to forward actions from operator to the controlled entity. Execution of applications (applets and scripts) on Java language at the client part can carry out specialized functions.

The Web-based interface allows visualize any process, element or information in a form, which is convenient for an operator. It is simple and clear for a user, which is not a computer expert. He will receive unified access to control devices, will operate complex system via Intranet/Internet.

Standard browsers can be used as client programs - the problems of the information display on client side is already solved by Netscape and Microsoft. The Web technology makes the development of a particular user interface for an embedded system rather simple. The particular user interface coding can be placed at the embedded system itself, thus the problem of software distribution is solved too. Java applets in the Web interface makes it an advanced dynamic user interface. It improves user software functionality but do not require special tools for distribution, [6].

In distributed systems and in remote control and maintenance tasks the Web applications can be used instead of the out-of-date applications based on TTY/Telnet. Such applications facilitate remote configure, debug of an object, remote access to resources, remote help, remote support, fast update of embedded system software, data uploading and downloading

B. Industrial implementations

An example of industrial application for embedded Web-service technologies is the control system of the power station turbine [2]. The turbine and its control system require an external monitoring by operational personnel. It can be reasonable to monitor the system in a remote mode, thus reducing the requirements to quantity and qualification of the personnel, which operates the turbine at the place where the turbine is situated. In some cases it can be also caused by adverse factors for the personnel in a working zone in close proximity to the turbine.

C. Aerospace area

Another application example for the embedded Web technology is the remote aircraft maintenance, repair and overhaul (for instance, monitoring and maintenance service of aircraft under field conditions, in an emergency). The remote information access for heavy maintenance is already used in the SITA’s AeroNet applications. The next step could be direct remote testing and trouble shouting. The high-qualified personnel at the main technical center of the aircraft manufacturer can do it through SITA’s intranet and the embedded Web technology. An embedded Web server in on-board computer will provide the personnel with all necessary information and control for maintenance and testing. Data from on-board computers (at the airline's main base) can be transmitted, real-time, via AeroNet and SATCOM as it is done now by SITA for use by airline operations and maintenance organizations located throughout an airport. Trough air-communications, SITA's ACMS-AeroNet connectivity, it could be done even for an aircraft in flight.

A specific feature of many industrial embedded systems is their dynamic and real time nature.  These features must be reflected in the Web interface developments, in the embedded Web server architecture and in the communication technologies for distributed and remote systems.

## 3. Data representation

A. Static and dynamic data

There are many data units and flows in industrial embedded systems and controlled objects. Some of data are used by operators to monitor the object's state and to form an action. These data dynamically change in time. A set of monitored data must be derived, processed and represented to operator. Processes that perform data acquisition and preprocessing constitute an essential part of the embedded system software. The Web technology offers unified mechanisms not only for the measured values representation, but for the control of data capturing processes too. A set of monitored data is the main source for embedded Web server. Using main terms of Web technology - HTML tags, we may build different kinds of data into an HTML form.

B. Data classification

Data types in industrial embedded systems can be classified as following. It is not exact classification but it may show main features of Web technology applications.

Data types:
Issued by server
Sensor states, indicators. Shows finite number of states in controlled object.
Current numeric value. Shows the number that is one digital parameter.
View of module or detail. Shows real view of object block.
Set of characteristics. Shows number of digital or quality characteristics.
Dynamic data f(t). Represents dynamics behavior of state.
Issued by user
Variant. Selection from list one of allowed variants.
Numeric value. Setting value of remote system parameter.
Action. Executing one or more operation on remote site.

C. Data representation

| Data type | Web elements |
|---|---|
| Sensor states, indicators | Small pictures or smart Java applets |
| Current numeric value | Data field |
| View of module or detail | Detail pictures or complex Java applets drawing view of module in dynamics, also may be image maps. |
| Set of characteristics | Tables, bulleted lists, etc. |
| Dynamic data f(t). | Pictures as background, Java applets that takes data from a host for dynamic view. |
| Variant | List box or special applet. |
| Numeric value | Input field |
| Action | Button |

These elements can be easily built in HTML pages, linked by hyperlinks. Just these HTML files constitute a user interface coding at the client side. Dynamic data are formed by some process, which runs in the embedded system. A set of data, which operator needs for monitoring of the object (dynamic data included) may vary in time. For dynamic data it means that a set of data acquisition processes, which should run, will very in time also. In the Web technology we actually control it from the client's browser when click a button or go through a hyperlink to a page with an URL to the process. It forms a request to the embedded Web server that fires an appropriate data acquisition process. An embedded Web server and the operating system in which it runs must support these dynamic changes of running processes set.

## 4. The embedded Web server architecture

### A. Architecture requirements

The nature of industrial embedded control and monitoring systems define them as real-time systems. It should give the information to a system operator reliably and in a real time scale. Thus the time constraints require real-time operation of an embedded Web-server. An embedded Web-server should implement standard functions of HTTP protocol (such as HTTP GET method) and should be independent of a type of the in-system communication network. If we design an embedded Web-server as not a proprietary, specific for particular application, but a volume product, it should be scaleable software system, capable to satisfy a wide range of time constraints. It can be built as multiprocess system, which can run on one processor in multiprogram mode as well as on parallel multiprocessor or distributed system.

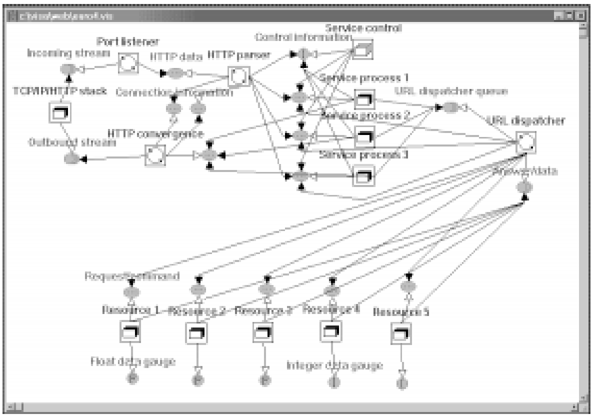



*Figure 1 VISA scheme of network of processes*

B. Architecture design and tools

The Web server tasks are parallel in their nature and it can be presented as a dynamic network of communicating processes. For development of Web server as a network of processes we use a modern CASE tool – VISA, the visual language and interactive environment for parallel and distributed programming [7]. The Web-server parallel program is presented as a network of parallel processes (see figure 1). The first module is the process, which listens of the incoming data flow for requests to the Web server. It extracts such requests and transfers them further for processing. The following module carries out disassembly of the incoming requests and distributes them over service processes. Each service process is an independent module, which carry out standard functions: connection establishment, request reception, answer request, connection closing. The URL-dispatcher is granting data for service processes. The URL-dispatcher determines by a Uniform Resource Locator (URL) [8] what resource of the Web server has been requested by the client. It establishes connection between the resource and service processes. If a dynamic data are requested, than a data acquisition process may be fired. Many embedded control systems give a parallel or distributed microcomputer platform for embedded Web server implementation.

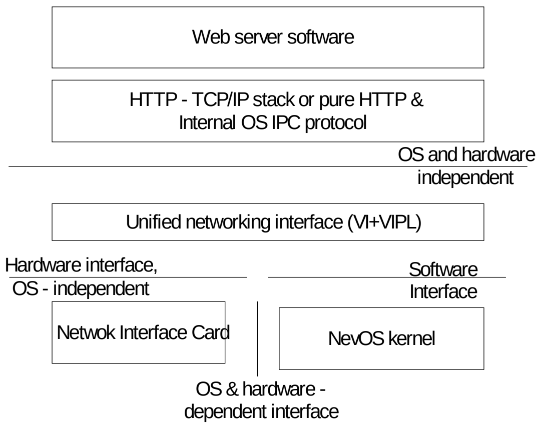



*Figure 2 The VI in distributed Web server architecture.*

C. Network architecture

The control system of the power station turbine [2] consists of a set of microprocessor modules connected by a communication network, which forms a SAN (System Area Network) for the embedded system. An aircraft on-board embedded control system has rather similar structure. In such a distributed multiprocessor system the Web server processes can run simultaneously on different units. The parallel data processing is necessary not only for acceleration to satisfy time constraints of a request, but to process multiple sources of the data that are typical for large controlled objects.

D. Used network and system technologies

A distributed real time operating system is necessary to support real time mechanisms for an embedded Web server. Also it should support the distributed parallel data processing. We use the distributed RTOS NevOS [9].

The network interface for communications in an embedded system in general and embedded Web-servers in particular should be network-independent. To follow the trend of embedded Web server as a volume product, which must run on various embedded microcomputer platforms, we need to make it independent of the embedded system communication network. Network protocols in RT systems should be protocols with the small overheads and should have small time latency. To make embedded Web-server independent of the SAN implementation we use the VI (Virtual Interface) concept, [10]. It suits RT systems by its characteristics: a low overhead, low latency time and standard program interface (so-called VIPL - virtual interface primitives library). The general scheme of VI in embedded Web server architecture is shown on figure 2.

## 5. Global Networking

The Web technology is usually associated with the Internet/Intranet, with their TCP/IP protocol stack. It allows an operator interaction with a remote system via global Internet network. When we use the Internet we must ensure not only access to the embedded system but security also. There should be a special block in the embedded Web, which implements the SSL (Secure Socket Layer) protocol for using public networks in private applications. Intranet networks are more reliable and secure.

We consider Web technologies for remote monitoring and management systems, which are not included in a main loop of control of the object. Delays in such a system can be up to several seconds or even minutes. Though it makes possible to use general-purpose network technologies, we should take into account the real-time features of a globally distributed system – embedded system with the Web server and globally distributed clients.

In case of TCP/IP protocol stack usage, which does not provide Quality of Service (QoS) procedures, the reliability and response time characteristics of a Web server access are limited. The ATM technology is more appropriate for these tasks[11]. The QoS characteristics management is implemented in the ATM technology as its standard feature. An ATM based intranet network should be a prospective backbone for networking in communications with remote systems with embedded Web. Also we should use VI architecture for seamless transition between fixed and mobile communication [10, 12].

## 6. Summary

A. Tasks for investigation and implementation

Embedded systems such as embedded Web server implementation is limited by mobile communication requirements. We must consider following issues involved in embedded software design of mobile computing. Communications and types of applications consist of several types: movable (portable) implementations, slow mobile users, fast mobile users [12]. We also should pay attention to system requirements issues, including application requirements. Requirements may be applied both to Web middleware and mobile communicated user. We should consider symmetry or asymmetry of information flows. Some combination of flows may exist. Mobile user may communicate with mobile system with embedded Web server, mobile user may try to access fixed server etc. Some issues exist in the field of specific implementation. This problems is problems of architecture and technical design.

B. Future working program

Our future work include theoretical investigation in area of embedded and mobile systems. We want to simulate our system using special tools including VISA. As one of the significant steps we consider development hardware and software prototypes of future systems. For our customers, students and for exhibition purposes we are designing demonstrator system - example of application system design using intended hardware and software tools.

C. Resume

We have discussed the main principles of the embedded Web service for industrial applications. The Web technology gives unified, cost-effective means for information representation, for system control and monitoring via standard Web interface and navigation technique. Modern parallel and distributed embedded microprocessor systems give prospective platforms for high-performance real-time embedded Web servers. Dynamic multiprocesses architecture forms scaleable embedded Web servers as volume product for a variety of applications. It is based on employment of advanced distributed real-time operating system and VI as network-independent interface. A remote access of globally distributed clients to an embedded Web server may use Intranet/Intranet networks with TCP/IP protocol stack, though based on the ATM technology networks could be more prospective for the real-time communications with embedded Web servers of industrial systems.

## References


1. The Open Microprocessor systems Initiative (OMI). // ESPRIT Programme of the EU Research & Development Framework IV programme.

2. M.S. Fragin, S.V. Gorbachev, M.B. Ignatiev, S.Y. Pomelov, Y.E. Sheinin "The Modular Multimicroprocessor System Architecture for the Power Turbine Control" // Proceedings of the Sixth Euromicro Workshop on Real- Time Systems. June15-17, 1994. Vaesteraas, Sweden. IEEE Computer Society Press, PP.124-127

3. Ignatiev M.B., Burakov V.V., Emelianov M.M.. "Internet as information technology" // "Aspects of developing and running the new types of equipment for power engineering (Issue 1)” St.-Petersburg, NIIelectromash, 1996.

4. HTTP Hypertext Transfer Protocol. (http://www.w3.org)

5. RFC-1866 T. Berners-Lee, D. Connolly, "Hypertext Markup Language - 2.0", 1995

6. David M. Howard "The Embedded Web Server" // NV, USA, 1996

7. Ignatiev, Y.E. Sheinin, D.E. Tatkov. " VISA - The Interactive Visual Language for Parallel and Distributed Programming ". // Supercomputing'96 conference, Pittsburgh, USA, 1996

8. RFC-1738 T. Berners-Lee, L. Masinter, M. McCahill, "Uniform Resource Locators (URL)", 1994

9. Aladova T.E., Nedzelskaya N.D., Sheinin Y.E. NevOS – The Scaleable Real-Time OS for Embedded and Distributed Systems. // Real Time Magazine, 1997 N 3, PP.23-28

10. Virtual Interface Architecture Specification for System Area Networks. Revision 0.9 // Intel Corporation, 1997, 86 p.

11. Emelianov M.M., Ignatiev M.B., Sheinin Y.E. Parallel Web Service Over ATM Networks of Microcomputer Platforms. // Int. Symposium on Problems of Modular Information Computer Systems and Networks (ICS-NET'97). Abstracts. Moscow, 1997. PP.17-18.

12. V.I. Nagornov, V.B. Smirnov Multiprocessor computing systems in zone broadcast networks // In "Distributed computing and VLSI systems", Leningrad, LIAP, 1988, pp.23-29.

# Keynote Session 2: "Satellite Systems for Technophony and Multimedia Services"

**Speaker**

Frédéric Patricelli

*Italy*

# SATELLITE SYSTEMS AND TELECOMMUNICATION SERVICES

Frédéric Patricelli, Scuola Superiore G. Reiss Romoli S.p.A.,
Via G. Falcone, 25 67010 Coppito (AQ) Italy

## Abstract

The last decade has been charac-terised by a fast-growing use of systems for mobile communications. Today, cellular radio mobile systems, a name which derives from the fact that the territory is partitioned into a set of areas named 'cells', are very common and it is easy to forecast that they will become even more popular in the near future. One of the main limitations of this kind of system concerns the inherent difficulty and cost of building network infrastructures for coverage of very large areas for the provision of telecommunication services. In order to solve that problem, the synergy between radiomobile and satellite technologies has led to the design of Satellite Systems which, by means of a constellation of Low Earth Orbit (LEO) satellites, are able to provide global coverage of the whole planet (Anywhere, Anyone, Anytime). The purpose of the paper is to present the main systems/projects/proposals of Satellite Systems, their characteristics and the types of services they will be able to provide.

# 1. Introduction

As is well known, the exponential evolution of technology is closely related to the growing communication facilities between human beings. The two most important 'inflection points' of this evolution coincide with the invention of the printing, by Johann Gutenberg in 1430, and with the more recent invention of computers and computer networks. As a matter of fact, by analysing the last 1500 generations of human beings that lived on our planet, it is easy to observe that only 60 were able to write and only 6 lived in an industrial civilisation. Furthermore, by focalising the attention on more recent events, only 2 generations have lived through the progress made in the automobile, electronics and aeronau-tics industries. The piston-engine, for example, was invented by Eugenio Barsanti only in 1853. Since the first radio communication experiment, made by Guglielmo Marconi in 1895, radio technology has been subject to the same exponential evolution. In a little bit more than one century, that technology passed from Hertz's first links to the recent analog radiomobile cellular systems (first generation) such as AMPS (Advanced Mobile Phone Service), NMT (Nordic Mobile Telephone) and TACS (Total Access Communication System), to the digital systems (second generation) such as D-AMPS/IS-54 (Dual-mode AMPS/Interim Standard-54), CDMA/ IS-95 (Code Division Multiple Access/Interim Standard-95), JDC/PDC (Japanese Digital Cellular/Personal Digital Cellular) and - maybe the most popular - GSM (Groupe Spécial Mobile and successively Global System for Mobile communications) conceived in 1982 and commercialised in 1992. The development ([Giordani93],[Lobley95]) of this kind of communication systems became a reality because of the innovations in various research fields, which have provided a very compact radio technology and a higher "intelligence" in the mobile terminal equipment and networks.

Furthermore, some years ago, the scientific community began to define standards for the so- called third generation cellular systems, known in Europe under the UMTS (Universal Mobile Telecommunications System) acronym. These systems will further satisfy the communication requirements of the 21st century: i.e. greater variety of services, throughput higher than 2 Mbits/s, multimedia services, quality of signal comparable to that of the fixed networks, higher security, open and flexible network architectures, international roaming and inter-system hand-over.

Meanwhile, mobile communications based on satellite technology has been subject to rapid development too: from its conception, by Arthur C. Clarke [Clarke45], who since 1945 defined the principles of communications through three geo-stationary satellites, to satellite systems based on constellations composed of dozens of LEO satellites. In order to give some idea of how many metallic objects are orbiting "over our heads" today, it is enough to think that NASA (National Aeronautics and Space Agency) traces daily: approximately 500 satellites for telecommunications, several dozen thousands of satellites for military applications (3000 Russian spy satellites of the Cosmos series were launched) and millions of metallic fragments ranging from some centimetres to some millimetres.

The present paper would be only a short survey on the main systems/projects /proposals of satellite systems that today are under development and/or study for providing telecommunication services. Attention will be focused on the type of services (messaging, telephony, multimedia) that these systems will provide in the next future. A short comparison between the different satellite systems will be also reported. Due to the limited length of the paper, the description of the systems and the technical details will be presented during the tutorial.

- Technological evolution in the telecommunications world progresses at unimaginable speed.
- In the last century, technology rapidly passed from the first Hertz's links to the recent Cellular Radio mobile Communication Systems.
- Satellite technology allowed the transcending of the limitations of modern Cellular Systems by providing global coverage of the whole planet.

## 2. 2. Historical background

For a long time [Finean96] research on communication systems based on satellite technology was confined to organisations like: ESA (European Space Agency), ‘Archimedes’ and ‘Artemis’ programmes; NASDA (NAtional Space Development Agency), ‘engineering test satellite’ programmes; and NASA, ‘advanced communications satellite technology’ programmes.

By the beginning of '80, the FCC (Federal Communications Commission) released a series of licenses for the development of RDSS (Radio Determination Satellite Service) systems to a large number of American companies. Only one of them, Geostar, developed a system which was not successful, mainly due to the limited type of services (messaging) which it was able to provide. Successively, Geostar submitted another project for the development of a new system which, in May '91, was rejected by FCC. Geostar then left the so- called "space race". The frequencies assigned to Geostar were used for Glonass, the Soviet positioning system, and for some radio astronomy experiments. In the meantime, in September 1990, the FCC requested by means of a ‘Public Notice’, comments on the second proposal of Geostar and in November 1990 accepted a project from Ellipsat which was able to provide both messaging and telephony services. Immediately after this, in December 1990, Motorola presented the proposal for *Iridium* - more ambitious than Ellipsat and already announced in June 1990. From that time many further systems were proposed by other companies, and in January 31$^{st}$ 1995 the FCC authorised the development of 3 systems: *Iridium* from Motorola, *Globalstar* from Loral Aerospace, Qualcomm et ali, *Odyssey* from TRW (Thompson-Ramo-Wooldridge Inc.), and Teleglobe. A few months ago, however, Odyssey left the "space race". The main activities concerning these systems was therefore confined to North America, although INMARSAT (INternational MARitime SATellite organization) one of the most active organisations, outside the USA, in the field of communications based on satellite technology, proposed its own project: ICO (Intermediate Circular Orbit) ex Project 21, to be developed with Hughes Electronics, NEC and others.

Due to the limited bandwidth capacity, these three systems will be able to provide only telephony services (message, telephony, fax and data), but the fast-growing demand for communications services, since these systems were conceived, requires the possibility of transmitting other types of information (image, video, etc.). Therefore other satellite systems were conceived to provide multimedia services. The current main proposals are: *TELEDESIC* (TELEcommunications geo-DESIC network) by Microsoft and McCaw Cellular Communications, *Celestri* by Motorola, *Euroskyway* by Alenia Spazio, *Astrolink* by Lockheed Martin Telecommunications, etc.

- Three Satellite Systems for Telephony Services are today in the implementation phase: Iridium, Globalstar and ICO.
- Other Satellite Systems for Multimedia Services are already in the development phase: Teledesic, Celestri, Euroskyway, Astrolink, etc.

## 3. 3. LEOs vs GEOs

Until recently, the majority of satellites for telecommunications were positioned on a Geo-synchronous Earth Orbit (GEO). A satellite on such an orbit moves synchronously and therefore its relative position to the earth do not change: once pointed, the antennas do not need further adjustments. The main roblems with this kind of orbit relate to: (i) frequency allocation due to the growing number of satellites; (ii) an altitude (36.000 km) which requires powerful and therefore expensive radio equipment and (iii) transmission delay (250 ms at minimum), unsuitable for classical telephony services. The new service demands expressed by the market and the need to decrease the cost and dimension of the equipment, especially in the field of services for mobile communication and/or broadband, led to consideration of the use of lower orbits. In this new scenario, each Low Earth Orbit (LEO) or Medium Earth Orbit (MEO) satellite changes its relative position and is visible from a point on the earth surface for a limited time period ranging from 5 to 15 minutes. In order to offer a global coverage and continuity of service, many satellites (a constellation) must be available on the appropriate orbit to ensure that at least one satellite is visible for transmission at all times. Sophisticated hand-over procedures were developed from cellular radio mobile technology in order to switch a channel from a setting satellite to a rising one. Otherwise, *store and forward* solutions must be taken into consideration (i.e. a message is stored when a satellite passes over the source and released during the visibility time period of the destination). However, while this technique can be used for services that do not have real-time requirements such as E-mail it cannot be used, of course, for telephony services.

## 4. 4. Satellite Services and Systems

The satellite constellation based systems, available in the near future can be classified in two categories: (i) *big-LEOs*, characterised by a con-stellation of large satellites providing a global (or almost global) coverage of the planet and suitable for real-time telecommunication services, and (ii) *little-LEOs*, characterised by cheaper and smaller satellites providing store-and-forward messaging services. In this paper attention will also be focused on three classes of services that satellite systems are going to supply: (i) *Messaging* services; (ii) *Telephony* services (narrow band) and (iii) *Multimedia* services (broadband).

### *A. Messaging Services*

This first class of services allows the transfer and delivery of data messages between equipment located into the coverage area and from/to sites connected to conventional fixed data networks. These systems are usually characterised by low cost and low power consumption equipment able to communicate at low throughput (some hundreds of bits per second) with one or more terrestrial station that provide gateway functions to the conventional terrestrial networks. This equipment (terminal set) is mainly used for two kinds of services: (i) mobile communication applications for vehicles (cars, trains, air planes, boats, etc.) in order to provide mobile service assistance and (ii) transmission services such as telemetry, alarms, etc. generated by sensors that are typically located in rural areas lacking terrestrial infrastructures.

The main systems/projects/proposals for messaging services are: Orbcomm, Final Analysis, Leo One, E-Sat, LeqO, Euteltracs, GE Starsys, etc. In particular, Orbcomm is a system based on a constellation of 26 LEO satellites (36 for future expansion) designed to provide bi-directional messaging (up to 3000 chars) services (E-mail, Short Message Service, Data) between Subscriber Communicators (SC) and fixed equipment. The addressing is compatible to ITU-T, X.400 and Internet E-mail standards. The first 2 satellites of the constellation were launched in April 1995. The constellation will be completed by June 1998. The transfer of messages (current throughput of 4800 bits/s and 9600 in the next future) from SC to fixed equipment is issued in the following way: (i) the originated message is transferred from the SC to the GES (Gateway Earth Station) through the constellation, (ii) forwarded to the GCC (Gateway Control Centre) and then (iii) routed to the final destination through terrestrial networks (dial-up, Internet or dedicated lines) and vice versa for fixed equipment originated messages. Typical Orbcomm applications are: (i) localisation and monitoring of vehicles (boats, trucks, containers, etc.), (ii) status control of ground located equipment and sensors, (iii) remote control of actuators on vehicles, (iv) environment monitoring, (v) emergency/security, and (vi) communications and personal localisation.

### *B. Telephony Services*

This class of narrow band services relates to the conventional voice, data, fax and Short Message Service (SMS). The coverage characteristics of satellite systems allow them to be particularly suitable for mobile telephony applications. As mentioned above (see section 3), the recent introduction of LEO systems also allowed the minimisation of transmission delay and the mobile user terminal dimension (palmtop or hand set). As a matter of fact, these systems are today considered as complementary to - or even competitor of - terrestrial cellular radio mobile systems. They form an important part of the Universal Mobile Telecommunication Systems (UMTS). Characteristics comparisons of the main systems that will provide telephony services are given in Figure 1.

*Table 1. Comparison between the Iridium, Globalstar, Odyssey and ICO Satellite Systems for Telephony Services.*

| Parameter \ System | Iridium | Globalstar | Odyssey | ICO |
|---|---|---|---|---|
| Investors | Motorola et ali. | Loral et ali. | TRW et ali. | Inmarsat et ali. |
| System Cost (bn US$) | 3.7 | 2.2 | 1.8 | 2.6 |
| Hand set Cost (US$) | 2000 | 750 | 300 | 300 |
| Service Cost / Min. (US$) | 3.5 | 0.45 (*) | 0.65 (*) | 1-2 |
| Services | Voice, Message, Data, Fax | Voice, Message, Data, Fax | Voice, Message, Data, Fax | Voice, Message, Data, Fax |
| Voice throughput (kbit/s) | 4.8 | 4.8 | 4.8 | 4.8 |
| Data throughput (kbit/s) | 2.4 - 4.8 | 7.2 | 9.6 | 2.4 |
| Irradiation Power (W) | 3.7 | 2 | 0.5 - 5.0 | 0.25 - 5.0 |
| First launch | 05 /05/ ‘97 | 14 /02/ ‘98 | 1998 | 1998 |
| Market launch | 1998 | 1999 | 2000 | 2001 |
| N° Satellites | 66 | 48 | 12 | 10 |
| N° Orbital planes | 6 | 8 | 3 | 2 |
| N° Spare Satellites/orbital plane | 1 | 1 | 1 | 1 |
| N° Total satellites | 72 | 56 | 15 | 12 |
| Quota (km) | 780 | 1414 | 10354 | 10355 |
| N° Satellites to cover an area | 1 – 2 | 2 | 2 – 3 | 2- 3 |
| N° Ground Stations | 16 | 140 | 7 | 12 |
| Access Technique | TDMA | CDMA | CDMA | TDMA |
| Modulation Technique | QPSK | QPSK | BPSK | QPSK |
| N° Beam per satellite | 48 | 16 | 37 | 163 |
| Freq. Reuse (beams) | 7 | 1 | 3 | 4 |
| Available channel / beam | 23 | 175 | 62 | 28 |
| Uplink (GHz) | 1.62-1.63 | 1.61-1.63 | 1.61-1.63 | 1.98-2.01 |
| Downlink (GHz) | 1.62-1.63 | 2.48-2.50 | 2.48-2.50 | 2.17-2.20 |
| Inter-SV Link (GHz) | 23.2-23.4 | / | / | / |

(*) + Terrestrial networks transit costs.

A particular note must be given on the Iridium system: at the time this paper is being written, Motorola has completed the entire constellation and therefore achieved the world record of launches (66 LEOs satellites in a little bit more than one year). Furthermore, Iridium is the unique system that will route phone calls without the use of terrestrial networks (cut-through) i.e. the satel-lites are linked by using cross link connections and have routing capabilities. The first Iridium test call was successfully set-up in December 1997. In addition, Iridium represents the only example of a large-scale alliance between dozens of International companies that worked without "divorce" for more than seven years.

### *C. Multimedia Services*

This class relates to the broadband services ranging from E-mail/VSAT (16 Kbps) to PBX LAN (100 Mbps) passing through Desktop video conferencing (144 Kbps), High quality video /Internet access (384 Kbps), Document imaging /data access and distribution (1.544 Mbps), broadcast-quality video (6 Mbps), Ethernet (10 Mbps), HDTV (25 Mbps) and bulk transport (50 Mbps). The systems that will provide these services will be compatible with IP, MPEG2, ATM and ISDN standards.

Comparisons of the characteristics of the main projects/proposals of satellite systems providing multimedia services are given in Table 2. A special note must be added on Motorola's Celestri, the unique system composed of the integration of a LEO constellation for real time and a GEO constellation for broadcast applications. That system will be able to provide, on demand, bandwidth ranging from 64 Kbps to 155 Mbps.

*Table 2. Comparison between the Teledesic, Celestri, Euroskyway and Astrolink Satellite Systems for Multimedia Services.*

| Parameter \ System | Teledesic | Celestri | Euroskyway | Astrolink |
|---|---|---|---|---|
| Investors | Microsoft et ali. | Motorola et ali. | Alenia Spazio et ali. | Lockeed Martin et ali. |
| System Cost (bn US$) | 9.0 | 9.0 | Unknown | 4.0 |
| Services | Broadband Multimedia | Broadband Multimedia | Broadband Multimedia | Broadband Multimedia |
| First launch | 2001 | 2002 | 2000 | 2000 |
| Market launch | 2002 | 2003 | 2001 | 2001 |
| N° Satellites | 288 | 63 LEO + 9 GEO | 5 | 9 |
| N° Orbital planes | 12 | 7 | / | / |
| Quota (km) | 1350 | 1400 and 35786 | 35786 | 35786 |
| Note | Coverage of 95% of the earth and 100% of the population | Three types of CPE: Residential 750 US$ Corporate 7500 US$ Small Buss. 14000 US$ | Three types of CPE: Portable 144 kbps Standard 512 kbps High capacity 2Mbps | Three types of CPE: Small 384 kbps Medium 2 Mbps Corporate 10 Mbps |

## References


*[Clarke45]* A.C. Clarke: “The Future of World Communications”, Wireless World, October 1945, pp. 305-308.

*[Balston93]* D.M. Balston, R.C.V. Macario: "Cellular Radio Systems", Artech House, Boston, 1993..

*[Giordani93]* M. Giordani, F. Grimaldi: "Reti Radiomobili Cellulari: il sistema a 900 Mhz analogico standard ETACS", Technical Journal of Telecom Italia, N° 3, April 1993, pp 23-36.

*[Lobley95]* N.C. Lobley: "Intelligent Mobile Networks", BT Technology Journal, Vol. 13, N° 2, April 1995, pp 21-29.

*[Mouly95]* M. Mouly, M.B. Pautet: "The GSM System for Mobile Communications", 1995..

*[Pandya95]* R. Pandya: “Emerging Mobile and Personal Communication Systems”, IEEE Communications magazine, Vol. 33, N° 6, June 1995, pp 44-51.

*[Finean96]* R.J. Finean: “Mobile Satellite Communications Proposals”, BT Technology Journal, Vol. 14, N° 3, July 1996, pp 74-80.

# Session 3:
# "Enabling technologies"

# Current (Wireless) Inter-technology Roaming Efforts

Roman Pichna, Jari Vallström, Mika Ylianttila, Jaakko Talvitie
University of Oulu, P.O. Box 444, 90571 Oulu, Finland

## Abstract

This paper presents an overview of the present-day efforts that concentrate on inter-technology (vertical) roaming. The emphasis is on seamless roaming over wireless media. The efforts are documented with a representative selection of research projects. The projects are grouped according to the protocol stack level where the core part of the mobility is implemented. As this overview shows many major projects use Mobile-IP (M-IP) as a generic vehicle for the wireless inter-technology roaming.

## 5. Introduction

The current development of mobile wireless communications has led to a plethora of incompatible solutions, many of which have already been implemented and deployed. The main difference making features among them are the used wireless interface, quality and choice of the provided services, geographical availability and the cost. It may be clear by now that no single wireless communication system can cover all the possible needs of users in all situations. To satisfy the users' need for ubiquitous communication the users will have to roam between different wireless systems and thus different technologies. It will be of great benefit if the switching or inter-technology handover is provided in a seemingly transparent fashion. In other words, the user should not notice the handover process (if he/she chooses to), except maybe by the perceived change in the quality of service (and cost). To illustrate the problem, one possible scenario for this case is depicted in Figure 3. A mobile user is moving away from the coverage of a wide-band, small coverage underlay network (wireless LAN) with a connection in progress and has to hand over to a wide coverage, low bandwidth overlay network (UMTS, GPRS, CDPD, etc.). A particular case of wireless LAN (WLAN) – UMTS seamless roaming is being explored within the WiLU (Wireless LANs for UMTS) project 0 run at the Centre for Wireless Communications, University of Oulu, Finland in cooperation with the Center for Wireless Information Network Studies, Worcester Polytechnic Institute, MA, USA.

Based on the prevalence of the IP protocol in the current networks we may assume that the user terminal uses IP protocol stack. The current IP protocol version 4 does not support any hand-over features since it was designed for static networks with permanent point of attachment. Neither the IP version 6 fully facilitates seamless mobility for now.

There are four possible levels where the inter-technology handover problem can be solved: above TCP/IP level, transport level (TCP), network level (IP), underlying protocol level.  All of these have attracted a significant research effort all around the world, confirming that up to the present date the inter-technology handover among different technologies has not been solved satisfactorily.  This paper will provide a representative overview of major projects.  An interested reader may want to browse through compiled lists of URLs 0, 0 and 0.

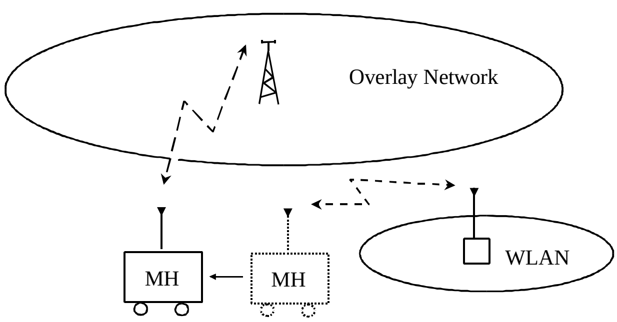



*Figure 3: Handover from a WLAN to an overlay network.*

## 2. Above TCP/IP-level Solutions

In general, a special inter-technology roaming protocol layer can be inserted between the application and the transport protocol layer. Representatives of this approach are the project On The Move, and Mobile TCP/IP which will be described below. There is another category of approaches that offers modifications of existing protocols sitting above the TCP/IP stack. One of them is the work on the X-interface mobility. An example is described in 0, 0 and 0. The basic idea is that the X interface is following the user as he/she moves. This can be also used, in principle, in inter-technology roaming as the user changes the point of attachment to the network.

### *A. On The Move (ACTS)*

On The Move 0, 0 is an ACTS project in the 4th framework. In UMTS, local and wide area networks are integrated in one system. Intention of the On The Move project is to make different wireless bearers to appear to the applications as a seamless and homogeneous communication medium. This is achieved by building a Mobile Application Support Environment (MASE) to support both mobile-aware and legacy (i.e., non-mobile-aware) applications (see Figure 4). Mobile Application Programming Interface (M-API) allows mobile-aware multimedia applications to signal their special quality of service requirements to the computing and communication subsystem below. The UMTS Adaptation Layer (UAL) adapts MASE to various networks. MASE implements protocols for various tasks between these two layers.

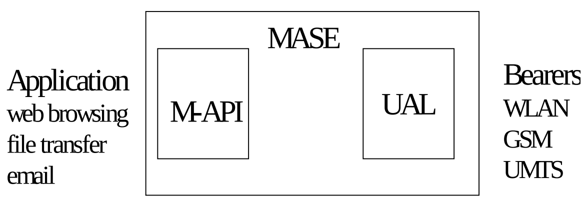



*Figure 4: Components of MASE software.*

In many cases it will not be possible for the MASE to get all the necessary information and resources from the mobile terminal itself. For this reason, a distributed approach will be taken using Mobility Gateways. Mobility Gateways are attached to the wired network and are aware of the current status of mobile terminals and the underlying communications networks. Based on user and application profiles, the MASE is then able to gather information and resources from the mobile. Project aims also to provide servers with more flexibility to serve mobile users where the server is aware of the communication conditions and can automatically select the most appropriate way to send the user's desired information over the network.

### *B. Mobile TCP/IP*

Mobile TCP/IP 0 was developed at the National University of Australia. A mobile mapping is introduced which maps TCP associations to underlying TCP connections. The mobile mapping is implemented in the socket layer. Mobile TCP socket is implemented over TCP/IP stack and it adopts the same API of standard sockets. No special functionality is required from TCP/IP or lower layers to support mobility. Network is not required to support any new routing mechanisms.

## 3. Transport-level Solutions

At the transport level, the mobility can be provided if a mobility gateway intercepts the IP traffic, terminates the TCP connection and re-establishes a new connection to the mobile host. In this way, the remote application is always communicating with the mobility gateway and its TCP connection is maintained even if the mobile changes the point of attachment. Good examples of this approach are Indirect TCP 0 and MSOCKS 0, where MSOCKS is the most recent.

MSOCKS is run at Carnegie Mellon University in co-operation with IBM. The MSOCKS protocol is built on top of SOCKS protocol for firewall traversal 0. The basic idea is TCP-level connection splicing at a proxy. Due to the peculiarities of some TCP applications, such as ftp, a control channel is set up for communication with the TCP protocol on the mobile host and the proxy.

MSOCKS takes care of a proper in-sequence delivery of packets during the handover procedure. Another benefit of MSOCKS is that it inherently supports firewall traversal.

## 4. Network-level Solutions

There were several extensions and modifications proposed to accommodate mobility in IP networks at the network level. An overview with good references is given in 0. The most widespread and known approach is the Mobile IP (M-IP) extension to the IPv4 as described in 0. For an overview of the available implementations see 0. The principle of M-IP is to intercept IP packets destined for a Mobile Host at its Home Agent, encapsulate it in another IP packet and tunnel it to the new temporary address at the new point of attachment. IP-in-IP tunneling increases the amount of the overhead by basically doubling the headers. If this is a problem, e.g., due to the limited bandwidth of the overlay network, minimal encapsulation, adding just 8-12 octets of the overhead may be used. This problem was effectively addressed in the design of IPv6. One shortcoming of the M-IP approach is triangular routing, which means that the outgoing packets from the mobile are not tunneled to the Home Agent but sent out using the user's home address. This may pose some problems in foreign networks that do not allow a flow-through traffic. A M-IP extension to solve this problem is Reverse Tunneling 0, where all traffic to and from the Mobile Host is tunneled from and to the Home Agent.

Another shortcoming of M-IP is that some IP packets can be lost during the handover. The common assumption is that it is a transient problem and it is taken care of by the higher layers, such as TCP, or by the application in the case of UDP. As representatives of their class we have chosen the projects Monarch, MosquitoNet and Daedalus/Barwan.

### *A. Monarch*

Mobile Networking Architecture (Monarch) 0, 0 project group at Carnegie Mellon University has done a remarkable work in the area of inter-technology (vertical) roaming. They have implemented Mobile IP protocol for IPv4 and IPv6 on 4.4BSD-based UNIX platform. The implementation contains all features of the basic IETF Mobile IP protocol. The source code is available over the Internet and the authors keep maintaining it by distributing patches and software updates. These protocols support transport-level transparent roaming of mobile hosts throughout Internet, including dynamic switching between different bearers at any time.

### B. MosquitoNet

The MosquitoNet 0 project is run at Stanford University and is working towards providing seemingly continuous network connectivity for mobile computers on the Internet. The test-bed combines wired and wireless network access with mobile IP to make it possible to switch seamlessly between different types of bearers. The project has implemented Mobile IP for Linux platform and made it available over the Internet 0. The implemented Mobile IP protocol is a subset of that described by the Mobile IP working group. The implementation, however, supports complete mobility by using co-located foreign agent. The implementation also supports reverse tunneling.

### C. Daedalus/Barwan

The project Daedalus 0,0,0,0 is effectively a successor of the project Barwan (Bay Area Research Wireless Access Network) at the University of California at Berkeley. The goal of the project is inter-technology roaming for which the term overlay internetworking is used. The project is in some extent similar to the On The Move project. The project has taken special care in handling real-time applications handovers by using IP multicast for IP-in-IP tunneling and intelligent buffering at the near-by base stations to reduce the hand-over latency.

### D. Other M-IP Implementations

Although M-IP protocol by itself does not solve the whole problem of the inter-technology roaming it provides a vehicle for it. Besides the mentioned Monarch and MosquitoNet implementations there are many other M-IP implementations currently available 0. The major ones are presented below.

## 1) Royal Institute of Technology, Stockholm, Sweden

There are two M-IP implementations available from Royal Institute of Technology, Sweden. The first 0 was written by Anders Klemets and is based on draft version 11 of IP Mobility Support 0. It uses the minimal encapsulation protocol and MD5 authentication. It also provides support for using authentication algorithms other than MD5. Other features are Mobile-Foreign and Foreign-Home authentication and nonce based ID's. There is a provision for signaling between the link layer and the Mobile-IP code. This is useful when running Mobile-IP over a connection-oriented link layer protocol. The parts of the user level code that interface to the operating system have been separated into separate files. Memory allocation, timers, network interfaces and network I/O have been abstracted. Separate files are used to implement a system independent interface to these operating system functions.

Fredrik Tarberg and Fredrik Broman wrote the second Mobile IP implementation 0. It is based on the implementation by Anders Klemets (see above), draft version 14 of IP Mobility 0 and IP-in-IP Encapsulation draft version 1 0. This implementation develops and implements a Management Information Base for the Mobile IP Protocol. It also ports a Mobile IP Implementation for SunOS to MachOS and Solaris.

## 2) National University of Singapore

Mobile IPv4 and IPv6 have been implemented at the National University of Singapore for Linux and a Mobile Host for Windows95 (WinMIP) 0. As WinMIP uses Windows95's native TCP/IP stack, no third-party TCP/IP stack is required. It has also been tested to work successfully with NUS's Linux Home Agent and Foreign Agent implementations. One limitation of WinMIP is that it is able to provide portability (relocation) only and not mobility, this is attributed by Microsoft's rigid routing table implementation.

### *3) FTP Software and Telxon Corporation*

This mobile IP software 0 combines FTP Software's DOS and Windows network software, Aironet's wireless LAN access points and Telxon's portable and pen-based computers to enable the TCP/IP networking protocol to better meet the needs of mobile users. It allows users to roam across multiple segments of TCP/IP enterprise networks, without disrupting wireless network connections, and access applications and information, send and receive electronic mail, and update and query databases.

### *4) SUNY Binghampton*

The M-IP implemented for Linux at State University of New York (SUNY) 0 has the following features.

- Complies with Revision 16 of the IETF's proposed Mobile-IP standard.
- Uses IP-in-IP encapsulation.
- Supports co-located Foreign Agent.
- Supports MD5 authentication in prefix-suffix mode.
- Seamless reattachment to the Internet using PPP without disturbing any existing TCP connections.
- Router discovery ICMP messages are used for agent advertisements and solicitations.

SUNY Mobile IP implementation is quite similar to MosquitoNet Mobile IP. They are both implemented on Linux platform, and they are able to operate without Foreign Agent (FA) in the foreign network. However, this in not the default case in SUNY’s implementation (unlike in MosquitoNet). Home Agent and Mobile Host (MH) send out gratuitous ARPs when appropriate. The MH continuously monitors the IP address on the network interface through which its default route passes. If the IP address on this interface changes to something other than the MH’s permanent IP address and no FA can be ‘heard’ (either there is no FA in the foreign network or FA is unreachable for some reason), MH moves into a self-decapsulation mode.  This change may occur, for example, when a user replaces the PCMCIA network card with a modem card and initiates PPP. IP address is assigned to the 'dummy' interface and proxy-ARP is used to pick the necessary routing commands. This way connections initiated by MH always use its permanent IP address as the source.

## 5. Below-IP Solutions

The solution to inter-technology handover can be found also in layers below IP. This is in particular for networks that have their own inter-technology roaming protocols and carry the users' IP traffic as a payload. An example of this approach is the eNetwork (originally called Advanced radio Communications on Tour: ARTour) 0 aims at improving the performance experienced by data user in wireless media. An eNetwork layer is placed in between the network and the lower layers (see Figure 5). It provides address resolution and protocol conversion. Address resolution tables are maintained in the eNetwork Gateway. Within the eNetwork layer, there are several sub-layers of software architecture. The IP Interface layer deals with address resolution/mapping and filtering of unwanted packets to reduce traffic on low bandwidth segments. The Data Management layer focuses on data reduction and encryption. A link layer deals with a link protocol based on the IETF point-to-point protocol (PPP). The connection management layer suspends and reinstates the connection over a circuit switched network to save connection charges. A network interface layer deals with the interfacing to various dial-up and radio networks.

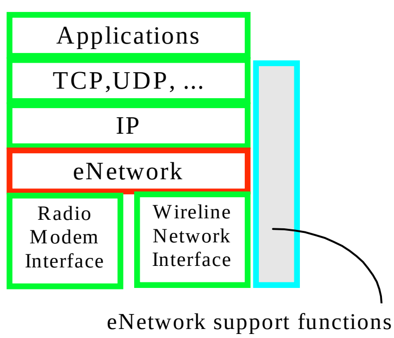



*Figure 5: eNetwork protocol and service layering.*

## 6. Conclusions

Representative projects dealing with seamless inter-technology handover were presented. Inter-technology handover cannot be handled within a proprietary protocol of one carrier (network technology). It has to be handled either in the existing layers above or a new layer has to be added solely for the purpose of handling the inter-technology handover. In either case, there is a need for modification of the existing protocols at least between the mobile host and a network entity that handles the mobility. There are several layers where the handover can be dealt with. The most common seamless inter-technology handover platform is Mobile IP (M-IP) and its modifications due to its generic nature and availability. It has been accepted as a platform for many major projects.

## 7. Acknowledgements

We would like to express our gratitude to our WiLU project partners at Worcester Polytechnic Institute, especially to Prof. Kaveh Pahlavan.

## References

1. http://www.cwc.oulu.fi/Research/Projects/Wilu/

2. http://www.cosc.canterbury.ac.nz/~hpark/MClinks.html

3. http://www.mcl.cs.columbia.edu/wwwlinks.html

4. http://www.cs.umd.edu/projecs/mcml/mcothers.html

5. Tristan Richarson, Frazer Benett, Gledford Mapp and Andy Hopper, “Teleporting in and X Window System Environment,” IEEE Personal Communications, Vol.1, No.3, pp. 6-12, 3rd qtr, 1994.

6. Andy Ward, Alan Jones, Andy Hopper, “A New Location Technique for the Active Office,” IEEE Personal Communications, pp. 42-47, Vol 4, No. 5, October 1997.

7. Ethan Solomita, James Kempf and Dan Duchamp, “Xmove: A Pseudoserver for X Window Movement,” The X Resource, No. 11, pp. 143-170, July 1994.

8. http://www.sics.se/~onthemove/

9. Meggers et al., 1996, “Roaming Between GSM and Wireless LAN”, ACTS Mobile Telecommunications Summit, Granada, Spain, vol. 2, Nov. 27- 29, pp. 828 - 834.

10. http://cs.anu.edu.au/techreports/1997/TR-CS-97-19.ps.gz

11. A. Bakre and B.R. Badrinath, “Implementation and Performance Evaluation of Indirect TCP,” IEEE Trans. Comp., Vol. 46, No.3, March 1997.

12. David A. Maltz, Pravin Bhagwat: MSOCKS: An Architecture for Transport Layer Mobility, to appear in IEEE INFOCOM 98.

13. M. Leech, David Koblas, et al. SOCKS protocol version 5. Internet Request for Comments RFC 1928, April 1996.

14. Stefan G. Hild and Peter Robinson, “Mobilizing Applications,” IEEE Personal Communications, Vol. 4, No.5, October 1997, pp. 26-34.

15. C. Perkins, Ed., “IP Mobility Support version 2,” IETF, draft-ietf-mobileip-v2-00.txt

16. http://www.neda.com/mobileIpSurvey/html/mobileIP_40.html

17. G. Montenegro, Ed., “IP Mobility Support version 2,” IETFdraft-ietf-mobileip-tunnel-reverse-04.txt

18. David B. Johnson and David A. Maltz, “Protocols for Adaptive Wireless and Mobile Networking,” IEEE Personal Communications, pp.34-42, February 1996.

19. http://www.monarch.cs.cmu.edu

20. Mary G. Baker, Xinhua Zhao, Stuart Cheshire, Jonathan Stone, “Supporting Mobility in MosquitoNet,” Proc. of 1996 USENIX Technical Conference, January 1996.

21. http://mosquitonet.Stanford.EDU/mosquitonet.html

22. Mark Stemm, Randy H. Katz, “Vertical Handoffs in Wireless Overlay Networks,” ACM Mobile Networking (MONET), Special Issue on Mobile Computing in the Internet, Fall 1997.

23. Srinivasan Seshan, Hari Balakrishnan, Randy H. Katz, “Handoffs in Cellular Wireless Networks: The Daedalus Implementation and Experience,” Kluwer International Journal on Wireless Communication Systems, 1996.

24.http://www.cs.berkeley.edu/~randy/Daedalus/BARWAN/BARWAN_index.html

25. http://daedalus.cs.berkeley.edu/

26. ftp://it.kth.se/pub/klemets/

27. draft-ietf-mobileip-protocol-11.txt

28. http://www.it.kth.se/~d91-fta/exjobb/exjobb.html

29. draft-ietf-mobileip-protocol-14.txt

30. draft-ietf-mobileip-ip4inip4-01.txt

31. http://mip.ee.nus.sg/

32. http://www.telxon.com/tch1.htm

33. http://anchor.cs.binghamton.edu/~mobileip/MIPv100/Doc/mip-doc.v100.ps

34. http://www.software.ibm.com/enetwork/

# Conformance Testing of DECT Protocols

K. Tarnay, Cs. V. Rotter,

MTA-SZTAKI,
P.O. Box 63, 1518 Budapest, Hungary

## Abstract

The main cordless communication system, the DECT (Digital Enhanced Cordless Telecommunication) is shortly introduced. The DECT protocol architecture is explained emphasizing the operation of the Data Link Control (DLC) protocol. An overview on conformance testing summarizes the basic steps of testing and as an example some DLC tests are discussed.

## 1. Introduction

Mobile and personal communications are the necessary building blocks of the telecommunications networks in the near future. There are two big mobile branches, the cordless and the cellular telephony, but the difference is not really sharp. Our paper deals with cordless telephony, with the wireless counterpart of the standard telephone.

After some analog cordless stations (CT1, CT1+) the digital cordless solutions (CT2, CT2+ and DECT) are now more attractive. Among the digital narrow-band mobile systems the DECT is a low power short-range telephone and data system, for local access. DECT closely follows the Open Systems Interconnection Reference Model and the DECT protocol stack covers the three lower layers. Every protocol has its own test specification. Our goal is to explain the place of the different protocols in the DECT system and the testing aspects. The paper after this introduction consists of 4 sections. First the DECT system is introduced, then the DECT protocols are presented shortly.

Afterwards an overview of conformance testing is given. Finally the test cases of the DLC protocol are discussed.

DECT system

The DECT Reference Model defines the relations and reference points within the DECT Network and with attached networks. The Reference Model describes these networks in both a logical and a physical domain.

Each logical and physical groupings in the reference model are defined in the Fig II.1. Reference points D3 always correspond to a physical boundary. Reference points D2 and D4 can never align to physical boundaries, they have been defined to clearly indicate the logical boundaries of the DECT Network. Reference point D1 may correspond to a physical boundary.

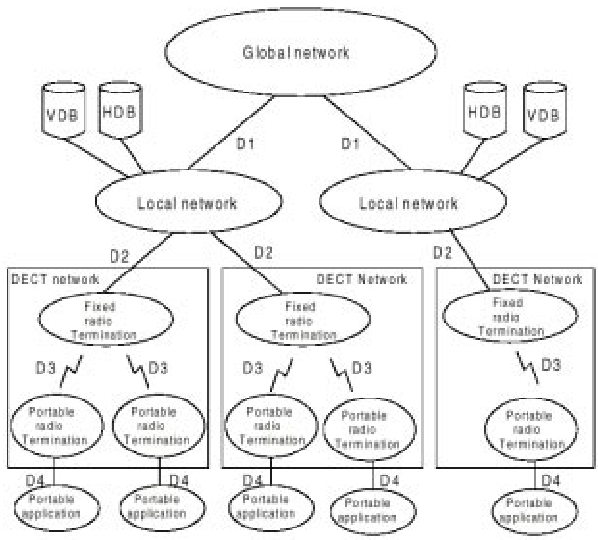



*Fig II.1: DECT Reference Model.*

The Reference Model describes the whole network architecture, where the DECT Network exists as a tree structure. This is to show that each logically higher grouping can communicate with a multiple of logically lower groupings, and the groupings on the same logical level cannot communicate directly, but only via a logically higher grouping.

In the reference configuration this multiple relation is not shown. For simplicity reasons only one branch of the reference model tree is described. In each reference configuration an example of a possible implementation is included.

*Logical Groupings*

*Global Network*

The Global Network (GNW) is a logical grouping that supports a long distance telecommunication service as well as address translation, routing and relaying between connected Local Networks. It has typically national or international extent.

The main Global Networks are e.g.: PSTN (Public Switched Telephone Network) and ISDN (Integrated Service Digital Network).

*Local Network*

The Local Network (LNW) supports a local telecommunication service. Typically, such a network is local in extent.

The Local Network is responsible for the translation of Global Network identities to DECT-specific identities.

The main Local Networks are e.g. analogue and digital PBXs (Private Branch Exchange) and LANs.

*Databases*

All network mobility functional entities must be external to DECT, either in the Local Network or in the Global Network. Two logical groupings that are needed in inter-DECT Network mobility are: the Home Data Base (HDB) and the Visitors Data Base (VDB).

The Home Data Base is a logical grouping in charge of the management of subscribers. Two kinds of information is stored there:

subscription information for charging and billing purposes and other subscriber parameters, like service profiles etc. and

location information, which enables the routing of calls from the home Local Network to the visited Local Network

The relation between identities of the DECT user or PP (Portable Part) is also stored in the Home Data Base.

The Visitors Data Base manages visiting subscribers or PPs.

*DECT Network*

The DECT Network interconnects the Local Network and the Portable Application. It can be considered as a cordless intervention between these attached networks. The DECT Network contains (in the logical domain) one or more Fixed Radio Termination (FT) and their associated Portable Radio Termination (PT).

*Fixed Radio Termination*

The Fixed Radio Termination (FT) is a logical grouping that contains all functions and procedures on the fixed side of the DECT air interface, which are in the CI specification.

*Portable Radio Termination*

The Portable Application (PA) is a logical grouping that contains everything beyond the DECT Network boundary on the portable side.

*DECT Interworking*

The connection of the DECT Network and the Local Network on the fixed side and the Portable Application on the portable side will in general require some kind of interworking. The CI (Common Interface) specification specifies messages and procedures, which support this interworking. It will not specify the Interworking Units that include these functions, or even the location of such entities. Both the units and their location will be matters for implementations and specific system requirements.

*Physical Groupings*

*DECT Fixed Part*

DECT is physically divided into two parts; a DECT Fixed Part (FP) and a DECT Portable Part (PP).

The only logical grouping in the DECT FP, relevant to the CI specification, is the Fixed Radio Termination, as described in chapter II.5. The DECT FP contains two types of physical groupings: Radio Fixed Part and Radio End Point. A Fixed Part may include more then one Radio Fixed Part, and each Radio Fixed Part includes all the Radio End Points connected to one single antennae system.

*DECT Portable Part*

The DECT PP is a physical grouping that contains DECT logical groupings (Portable Radio Termination) and non-DECT groupings (Portable Applications), i.e. all elements between the DECT air interface and the user.

The DECT Portable Part can be realized as one single physical entity, able to offer a teleservice to the user: a Portable Handset (PHS).

Dect protocols

The DECT protocol architecture is similar to the OSI lowest three layer structure. The lowest layer is specifying radio parameters such as frequency, timing, bit and slot synchronization and transmitter and receiver performance. The Medium Access Control (MAC) layer provides a broadcast message control service, a connectionless and connection-oriented message control service by selecting physical channels and allocating logical channels. The Data Link Control protocol is concerned with the provision of reliable data transfer. The main signaling layer of the protocol stack is the Network (NWK) Layer containing the functions for call control, mobility management, connectionless and connection-oriented message service and supplementary services. The management of all lower layers is built-in into the Lower Layer Management Entity (LLME). The communication either with higher layer applications or with the fixed network is solved by Inter-working Units (IWU). We pick up the DLC protocol for our further analysis. The DLC layer contains two independent planes of protocol:

the C-plane and

the U-plane

The C-plane is the control plane of the DECT protocol stacks, which shall contain all of the internal DECT protocol control with some external user information.

The U-plane is the user plane of the DECT protocol stack. This plane shall contain most of the end-to-end, external user information and user control.

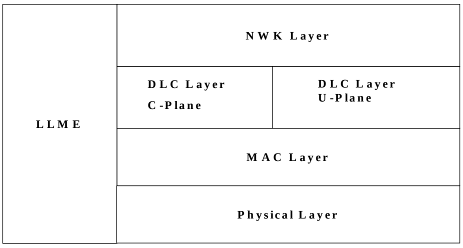



*Fig III.1 DECT protocol stack.*

In the following section of this chapter, we will introduce the DECT mobility function such as the handover procedure because this is one of the most interesting things that should be tested.

The handover is the process of switching a call in progress from one physical channel to another physical channel. These processes can be internal (internal handover) or external (external handover). There are two physical forms of handover, intra-cell handover and inter-cell handover. Intra-cell handover is always internal, inter-cell handover can be internal or external. The connection handover is the internal handover process provided by the DLC layer, whereby one set of DLC entities (C-plane and U-plane) can reroute data from one MAC connection to a second new MAC connection, while maintaining the service provided to the network layer. On the testing chapter we will explain a test case for connection handover.

The generic procedures to can execute a handover is:

establishment of a new MAC connection by the PT

normal release of the old MAC connection by the PT

Procedures 1 and 2 may be managed serially or in parallel. In the case of serial operation, where the old connection is released before the new MAC connection is established, there will usually be a short interruption to the offered service. In the case of fully parallel operation, where the new MAC connections fully established before the old connections released, the handover may give no interruption to the offered service. This results the so-called “seamless” connection handover.

An overview of conformance testing

The national and international standard organizations and the bigger computer firms are elaborating their network systems and protocol standards.

Protocol testing means the control of the protocol that is already in operation. It has two types:

Conformance testing: this tests whether the characteristics of the implemented protocol conform to those given in protocol specification.

Performance measurement: this determines how fast and how reliably can information transfer be provided by the protocol in operation, in the case of different loads

Conformance testing begins by checking the basic and simple things then advancing step-by-step examines the more complex processes. Testing is performed by layers separately, however the protocols under the layers to be tested should operate correctly. The test of the protocol is made easy by the functional partitioning. Therefore the test procedures should be elaborated in such a way that they will be able to test the protocol functions one-by-one. Conformance testing is always proceeded by the protocol specification. Based on the specification, test sequences can be generated. The test sequence is one of the central conceptions of the conformance testing.

Conformance testing can be classified according to the point of view of the tester or of the system to be tested. The tester can be passive making only observations or active generating sequences. The testing can be executed from a remote node or the node to be tested. Initially, conformance testing was realized with individual experiments, the methods were difficult to be generated and repeated after changing the network environment. The solution was the composition of the abstract testing method similarly to the Reference Model of the open system.

The abstract model of conformance testing is based on two fundamental terms:

the controllability

the observability.

The protocol entity is a black box, only its input and output are available. Protocol Data Units (PDU) and Abstract Service Primitives (ASP) can be controlled and observed at the input and output. The identity of the protocol to be tested and the reference protocol entity can be measured on the surface fitting to the lower or upper layer of the protocol entity. The testing possibilities are somewhat generally demonstrated previously. The reference entity indicates the basis of the comparison, the standard protocol entity.

The architecture of the conformance testers has a layered structure similarly to the OSI model. The reference protocol and IUT are placed in the some layer. The test driver allows to start the testing and the reference protocol stimulates the entity to be tested with the reference service primitives. The test responder registers and evaluates the reactions on the stimuli arriving from the IUT and initiates (when required) new messages from the reference side to the protocol to be tested. The test driver also makes observations and evaluations.

ISO has elaborated a new notation for the conformance testing called TTCN (Tree and Tabular Combined Notation). In this standard are specified all constants, variables, types, PDUs, ASPs and generally all data which is necessary to know on the testing process. In the dynamic part of the TTCN is specified the rule of the communication between the tester and the IUT. The test suite described in TTCN is organized in test groups. Each group can test a specific function or a specific type of communication and is composed of a group of test cases. A test case describes one specific testing procedure on the IUT. The advantage of TTCN which is an event oriented language is that test case description is applicable regardless of the test system's realization.

Conformance testing of DLC

The DECT standard is based on the DECT Common Interface (CI) ETS 300 175 [2]. The test suite for this Common Interface DLC layer is presented in other ETSI standard: ETS 300 497-5 [3].

For testing the DLC layer protocol the embedded variant of Remote Single Layer test method (RSE) is applied (Fig IV.1).

The RSE test method has been selected, because:

- this test method implies no specific requirements from the IUT
- the Upper Service Access Point (U-SAP) of the IUT cannot be directly observed
- the variety of the possible DECT
- implementations is a serious technical obstacle for the adoption of a different ATM
- this test method places the minimum limitations in the realisation of conformance testing.

The Embedded variant of the Remote test method provides sufficient control of the IUT DLC behaviour, through NWK layer messages conveyed by DLC frames.

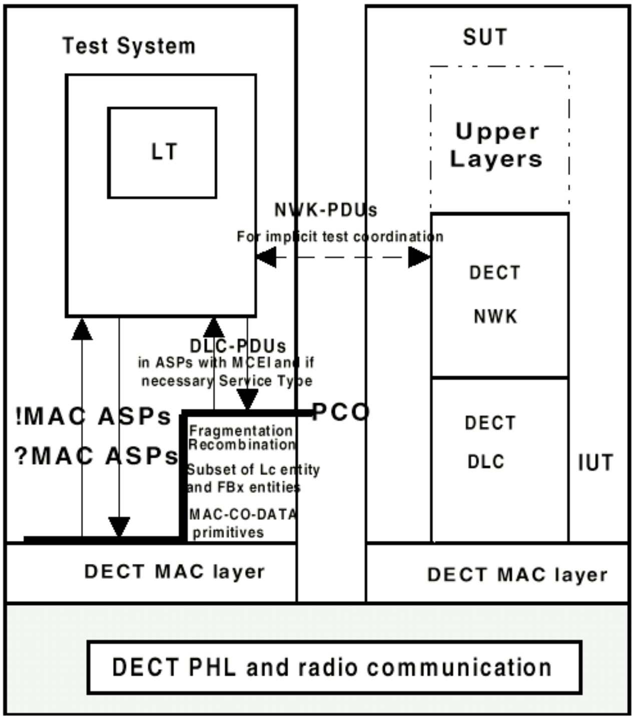



*Fig IV.1 DECT reference test configuration.*

From this part we will discuss the TC_A_BV_007 test case described in the test standard [3]. This test case purpose is to verify that the IUT manages rightly the PT intracell procedure for connection handover. The first step is to establish the initial condition. The initial condition means that the protocol can be in a state where the tester can verify its reaction for different sent events. In our case this initial condition is made by a preamble that brings the IUT into information transfer phase. In the next step we have to decide if we test a portable terminal or a fix terminal. If we test a portable terminal we must force this PT to create a new connection for connection handover. To effectuate this handover we must jam the currently occupied channel, to force an intracell connection handover. If the tester receives an indication of the new connection created by the IUT we can continue the testing procedure with an other preamble where the IUT checks if the new connection is in information transfer phase. If this preamble returns with PASS the handover was successful. In other test case if we need to test an intercell connection handover the "forcing" procedure is more complex because we need to force the PT to establish a new connection with an other Radio Fixed Part (RFP). For this the test operator must power down the signal strength of the currently used RFP stepwise by 1 dB/sec to force handover to a different RFP. In the final step of the test case the tester have to reestablish the initial state using for this operation a postamble. This operation is necessary for reconstruct the original stable state.

conclusion

This paper introduced the basics for conformance testing of DECT Data Link Control protocol. First the DECT system and DECT protocols were discussed, then the conformance testing. The principles of conformance testing were applied to the DLC protocol. An illustrative test case example demonstrated the test procedure.

The activity of the Hungarian Communication Protocol Laboratory (MTA-SZTAKI) focuses on the protocol engineering tools used for mobile communication protocols. These tools support the protocol specification, validation, implementation and testing. To implement the increasing number of mobile systems is much easier based on special protocol engineering tools.

Abbreviations

| | |
|---|---|
| CI | Common Interface |
| CT | Cordless Telephone |
| DECT | Digital Enhanced Cordless Telecommunication |
| DLC | Data Link Control |
| FP | Fixed Part |
| FT | Fixed Radio Terminal |
| GNW | Global Network |
| HDB | Home Data Base |
| ISDN | Integrated Service Digital Network |
| IUT | Implementation Under Test |
| IWU | Interworking Unit |
| LAN | Local Area Network |
| LNW | Local Network |
| MAC | Medium Access Control |
| PBX | Private Brach Exchange |
| PDU | Protocol Data Unit |
| PHS | Portable Handset |
| PP | Portable Part |
| PSTN | Public Switched Telephone Network |

| | |
|---|---|
| PT | Portable Radio Termination |
| TTCN | Tree and Tabular Combined Notation |
| VDB | Visitors Data Base |

## References

ETSI Technical Report ETR 056: "Radio Equipment System (RES); DECT; System description document".

ETSI standard ETS 300 175-4 “Radio Equipment and System (RES); DECT; Common Interface (CI) Part 4: Data Link (DLC) Layer”

ETSI standard ETS 300 497-5 DECT DLC Layer Abstract Test Suite

ETSI Report ETR 183 “Radio Equipment and System (RES), DECT Conformance Testing on DECT equipment”

K. Tarnay, Protocol Specification and Testing, Plenum Press New York 1991

Boja Zsuzsa Harangozó, Mária Tör• , TTCN and a Hungarian software, Selected Paper on Hungarian Telecommunication 1996, pp.25-30

M. Toro , K. Tarnay;Principles for validation of abstract test suites specified in concurent TTCN PSTV’95 Warsaw, Poland, 13-16 June 1995

J.E.Padgett, C.G.Gunther, T.Hattori: Overview of Wireless Personal Communications, Jan. 1995, pp.28-41.

# Information Billing System

M. Ignatiev, T. Karpova, S. Kruglov, Y. Pozdniak, A.Tsvetkov

St-Petersburg State University of Aerospace Instrumentation
67 Bolshaja Morskaja ul.,St-Petersburg 190000 RUSSIA

## Abstract

The Information Billing System, hereinafter referred to as “IBS”, or “the BIS”, is a programme complex to be used for the automation of the tasks of servicing and providing of services to the subscribers of an operator of cellular communication of the GSM Standard. The creation of BIS is the complex task and in our paper we consider the several problem of this task: the structure of activity in this system, the main functions of this system and corresponding programme complex.

## 1. DESCRIPTION OF PARTICIPATORS OF THE TECHNOLOGICAL PROCESS

The first problem is the system analysis of activity of the main dramatic personae:

Operator
An operator of the cellular communication who implements the Billing Information System of GSM.

Customer
A person who has concluded an Contract for servicing with the Operator in accordance with which Contract he can acquire and own one ore more SIM-Cards.

Subscriber
A person who uses SIM-Card of the Operator for obtaining the services of cellular communication; a number of the services ordered is determined for him, and the charges for the services provided are attributed to him also.

Group customer
A group consisting of two or more participants, who declares the common usage of payments received from any of the participants of the group, by their subscribers.

Roaming customer
A person who is in fact a customer of one of the companies (companies providing the services of GSM mobile communication) with which company a roaming Contract has been concluded by the Operator, and which person is provided with services in accordance with the said Contract.

Dealer
A company which concludes a servicing Contract with a customer, on behalf of the Operator.

Delivery Service
A service which organizes and carries out the delivery of bills and notifications to the customers. The service receives the printed out bills and the delivery materials from the group of accounting. It also provides the Operator with the customer's addresses defined more exactly.

Service Group (Back Office and Front Office)
A structural division of the Operator, which division interacts directly with the customer. Members of the Service Group are the main users of the IBS. They use the "Customers" application.

Payments Division

A structural division of the Service Group or the accounting group, which deals with input, distribution and correction of customers payments (according to the accountants information). They use the Customers application (illegible) of Payments).

Main Cashier's Office

A structural division of the Operator which performs the receiving of cash payments and payments via credit cards from a customer, receiving of initial payments and repayment of funds due to expenses orders. The Main Cashier's Office is located directly by the service division.

Remote Cashier's Office

A structural division of the Operator which performs the receiving of cash payments and payments via credit cards from a customer.

Book-keeping

A structural division of the Operator. It receives the information on receipt of customers funds onto the Operators account, and transfers the information to the Payments Division. Besides, it receives monthly reports on condition of balances of customers.

Operator of the BIS

A functionary of the company, who provides control of the IBS functioning, running of the information archives, verification of the condition of the database against the switch registers condition, reserve archiving and restoration of the database.

Administrator of (unreadable)

A functionary of the company, who is in charge of access of the users to work with the IBS BIS.

Technical group

A structural division of the Operator. Provides customers with SIM-Cards and information about SIM-Cards on a floppy disk.

Specification of instrumental and software means
A configuration of the instrumental means provision as well as the BIS software provision, determined by the BIS supplier.

## 2. GENERAL REQUIREMENTS FOR THE BIS

The BIS provides:
information support of the activity of the members of the Service Division.
exchange of data with the switch and the other auxiliary sub-BISs.
exchange of data with the Remote Cashier's Offices and Dealers on base of the existing operational technology.
exchange of data with the Remote Cashier's Offices and Dealers on base of ORACLE Web Server.
monthly billing procedures.

Accounting
The BIS provides the storage of data and technology of operation in accordance with the rules of book-keeping applied on the territory of Russian Federation.

User Interface
The subBISs operating under the control of Windows 95 are interacting with the user by means of a state-of-the-art graphical interface.

BIS Development
The BIS is developed on base of state-of-the -art solutions and open platforms providing if required, to carry out the further development of instrumental and software means.

Productivity
The BIS must comply with the following requirements of productivity on the recommended instrumental and BIS means:

the duration of performing the monthly billing (without an account of time taken for reports compilation and bills print outs) should not exceed 20 hours.
tariffing of calls must be performed on the inputting of the calls from the switch.

the duration of compilation of a bill due to a particular customer should not exceed one minute from the moment of reception of the accounting data from the switch.
a period of time from receiving of information on a payment till the switching of the services ordered by the customer should not exceed 5 minutes.

### Dependability

Dependability of the instrumental means of the BIS is provided by the use of highly reliable and fault-proof equipment. During the software development reliable BIS programming means and pre-tested algorithmic solutions were used.

### Security

Means of information protection and access discrimination are widely used in the BIS, the means being provided by the HP-UX system software and the Oracle software. All the actions with a potential of undesirable consequences, performed by a user, are registered in the BIS providing the possibilities of their further analysis.

## DESCRIPTION OF THE SUB-SYSTEM MAIN SUB-SUBSYSTEM

### Interface with Main and Remote Cashier's Offices

The main purpose of the given application is the provision of information support of cash desks operation and inputting into the BIS the data on cash payments made by the customers. The communication with the database server is carried out by means of ORACLE WebServer and ORACLE WebAgent. In the Remote Cashier's Office, a standard browser WWW (of a NetScape type or MicroSoft Internet Explorer) is installed, and the interface with a user is carried out by means of dynamic generation of html-pages. The physical linkage with the WebServer is established via the local network (Ethernet).

### Customers

This is the work place of a member of the Service Division. The information on new customers is transferred into the BIS and the current work with customers is carried out with the use of the given application.

SIM-Cards

This is the work place of an employee in charge of the operations with SIM-Cards, account of SIM-Cards, initial coupling to telephone numbers, allotment of SIM-Cards for direct sales and sales to dealers, compilation of orders and co-operation with dealers.

Input of bank payments

The input of data on banking payments made by the customers is carried out with the help of this application.

Payments reference books

This application is intended for the input of the US Dollar rate of exchange, as well as for logging various reference tables.

Control centre

This is the application, run on an allotted computer in the local network of the server of the database, and it is used for transmitting the commands from the orders processing monitor, or other applications, to the switch, the voice posting BIS, and the other external devices. The applications are recording the commands to the given sub-BIS in an agreed format, in a form of a text file, into an allotted directory onto the hard disk of the allocated computer. The sub-BIS translates the command into a sequence of MMl commands and transfers this command to the switch, emulating mode of operation of a terminal or, provides an access to the BIS of the voice post control. A response to the application which has sent the command for the execution is formed in a form of a file of the agreed format, which file is allocated within an allotted directory on the hard disk of the computer.

Input of registration records from the Switch

The application is intended for the input and primary processing of registration records from the Switch and for transferring of information to the database server of the billing BIS. The application operates on the registration records primary processing server emulating the operation of console of the switch via RS232. The registration records received from the switch are processed and reduced to the agreed format and then allocated within the table of calls on the database server by this application. During the preliminary processing, the transformation of the information from the registration files of the switch into text format and highlighting of admittedly fallacious records take place.

Tariffing

This application is used for the tariffing of registration records, received from the switch, on base of the tariff plan assigned to a subscriber.

Billing

This is the subBIS performing compilation of bills for customers, during a monthly billing and extraordinary billing among a group of customers.

AUXILIARY SUB-BISS

Input of data on new SIM-Cards

The main task of this application has been inputting of the information on new SIM-Cards on base of a diskette containing the information supplied from the technical division, into the BIS. The application is run on an IBM type computer, within Windows95 media. Communication with the database server is performed by means of SQL*Net via Ethernet network.

Capacity of numbers

This is a service application for the provision of operations with the capacity of numbers. The application is run on a personal computer of an IBM PC type within Windows95 media. Communication with the database server is performed by means of SQL*Net via Ethernet network.

Tariffs

The application is intended for tariff plans input, on which plans base the tariffing of the services provided for customers is carried out.

Service

Work place of the operator of the BIS. It is intended for inputting the adjustments information of the BIS and BIS efficiency control.

File interface with dealer

The sub-BIS is used for the processing of a file containing the information on a new customer, which file is transmitted from the dealer via modem in an agreed form.

File interface with cash desk

The sub-BIS is used for the processing of a file containing the information on a payment accepted in Remote Cashier's Office and transmitted via modem for the processing within the BIS.

Input of data on roaming

The sub-BIS is used for inputting the data on the roaming partners.

Reports

The sub-BIS is used for automatical sending FRAUD report to the roaming partners.

Automatical roaming

The sub-BIS is used for TAP files formation and processing of the TAP files received from the roaming partners.

# 3. IMPLEMENTATION PLATFORM

The information billing BIS is implemented within a customer server technology. Customer applications are run on personal computers of IBM PC type. Communication with the database server is carried out by means of Ethernet.The operation of specific work places of the users of the BIS is provided within the framework of the Intranet technology.As the customers applications work stations, IBM compatible work stations are used. The basic configuration of a work station includes: Pentium Processor, RAM 32Mb, at least 200 Mb free space on the hard disk of the computer, 17" Monitor. Russian language version of Windows95, and Explorer 3.x or higher should be installed in a work station.As a database server, a high efficiency HP9000 type server should be used. The database server specifications should be agreed additionally between the Operator and the Contractor.

## 4. SOFTWARE MEANS

Database server

As a software for the Database Server, the ORACLE server is used, the latter server working under HP-UX (Unix) operational BIS control.

Web Server

For organizing of a remote access to the Database Server, Oracle WebServer and Oracle WebAgent are used.

Network Interaction

For organizing of communication between customers applications and the Database Server, SQL*Net is used, the exchange of the messages being performed due to TCP/IP protocol.

Operational media

The Database Server operates under HP-UX (Unix) control.

Customers applications operating via Ethernet, are controlled by Windows95 (Russian language version).The operational BIS of the work stations used for the remote access to the Database Server must provide a possibility for the operation via Internet BIS. As a customers operational part in this case one of the standard browsers WWW is used.

Fax-server

As a fax-server, a commercially available software is used, the software supporting the operation due to MAPI protocol.

## 5. CONLUSION - INFORMATION ARCHITECTURE OF THE BIS

Information billing BIS consists of customers applications (operating in the local network of the Database Server), applications server (which is provided with the help of Intranet), the Database Server (on which the operational information is saved), the server of preliminary processing of registration records, archives server, (the base structure of this archives server is similar to that of the Database Server; loading and removal of the information to (from) the archives server is performed by means of Oracle Recovery from the archive sources of information in the manual mode), teaching server (its structure is identical to the Database Server structure, but it contains the information of the testing database), WebServer (for organizing the interface with remote work places), Sub-BISs for the Switch control and for the voice post control, fax-server.Under the availability of three servers (database, archiving and teaching servers), a user, when entering the BIS, defines the server to which he wants to get an access and the further work of the user is performed with the server selected. Archive and teaching servers can physically be allocated on the server of the preliminary processing of registration records. As a teaching database, the testing database created for the sub-BISs acceptance tests is used.

## References


1 M.B.Ignatiev “The structure of human communication cyberspace” Abstracts of ICS-NET 97, International Symposium on problems of modular information computer systems and networks. Moskow-St.Petersburg, 1997

2. ORACLE guide documentation. 1997.

3. T.S.Karpova “ Data  and knowledge bases in intellectual computing systems” GUAP, St-Petersburg, 1992

4. M.Ignatiev “ The problem of the sustainable development” Proceedings of the International conference on informatics, St-Petersburg< 1997

5.T.S.Karpova “ Automatic model for information documentary systems of acceptance of the decisions” Abstracts of ICS-NET 97, International symposium on problems of modular information computer systems and networks. Moscow--St-Peterburg, 1997.

Biografical note

Professor Mikhail B.Ignatiev,Doctor of technical science,Honorable scientist of Russia, Laureat of the State prize of USSR is the Head of Computer systems and networks Department of St-Petersburg State University of Aerospace Instrumentation. He published more then 400 scientific articles, inventions and books.

Docent Tatyana S.Karpova, Kandidat of technical science, published more then 50 articles.

S.Kruglov, Y.Pozdniak, A.Tsvetkov
are the students of St-Petersburg Stae University of Aerospace Instrumentation.

# Session 4:

# "Research Programs"

# Overview of the Moby Dick Project

Ger ard J.M. Smit, Paul J.M. Havinga, Sape J. Mullender, Arne Helme ♣
Gunnar Hartvigsen, Terje Fallmyr, Tage Stabell-Kulø ♦
Alberto Bartoli, Gianluca Dini, Luigi Rizzo, Marco Avvenuti ♥
University of Twente,
P.O. Box 217, 7500 AE Enschede, The Netherlands

## Abstract

The Moby Dick project focuses on developing theories, architectures and applications for a new generation of hand-held computers. The combination of an intelligent information system and a location system enables many new types of applications, such as admission control, digital chequebook, paging, and an automatic diary that keeps track of where you were and with whom. The design challenges lie primarily in the creation of a single architecture that allows the integration of security functions, externally offered services, personality, and communication. In the architecture, Quality of Service (QoS) is no longer a networking issue alone, but a framework to model integration and integrated management of all the system services and applications in the Pocket Companion.

---

♣ University of Twente, dept. Computer Science, P.O. Box 217, 7500 AE Enschede, the Netherlands

## 1. Introduction

In the Moby Dick project [10] we develop and define the architecture of a new generation of hand-held computers, the so-called Pocket Companion. With this Pocket Companion we shall be able to make telephone calls, make payments and we can use it to store all the information we now carry in our briefcases.

Several factors have hampered the advance of the use of electronic information in everyday life. One of the most important has been that the information cannot always be made available where it is needed; electronic information was never very mobile. This is now about to change. Digital wireless telephony can now be used to connect very small portable computers to the internet so that, wherever one is, one's data can be accessed and interaction with others is possible. Mobile computing, assisted by wireless networking, is an essential step in making electronic document exchange, electronic communication, and electronic commerce replace at least some portion of their non-electronic counterparts. This development will have to be accompanied by much better infrastructures for securing the privacy of one's personal data and the integrity and authenticity of financial and other transactions.

The combination of networking and mobility will engender new applications and services. Not only does it provide a means for users to stay in touch while on the move and to receive notifications of important events, it also gives people a whole new way to interact with the infrastructure of large public institutions, such as airports, supermarkets, or even whole cities. This interaction can be used for information about services, access to them and transactions with them. Standing in line for ticket or teller windows may become a thing of the past. Instead offices and public places will be equipped with access points, through which hand-held computer users will be able to communicate with the existing infrastructure.

♦ University of Tromsø, Norway
♥ University of Pisa, Italy

## A. The Pocket Companion

The technologies of PDA, digital cellular phone and, when combined and integrated well, have the potential of replacing all of the things people have to carry around with them by one small device, a Pocket Companion. This device is a small portable computer with a smart card and communications device that can replace a.o. cash, cheque book, passport, keys, diary, phone/pager and maps. The infrastructure should allow the user of the Pocket Companion to use e-mail and to access information services and data networks. The hardware basis of the Pocket Companion can be similar to the Personal Digital Assistants (PDA) on the market today, augmented with a security module and a wireless-network interface.

What makes a Pocket Companion different from a desktop machine? First of all its size and weight are small: a pocket companion has to fit into one's shirt pocket. Secondly, it is a truly personal machine: it might, for instance, contain sensitive information or store the owner's private keys or electronic money. And finally, it is a resource-poor device: there is a limited amount of energy in its batteries, there is only a small amount of memory, the CPU processing power is restricted and the communication bandwidth is moderate.

The Pocket Companion is more than just a small machine to be used by one person at a time like the traditional organizers. The Pocket Companion extends the notion of a so called ‘desktop companion’. A desktop companion is a hand-held machine that is designed to give roaming users access to their business data and applications while on the road (like a Windows CE system). Desktop companions are designed and optimized for compatibility and communication with the user's desktop machine(s), e.g. via modem, infrared or a docking station.

The Pocket Companion will run applications typically found in desktop companions, but it is designed to run other applications using external public services as well. A Pocket Companion interacts with the environment and so is part of an open distributed system. It needs to communicate with - possibly hostile - external services under varying communication and operating conditions, and not only to its desktop ‘master’. When Pocket Companions are incorporated into a global distributed system, their owners must be confident that the system is trustworthy.

## 2. Problem areas in Moby Dick

The design challenges of the Moby Dick project lie primarily in the creation of a single architecture that allows the integration of security functions, externally offered services, personality, and communication. In this paper we will address: QoS, security, networking, data consistency and energy management.

### *A. The QoS framework*

Applications will be used in a variety of computing environments. Some of these are distributed, and many make use of internet services. It used to be the case that applications were designed for particular computing environments, typically personal computers. But soon applications will have to run in environments that differ dramatically in processor performance, user-interface, communication performance and communication cost. Such applications will have to adapt their behaviour to the environment in which they run. Operating systems will have to provide assistance for this adaptation, which has become known as Quality of Service. The term stems from the notion that the quality of service an application can deliver depends on the resources that can be made available to it. Originally, QoS was used in the context of network communication resources in wide-area networks. Later it was also applied to systems resources needed for multimedia applications. In Moby Dick we have extended this notion to applications in mobile-computing environments [3].

The architecture integrates QoS management into every software module, and all modules are responsible for the collection of the QoS management information they require. In the design of a module, it is important to express both the resources it needs from other modules and the adaption that is required based on what resources the module actually gets. The design of software modules for the Pocket Companion therefore focuses on cooperation and adaptation issues rather than performance.

### *B. Personal Computing and Security*

When computers become more involved in people's personal and business activities security i.e. confidentiality, privacy, authentication, authenticity and non-repudiation become important concerns. Judicious application of cryptography can satisfy these concerns, provided systems provide a secure environment for users in which the appropriate cryptographic algorithms can do their work without any risk of compromising (losing) keys or confidential data. We believe that in order to use the full potential of these truly personal machines, the owner must trust his machine and he must be in full control over its machine, its information flow, and who can access it. In our view the security environment must be separated from the general-purpose computing environment.

The Moby Dick security architecture allows secure signing of arbitrary contracts between mutually authenticated principals. The architecture focuses particularly on the human/computer interface. People's digital signature will not be placed without their consent and only on the document they can see. Signature keys are managed such that their owners cannot divulge them accidentally, or be lured into doing so by a malicious expert; this helps to assure non-repudiation [7,8].

We have devised a hardware architecture where the Trusted Computing Base (TCB) is limited to a security module and supported hardware (i.e., explicitly excluding software). The architecture has been designed specifically to support signing of contracts in a (possible) hostile environment, a feature we believe will be crucial for a Pocket Companion. Execution of foreign code on your personal computer is not a new phenomenon, but additional work and experimenting is required to make it secure. We have built an experimental security framework for executing foreign programs, called helpers, on a Pocket Companion [12]. Helper programs have a profile associated with it, that specifies what files and resources will be accessed, the way they are accessed, and the capabilities of the helper. This mechanism uses a two phase approach: it checks the profile in order to make the decision whether to run the application or not, and after that it monitors the behaviour of an application.

### *C. Hybrid networks*

Future mobile information systems will be built upon heterogeneous wireless (possibly overlapping) networks, extending traditional wired networks to hosts moving over a wide area. Distributed applications in mobile-computing environments must be prepared to deal with partitions - a mobile computer will, at times, not be able to communicate - or changes in the communication infrastructure.

Although it should be possible to access all one's information via one wireless link, we must assume that most users will access their data from the type of network that is available or most convenient at any particular time. Mobile computers need to be able to move seamlessly from one communication medium to another, for example from a GSM network to an in-door network, without rebooting or restarting applications. The system must even be prepared to deal with complete disconnection from the network. Even then applications might continue, but are notified when inconsistencies occur. We have built a prototype implementation that allows all applications to keep their connections, while the computer switches from one network to another. The implementation of the Software Network enables multihomed mobile computers to switch dynamically and seamlessly from one network to another [2].

## D. Data consistency

Wireless networks will become faster and more reliable than they are today, but it is unlikely that they will become as reliable as today's local-area networks. Mobile computers, therefore, will be disconnected from the network from time to time and applications will have to deal with this situation [11]. Since user data will be accessed from different locations, sometimes concurrently (a diary may, simultaneously, be updated by a manager, her secretary and her husband), the underlying system must support distribution. However, unlike in conventional distributed systems, distributed applications in mobile-computing environments must be much better prepared to deal with partitions [1].

To give users reasonable performance over slow networks and reasonable service during network outages, storage systems must be distributed and do extensive caching. To give optimum service, distributed storage systems must be aware of the bandwidth and communication costs of the links that connect their parts. Storage systems will be a vital guardian of consistency for most applications. Partitions are possible and are expected primarily in wireless networks; when a partition occurs, file replicas (cached files) are accessible, but applications are notified when a possibility exists that inconsistencies occur. Applications must be prepared to deal with inconsistency. Simple applications, such as editors, can delegate the decision to edit a file under threat of inconsistency to the user. More complicated applications, such as a distributed diary application must implement the protocols to deal with inconsistencies and resolve them later, when the partition has been fixed.

The behaviour of an application may have to change as a consequence of network outages. The QoS manager must be aware, when making an appointment, that there is such an outage and that updates from the secretary may not have been able to reach it for a day. The user interface of a diary application must reflect such information.

### *E. Energy management*

The Pocket Companion is a hand-held device that is resource-poor, i.e. small amount of memory, limited battery life, low processing power, and connected with the environment via a network with variable connectivity. The increasing levels of performance and integration that is required will be accompanied by increasing levels of energy consumption. Without significant energy reduction techniques and energy saving architectures, battery life constraints will limit the capabilities of a Pocket Companion [5]. Because battery life is limited and battery weight is an important factor for the size and the weight of the Pocket Companion, energy management plays a crucial role in the architecture.

The power management system is based on three issues: system decomposition, integration of QoS management and communication. First of all, because the system architecture is decomposed of dedicated application specific subsystems that are connected with each other, energy consumption is reduced. Secondly, one of the key aspects of our QoS approach is to move power management policy decisions to the user and coordination of operations into the operating system. The operating system will control the power states of devices in the system and share this information with applications and users. Each module has its own - dedicated - local power management. Only the module is able to, and has the knowledge to implement the necessary power management fine-tuning of the internal functions. However, the overall power management control of the modules is done by the operating system and the user. So also here the user is in the ‘control loop’. Finally, as the wireless network interface of a mobile computer consumes a significant fraction of the total power, we put considerable effort in reducing energy consumption of communication interfaces [4,6]. There are several ways to achieve this: by system decomposition, using hybrid networking, and by applying power aware MAC protocols. We have developed a MAC protocol that is suitable for real-time multimedia applications and that has low-power characteristics [9].

## 3. Status and future directions

In the first phase of the project we have implemented a small set of applications to evaluate our solutions to a number of key problems and reveal the potential of the system. These demonstrators have given us a better understanding of the real world parameters connected to the research problems in the Moby Dick project. Currently we are building a testbed using off-the-shelf technology, that is several modules connected via a system-scale network. Since we envision that future hardware will be smaller with much more functionallity, we anticipate on this by doing experiments with equipment that has similar behaviour - except for size. The motivation for using off-the-shelf technology, even though it does not match our idea of hand-held devices, is that it will allow us to use existing development tools (e.g. compilers, debuggers, windowing systems, transmission software). This is essential to keep the size of the project reasonably small and to enable people to focus on the essence of the problems related to systems research and application design for Pocket Companions.

## References

All publication of the Moby Dick project can also be found on http://www.cs.utwente.nl/
~havinga/mobydick.html.

1. A. Bartoli: "Group-based Multicast and Dynamic Membership in Wireless Networks with Incomplete Spatial Coverage", ACM/Baltzer Journal on Mobile Networks and Applications, 1997

2. D. Brattli: "System support for hybrid networks", deliverable II.3.2 of the Moby Dick project, 1997.

3. T. Fallmyr, T. Stabell-Kulø, "QoS applied to security in mobile computing", submitted for publication

4. P.J.M. Havinga, G.J.M. Smit: "Minimizing energy consumption for handheld computers in Moby Dick", proceedings Euromicro 97, short contributions, pp 196-201, September 1997

5. P.J.M. Havinga, G.J.M. Smit: "Low Power system Design techniques for mobile computers", CTIT technical report 97-32, December 1997.

6. P.J.M. Havinga, G.J.M. Smit: "Minimizing energy consumption for wireless computers in Moby Dick", Proceedings 1997 IEEE International Conference on Personal Wireless Communication, December 1997

7. A. Helme, T. Stabell-Kulø: "Security functions for a file repository", ACM Operating Systems Review, 31(2):3-8, 1997

8. A. Helme: "A system for secure user-controlled electronic transactions", Ph. D. dissertation, University of Twente, 1997, ISBN 90-423-0011-6)

9. Linnenbank, G.R.J. et al.: "A request-TDMA multiple-access scheme for wireless multimedia networks", Proceedings Third Workshop on Mobile Multimedia Communications (MoMuC-3), 1996.

10. Mullender S.J., Corsini P., Hartvigsen G. "Moby Dick - The Mobile Digital Companion", LTR 20422, Annex I - Project Programme, December 1995 (see alsohttp://www.cs.utwente.nl/~havinga/pp.html).

11. L. Rizzo: "Effective Erasure Codes for Reliable Computer Communication Protocols", ACM Computer Communication Review, Vol.27, n.2, Apr.97, pp.24-36

12. G.J.M. Smit, P.J.M. Havinga, D. van Os:, "The Harpoon Security System for Helper Programs on a Pocket Companion", Proceedings Euromicro 97, September 1997, ISBN 0-8186-8129-2, pp 231-238

# Overview of the Mowgli Project

Timo Alanko, Markku Kojo, Kimmo Raatikainen
University of Helsinki,
P.O. Box 26, 00014 Helsinki

## Abstract

Modern cellular telephone systems extend the usability of portable personal computers enormously. A nomadic user can be given ubiquitous access to remote information stores and computing services. However, the behavior of wireless links creates severe inconveniences within the traditional data communication paradigm. In this paper we give an overview of the problems related to wireless mobility. We also present a new software architecture for mastering the problems and discuss a new paradigm for designing mobile distributed applications. The key idea in the architecture is to place a mediator, a distributed intelligent agent, between the mobile node and the wireline network.

## 1. Introduction

Developments in mobile communication technology have radically improved the information processing capabilities of man. Usage of modern cellular telephone systems is expanding dramatically; in the Scandinavian countries the penetration has exceeded 30%, and other industrialized countries are following at a rapid pace. At the same time most of the urban areas tend to be covered by mobile telephone networks. Hence, even nomadic users, not working in their offices, can now have ubiquitous access to various information processing services, such as retrieval of reports, access to e-mail, data base queries, WWW browsing. In addition, the current technology of personal computers has achieved a level where nomadic users more or less expect to get, even out of office, a similar quality of service as when working with their desktops. In practice, however, there are several reasons why these expectations cannot always be fulfilled.

The wireless world has behavioral characteristics for which the wireline world---neither the communication infrastructure, nor the applications---is prepared. The first aspect is the modest performance of the cellular telephone system: the throughput is low and the latency is high. The transfer of a 3 megabyte picture over a 9.6 kilobits/s link takes at least one hour---possibly a surprise to a customer used to retrieve imposing WWW pages in some few seconds.

A second source of problems is the instability of the wireless link. The user may move out of the coverage area, or the radio conditions may deteriorate. Transmission errors are detected by the wireless link protocol, and, to some extent, the protocol can recover from them. However, there is a cost involved: an increased delay. This delay may create problems in the fixed network. The protocols interpret increased delays as a symptom of a congestion, and trying to relieve the problem they decrease the transfer rate. This recovery action does not cure the illness, and it decreases the performance.

Changing conditions can even lead to a disconnection of the link. In fixed networks a broken link typically means an irrecoverable failure for the end user. In a wireless environment the problem may disappear within minutes or even seconds. However, current systems expect the user to repeat the whole dialup-login procedure, and the applications typically have lost the information that belonged to the interrupted session. In some field experiments up to 40% of the sessions were broken unexpectedly, and the user had to reestablish the session.

Protocols tuned for the traditional, well-known wireline world are sometimes grossly inadequate and sometimes even counterproductive when cellular telephone networks become involved. This is true for all protocol layers: link control, control of network usage, and control of application functionality.

The wireless telephone link is the bottleneck of the data communication path. Thus, tuning of the protocols that control the traffic over it is of vital interest, and this tuning should focus on new areas: local efficiency, adaptability to changing conditions, and ability to recover from certain types of failures.

We have developed the Mowgli[2] architecture to alleviate these problems [1]. The key idea is to place a mediator, a distributed intelligent agent, between the mobile node and the wireline network. The mediator controls the underlying wireless link according to rules specified by the end user. It can adapt its behavior to different conditions of the wireless link and to different wishes of the end user. In a disconnected state both of its parts can act rather independently in the role of the "non-accessible" partner of distributed computation. It can also be used as a platform for user-accessible control tools. In essence, there is a shift of paradigm: the traditional "client-server" paradigm has been replaced with a new "client-mediator-server" paradigm.

Similar kinds of mediator-based approaches have been introduced for wireless LANs. In these approaches the problems are treated only at the transport layer. However, when slow wireless links are involved, all protocol layers need to be addressed. In the Mowgli approach, we cover all the layers up to the application layer and the user interface.

A prototype implementation of the Mowgli architecture exists in an environment consisting of Linux and Windows platforms and the GSM network.

## 2. Mowgli communication architecture

Architectural Overview

In the Mowgli architecture a mobile node is connected to the fixed network through a wireless telephone link. The key idea in the architecture is to separate the control of the behaviorally different wireline and wireless worlds. The central role in this separation is given to the node on the border of these two

---

[2] Mowgli is the acronym of the project name *Mobile Office Workstations using GSM Links.*

worlds, which provides the mobile node with a connection point to the wireline Internet; the node is called the *mobile-connection host* (MCH). This node also offers a platform for intelligent agents, called *proxies*. The proxies represent mobile-node applications in the wireline network, and they are capable of operating autonomously, even when the mobile node is not reachable.

The mediator approach allows us to replace the regular

TCP/IP protocols of the Internet with a specialized set of communication services for the slow wireless link. These services are provided at three different layers: at the agent-proxy layer, at the data transfer layer, and at the data transport layer.

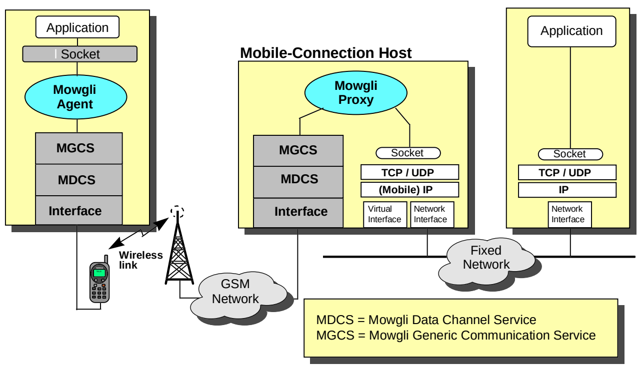



*Figure 6: Mowgli architecture.*

Figure 1 depicts the logical organization of the service architecture, provided on three different levels: on the agent-proxy level, on the generic communication level, and on the data transport level. The basic communication services, which are available through an application programming interface called the Mowgli socket interface, include the socket operations in the traditional BSD socket interface. Hence, existing applications can be executed on the mobile node without modifications.

This approach amounts to splitting the traditional client program into two parts---one on each side of the wireless link.

## *A* Agent and Proxy

On the agent level the communication services are implemented as agent-proxy teams. In the Mowgli system we have two basic kinds of teams:

*Generic agent and proxy* take care of delivering data over the wireless link; for each socket a pair is generated.

*Application specific agent and proxy* are tailored for a specific application protocol; when controlling data transfer they can adapt their behavior depending on the characteristics of the application, the needs of the user, and on the quality of service available in the wireless environment. The Mowgli WWW software is [4] a good example of an agent-proxy team taking advantage of application semantics.

In Mowgli the communication services available in the generic communication service level and in the data transport level are specifically designed to be used in communication over a wireless (telephone) link with an application-specific proxy.

## *B* Mowgli Data Channel Service

At the data transport layer, the TCP/IP core protocols are replaced with the *Mowgli Data Channel Service* (MDCS) [2], which is designed to cope with the special characteristics of slow wireless links. All communication above the data transport layer uses the transport services of the MDCS.

The communication in the MDCS is based on data channels. Each data channel has a priority and a set of additional attributes for controlling the behavior of the channel in case of exceptional events. Multiplexing of data channels is based on a priority scheme. This technique can be used to significantly improve the user-visible performance of data transfers. For example, when interactive communication is directed over a high-priority channel, the response times are in practice independent of the existence of any low-priority background data transfers. In addition, the user may start several simultaneous file transfers, which can be executed either serially or in parallel.

Another important feature of the MDCS is improved fault-tolerance. Particularly, the MDCS is designed so that recovery from unexpected temporary disconnections of the wireless link is efficient. In order to accomplish this, MDCS maintains the state information of the ongoing transmissions; after an interruption the transmissions can be resumed.

In the design of the *Mowgli Data Channel Protocol* (MDCP) we have solutions that are different from those of standard TCP/IP. There are several features in the TCP/IP protocols that do not suit well to the characteristics of the wireless links. The headers are long, and the usage of link capacity is wasteful. In order to adapt the transfer process to the current state of the network TCP employs sophisticated techniques, like the slow-start congestion control algorithm and adjusting of retransmission timers. While leading to proper behavior within a fixed network, these techniques are not suitable in a wireless environment [2,3].

The high performance of the MDCP protocol has been achieved by minimizing the amount of protocol overhead, specifically protocol headers and superfluous round-trips over the wireless link. In addition, MDCP always tries to transmit data at the best possible speed the wireless link is capable of; acknowledgments it also uses only sparingly. Thus, MDCP reaches the full bandwidth much faster than TCP, and it is capable of retaining the transfer rate rather stable even when temporary delays occur on the wireless link [2].

## *C* Mowgli Data Transfer Service

The *Mowgli Data Transfer Service* (MDTS) is one of the services offered on the generic communication service level. This service is able to take over the responsibility of transferring structured user data over the wireless link. The basic element of information for transfer operations is an *Information* eXchange Unit (IXU). In general, an IXU is something that the user considers as an independent unit of information, the transfer of which he or she may want to control. Examples of IXU's include mail messages, files, print jobs, and WWW pages or inline images of WWW pages.

Each IXU has a set of attributes, which are used in controlling the transfer. The MDTS provides operations for creating IXU's to be sent, for receiving IXU's, for managing transfer queues, and for changing attributes of specified IXU's. Due to the attribute system, the MDTS is able to operate rather independently of various kinds of background transfers: it can make decisions about invoking transfers when conditions are favorable, about postponing transfers when conditions deteriorate, about trying to recover from failures, and about canceling operations.

## 3. An example: WWW Agent and Proxy

The Mowgli WWW software [4] is an excellent example of customized agent-proxy team that is optimized for a specific application. Mowgli WWW design is based on the following key principles:

- Allow use of popular WWW browser and server software unmodified.
- Minimize round trips and the volume of data transmitted over the slow link.
- Discourage burstiness on all levels, since bursty data traffic translates to low utilization of the link.
- Allow operation without network connectivity.
- Enable asynchronous background fetching of documents.
- Provide the user with fine-grained control over how the wireless (or slow) link should be used.

The Mowgli WWW Agent and Proxy cooperate in order to fetch hypermedia documents from WWW servers to the mobile node. They communicate with each other using the optimized Mowgli HTTP (MHTTP) protocol, which minimizes the data traffic over the wireless link.

The Mowgli WWW Agent and Proxy establish a binding when the user starts a browsing session. The binding is maintained until the user declares that the browsing session is over. This means that the Mowgli WWW Proxy retains awareness of the user even over periods when the mobile workstation is disconnected from the network. In this way the proxy can perform operations in the fixed network on behalf of the mobile node when the node is momentarily unreachable.

The MHTTP is highly optimized, but the true power of the approach lies in the way the agent and proxy exploit available knowledge about the behavior of the application. An example: it makes extensive user of *predictive upload* of documents and document objects by prefetching document objects embedded in WWW documents. When the proxy receives a request for a WWW document from the agent, it fetches the document from a WWW server and forwards the document to the mobile node. While doing this the proxy looks at the document and starts prefetching the embedded objects from the WWW server and forwarding them to the agent with the assumption that the WWW browser will request them soon in any case. This technique significantly reduces the number of round-trips and makes the data traffic less bursty, which, in turn, reduces the response times observed by the end-user fetching typical WWW documents [4].

To reduce the transfer volume over the wireless link Mowgli WWW exploits three primary techniques: data compression, intelligent filtering, and caching. Mowgli WWW supports *content-type specific compression*: Each document type ---for example text, image data, audio data---can be assigned a different compression algorithm, such as lossy compression for images; after all, there may be no visible change in the quality of the image. *Intelligent filtering* is a drastic form of lossy compression, in which parts of the document are completely omitted. For example, the user can specify a maximum size for objects the system is allowed to retrieve without an explicit permission from the user. Furthermore, the Mowgli WWW Agent caches documents to avoid needlessly retrieving them over the slow link. This also enables browsing of cached documents in disconnected mode: the agent serves the WWW browser's requests from the cache.

Mowgli WWW implements a convenient *background transfer* feature. For instance, the user can instruct the agent to start fetching referred documents in the background while the user continues reading the current document or other documents already present. Background transfers use low-priority channels while foreground fetches always have a higher priority. This ensures that background transfers do not hamper interactive use.

## 4. Discussion

The conceptual design of the Mowgli system has its origins in the different natures of the wireless and wireline worlds, in some specific features of the wireless link, and in the problems of a nomadic user. The main idea is to develop new functionality using the concept of a mediator that consists of an agent-proxy team and of associated infrastructure.

This approach leads to an essentially improved performance when compared to the regular TCP/IP-based solution [2,4]:

- The overall performance was improved, and the number of very long delays was decreased dramatically; for example, in bulk data transfer benchmarks the upper quartile of the transfer time was decreased by 50%.
- Response times for interactive queries could be kept at a low and predictable level, independent of the background traffic; in the benchmarks the response times remained at the level of 2--3 seconds instead of about 8 seconds with regular TCP/IP.

The idea of splitting the end-to-end control turned out to be fruitful. Firstly, it helps to utilize the capacity of the bottleneck device, the wireless link. The proxy on the MCH acts as a high-level buffer balancing variability in arrival rates from the wireless side, which are due to distortions on the radio link, and from the wireline side, which are due to congestion in the Internet. Secondly, the splitting prevents irregular behavior on either side from harming the control of the other side: wireless delays do not cause unnecessary end-to-end retransmissions, nor does wireline congestion affect the operation over the wireless link. Of more importance for the end users are the qualitative improvements: occasional breaks in the radio link have no visible effects at the user interface level, automatic control of the telephone connection is a blessing for unaccustomed nomadic users, and the possibility to transparent background operations relieves the nervous end user from a lot of idle waiting.

Today the research in mobility is primarily concentrated on the problems of physical mobility: The system perceives how the nodes move. In the Mowgli environment the cellular telephone system hides certain issues in the terminal mobility like paging, hand-overs, and location updates. Therefore, we are able to focus on the next set of problems: when to connect, how to control disconnected agents, and -- in general --- how to improve the dependability of applications comprising wireless data communications.

## REFERENCES

1. M. Kojo, K. Raatikainen, and T. Alanko. Connecting Mobile Workstations to the Internet over a Digital Cellular Telephone Network. In Mobile Computing, pp. 253-270, Kluwer 1996.

2. M. Kojo, K. Raatikainen, M. Liljeberg, J. Kiiskinen, and T.Alanko. An Efficient Transport Service for Slow Wireless Telephone Links. IEEE J. On Selected Areas of Communication, 15(7):1337-1348, September 1997.

3. T. Alanko, M. Kojo, H. Laamanen, M. Liljeberg, M. Moilanen, and K. Raatikainen. Measured Performance of Data Transmission over Cellular Telephone Networks. Computer Communications Review, 24(5): 24-44, October 1994.

4. M. Liljeberg, H. Helin, M. Kojo, and K. Raatikainen. Mowgli WWW Software: Improved Usability of WWW in Mobile WAN Environments. In Proc. IEEE Global Internet 1996 Conference, pp. 33-37, London, UK, November 1996.

# The Ubiquitous Communication Program

The Ubiquitous Program P3 Consortium
contact: Hans Toetenel
Delft University of Technology,
P.O. Box 356, 2600 AJ Delft, The Netherlands

## Abstract

Recently, a large inter-faculty top-tech research program was initiated aimed at mastering the technology of the wireless communication of data, speech, and vision of mobile multi media applications. This position paper describes the overall structure of the program andhighlights in particular the aims of the subproject concerned with system and application specification, emulation, and evaluation.

## 1. Introduction

The objective of the Ubiquitous Communications Program (Ubicom) is to contribute to the extension of the functionality in wireless communication of a mobile person. The program's scenario is a campus-size wireless network providing to the mobile user broadband information exchange functionality. The high data rates envisioned ((10Mb/s) require operation at high frequencies (17GHz). The wireless network is connected to a broadband network backbone offering computing and storage resources (see Figure 1). The focus of the program is on hands-free visual communication with agreed measures of reliability, quality, synchronization and real-time performance. This focus is differentiating the Ubicom concept from autonomous wireless networks and mobile terminal networks.

The ubiquitous communications scenario and focus have several challenging implications to which the projects in the program address: the processing and displayingof moving pictures in a light-weight, low-power, head-up transceiver;the communication through high frequency, bi-directional wideband,multiple access links; the emulation and performance analysis of theoverall communication system; and the manipulation and compression ofaugmented reality and real-life image sequences. The assessment of the concept's feasibility is through a case study - a visual geographic information system in an educational setting - and the delivery of a specification of the corresponding demonstrator.
Ubiquitous Communications aims to produce five main results.

First, a specification of the overall communication system based on quality of service requirements, limitations, and benchmarks. This includes the specification of all subsystems and their interfaces. It also includes the specification of an embedded application and an assessing demonstrator.

Second, signal processing components in the high-frequency, high-bandwidth wireless network data links. This includes the analog front-end and digital core in the mobile person's low-power transceiver, the uplink and downlink multiple access smart antenna techniques, and the visual information head-up display implementing a retinal scanning technique.

Third, emulation and performance evaluation of the overall communication system, as well as the application on the scale of a demonstrator. It also includes software, hardware, and protocol issues related to the wired and the wireless networks.

Fourth, visual information processing for display (rendering), transmission (compression) and representation (object based), exclusively and as far as the needs go to adapt such techniques to the specific requirements that are imposed by the transmission link, the processing power asymmetry in the wired/wireless network nodes and the personal display technology. A virtual reality component will focus on the mobility aspect requiring integration of head positioning and orientation sensing into the mobile network infrastructure.

Finally, a geographic information system application and corresponding demonstrator specification, with emphasis on visual information storage, retrieval, and manipulation. Both application and demonstrator focus on personalized education on campus, so that available backbone services and application expertise can be fully and easily integrated. These applications are projected on the campus of the Delft University of Technology.

The Ubiquitous Communications Program consists of three projects: Access Points and Personal Transceiver (P1), Visual Information Processing and Application (P2), and System and Application Specification, Emulation, and Evaluation (P3). The program will pursue an application driven approach by first constructing a straw-man prototype consisting of available, but slow, components for wireless communication and an application core with little functionality. By the end of 1998, this test bed should be available for measuring key system and application parameters. The analysis and extrapolation of these measurements will consolidate and demonstrate available techniques, determine their strength and weaknesses, and steer the program in the direction of increased sophistication and improved quality.

The Ubicom program covers a broad research area and brings together partners from the departments of Electrical Engineering, Applied Physics, Applied Mathematics and Computer Science, and Geodetic Engineering.

## 2. Ubicom Architecture

In the Ubicom system, both hardware and software components are developed. Regardless of the actual applications being investigated in the Ubicom project, the systems architecture should be designed such that it is general enough to cover a wide range of mobile applications, it is sufficiently scalable in terms of resources, it should facilitate a convenient programming model for mobile applications and it can easily be interfaced with the outside world (including applications).

To give an outline, a picture of a general architecture is sketched in Figure2. In this setup, multiple mobile users can communicate with a cellular network of base stations. Because of the high frequency of the wireless link, the cell size of a single base station will be rather small (radius of 50 meters) to limit the effects of multi-path fading and other sources of errors. To reduce complexity and cost, a number of base stations is grouped together to increase the logical cell size. Each logical group is served by a compute station, which is part of the campus backbone. This is the current scenario, but may be subject to restrictions if more is known about the capabilities of the wireless communication channel.

Starting point is the limited processing capabilities of the mobile stations. Effectively this leads to a model where the main computational power is located at the backbone side (denoted as Compute Stations in Figure2). Possibly the mobile application requires additional resources situated on the campus network, but the application must be able to cope with the unpredictable long latencies on the campus network. This leads to a programming model where the application effectively runs over two or three sites: mobile station, compute station, and possible some remote server on the backbone.

Instead of sharing a compute station among multiple users, it seems to be more appropriate to let the compute station consist of multiple CPUs. In this way, more effective use can be made of the high bandwidth available for communication. It makes no sense to have high bandwidth available for multiple users and create a bottleneck in the CPU capacity of the compute station. This probably also makes it easier to guarantee certain quality of services.

If the mobile user is entering the domain of a different base station, the low-level communication software should reroute the traffic to the new base station transparent to the application. When the mobile user moves out of its logical cell, we plan to route traffic over the back-bone network instead of migrating the backbone part of the application to another compute station. This approach avoids the difficult problems associated with process migration, but probably degrades the quality of service because of the additional hop and long latencies on the campus network links.

It must be possible at the compute station to add and remove resources without reconfiguration or shut-down of the system, making the system flexible and fault-tolerant. If there are not enough resources available at the local compute station, free resources at other compute stations can be allocated provided that the campus network provides sufficient mutual interconnection bandwidth. If no free resources are available, degradation scenario's can be applied (e.g. sharing resources) in negotiation with the application.

An important aspect in the definition of the project is the expected properties of the mobile applications we are aiming at. We envision that these mobile applications will show a large variation in terms of algorithms and types of interaction. An application may have a number of open communication channels with the back bone, each logical channel having different QoS, error control, and compression characteristics. This implies that the software set-up on the mobile site can become quite complicated (e.g. concurrent running tasks) and must be sufficiently flexible to be able to accommodate the required functions. At the same time, resource limitations will impose restrictions on the the kind of functions allowed. Some functions, like compression, will be so critical in terms of performance, that they might require special hardware support in the mobile station. In that sense, the architecture of the mobile station should be designed such that it allows functions to be mapped to hardware when necessary (but still under software control).

## 3. The P3 project

The project concerned with System and Application Specification, Emulation and Evaluation (denoted P3 within the context of the Ubicom) constitutes of four dedicated areas, further characterized by: [P3.1] Overall System Architecture, [P3.2] Application programming environment, [P3.3] Resource scheduling and performance analysis, and [P3.4] Protocols and run-time system for mobile communication. These four areas will be introduced in the following subsections.

A. System architecture

This subproject is central to the complete Ubicom project. Its primary tasks are requirements analysis and specification of the overall system; the main role of the system architect is to manage and document the requirements analysis and specification process. After requirements analysis and system specification, the interfaces between the sub projects are to be defined and properly documented. Only after this phase design efforts in other sub projects can take place.

The next phase in this project is the design and implementation of a software simulator for the Ubicom system. This approach has been chosen to have a working system as early as possible in the program and to be able to identify problems that occur in putting system components together. In later phases components in the software simulator can be replaced by `real' components from other sub projects, as such having the possibility to gradually work to a demonstrator of the system.

Application programming environment

The programming model of a mobile application is a distributed programming model, where tasks have to be allocated to the various resources as sketched in the previous section. To realize this mapping, the application programmer must be provided with a set of abstractions that enables the realization of a mobile application. We call this the Application Programmers Interface (API). This API can take various forms, depending on the level of sophistication that can be reached in the project. Conceptually the execution model of an application can be viewed as a set of (possibly multiple) communicating threads through a number of channels on the wireless link.

At the lowest level of abstraction these communication channels simply move bytes between a thread at the back-bone and a thread at the mobile station. In this model the programmer is responsible for everything else (error control, scheduling, compression). This is clearly inconvenient, so a first research task in this subproject is to define an API that enables a programmer to define communication primitives at a higher level of abstraction. The kind of abstractions will have to negotiated with the project concerned with visual information processing and application. The API communication abstractions will be provided to the application programmer as a set of library routines.

A next research target is to investigate whether it is possible to build communication abstractions on top of a programming language or environment. The advantage of this approach, besides raising the level of abstraction, is increased flexibility and portability of application programs. A possible candidate language is Java, which also opens easy connection with outside environments. Such an approach brings object orientation and object sharing into the application programming environment. Programs written on this level will be translated to programs using the explicit API calls, making this approach compatible with the earlier derived API definitions.

Resource scheduling and performance analysis

Performance is a crucial aspect in the Ubicom system. The distribution of computation, combined with the need to minimize the data traffic between mobile station and backbone, implies that dynamic scheduling techniques be used that constantly optimize the performance of the ubiquitous communications system from a global users point of view as well as from an individual mobile user's perspective.

The performance quality of code and data scheduling in distributed real-time systems is highly dependent on the quality of the cost functions used to predict the temporal aspects involved with computation, synchronization, and data communication. Although in some cases scheduling can be done statically (i.e., at application code compile-time), in many cases scheduling decisions have to be delayed until execution time. This especially holds for resources whose availability can only be determined at run-time. Examples are the availability of communication channels with a requested QoS and available bandwidth on the campus network when a backbone server is consulted. Consequently, the performance models used must have a low solution complexity. At the same time, however, the predictions should be sufficiently accurate.

In typical scheduling techniques used for distributed-memory systems communication cost models are based on simple, analytic models that apply to individual communications and do not take into account the bandwidth degradation (network contention) that occurs under multiple, concurrent communications. Given the real possibility of the wireless communication subsystem becoming a (shared) bottleneck, performance models will have to be developed that are capable of predicting queuing delays as well as the basic latencies involved.

The main research task of this subproject is to define a real-time performance model and, based on this, to develop dynamic scheduling policies. It will be investigated if low cost, analytic models can be developed, that approximately account for the effects of sharing computation and communication resources.

Protocols and run-time system for mobile communication

The applications intended to run on the Ubicom system will require multiple,simultaneously open communication channels between the mobile station and thecompute station (and even other resources at the backbone). The communicationchannels will have different bandwidth and real-time requirements. Nevertheless, they must be multiplexed onto a single physical communication channel. Protocols and protocol stacks need to be designed that can handle the various data streams between mobile and compute station.

In software based communication systems, protocol stacks are usually embedded in a hierarchy of appropriately defined service access points (sap's) that merge and split communication streams. Although this solution provides optimal flexibility, performance is often limited by the complexity of the layers, OS system overhead, and copying of data.In the Ubicom system, performance might dictate that eventually some data streams must be handled directly in hardware, or selectively by-pass stack layers in software. This includes error control and compression (if applicable). This puts additional constraints on the protocol handling system. This should be designed such that flexible migration between hardware and software of functionality is possible (hardware/software co-design).

Another important aspect is fast data handling between communicating application threads. It is envisioned that a mobile station is equipped with some embedded micro-kernel (MK) capable of running a number of user and system threads. Whether this MK is resident or (partly) generated on the mobile station is not clear at the moment.

Open problems

Regarding the above P3 scenario, much of the implementation details depend on the options available in the wireless transceiver (P1) and the requirements from the application domain (P2). A short list of open problems is:

Can bandwidth be guaranteed during a certain time interval or not?

How many simultaneous users can be supported at a given time?

Will transmission be bandwidth symmetric?

Does compression, encoding, and decoding require special hardware, or can it be done at one of the resources of the mobile or compute station?

What is the nature of the data being communicated?

Will the application be able the handle transmissions errors in certain data streams, or should all communication be reliable?

The design and implementation of the Ubicom straw-man prototype, consisting of standard components and a simplified application core, will hopefully provide the answers to most of the above questions.

APPENDIX A

# Author Index

Timo Alanko
University of Helsinki
P.O. Box 26
00014 Helsinki
FINLAND

Pekka Ala-Siuru
VTT Electronics
P.O. Box 1100
90571 Oulu
FINLAND
pekka.ala-siuru@vtt.fi

M.M. Emilianov
St.-Petersburg State University of Aerospace Instrumentation
67, Bolshaya Morskaya
190000 St.-Petersburg
RUSSIA
mark@mark.stud.saai.ru

Paul J. M. Havinga
University of Twente
P.O. Box 217
7500 AE Enschede
THE NETHERLANDS

M.M.B. Ignatiev
St.-Petersburg State University of Aerospace Instrumentation
67, Bolshaya Morskaya
190000 St.-Petersburg
RUSSIA
kira@topol.stu.neva.ru

Mika Kasslin
Nokia Research Center
P.O. Box 45
00211 Helsinki
FINLAND
mika.kasslin@research.nokia.com

Tom Leskinen
Nokia Mobile Phones
P.O. Box 68
33721 Tampere
FINLAND
tom.leskinen@nmp.nokia.com

Frédéric Patricelli
Scuola Superiore G. Reiss Romoli S.p.A.
Via G. Falcone 25
67010 Coppito (AQ)
ITALY
fredy@ssgrr.it

Roman Pichna
University of Oulu
P.O. Box 444
90571 Oulu
FINLAND

Petri Pulli
VTT Electronics
P.O. Box 1100
90571 Oulu
FINLAND
petri.pulli@vtt.fi

J. M. Rivadeneyra Sicilia
Universidad del País Vasco/Euscal Herriko Unibertsitatea
P.O. Box 649
20080 San Sebastían
SPAIN

P. Sage
The Queen's University of Belfast
Belfast BT7 1NN
Northern Ireland, United Kingdom

Tapio Seppänen
VTT Electronics
P.O. Box 1100
90571 Oulu
FINLAND
tapio.seppanen@vtt.fi

Gerard J. M. Smit
University of Twente
P.O. Box 217
7500 AE Enschede
THE NETHERLANDS

K. Tarnay
MTA SZTAKI
P.O. Box 63
1518 Budapest
HUNGARY
tarnay@sztaki.hu

Hans Toetenel
Delft University of Technology
P.O. Box 356
2600 AJ Delft
THE NETHERLANDS
w.j.toetenel@twi.tudelft.nl

Rogelio Valencia
Universidad Pública de Navarra
Campus Arrosadía
31006 Pamplona (Navarra)
SPAIN