MIND Terminal Auto-adaptability to achieve a better user-perceived QoS in Networks Beyond 3G

Emilio García(1), Pedro M. Ruiz(2), Lars Kollecker(3), Tomás Robles(4)

(1)Agora Systems S.A, Spain, e-mail: emilio.garcia@agoratechnologies.com
(2)Agora Systems S.A., Spain, e-mail: pedro.ruiz@agoratechnologies.com
(3)T-Systems Nova, Germany, e-mail: lars.kollecker@t-systems.de
(4)Technical University of Madrid, Spain, e-mail: robles@dit.upm.es

ABSTRACT

Adaptive applications are a key concept to take into account when dealing with multimedia internetworking in wireless and mobile environments in which abrupt changes in the network conditions may occur. Using adaptation techniques at application layer is one of the solutions that allow multimedia applications to minimise the impact of vertical handovers in their performance, so that QoS perceived by the user remains in an acceptable level. This can be characterized by means of some tests and their empirical results.

I. INTRODUCTION

The IST MIND project[1] brings together major players in the mobile domain working on a vision of "systems beyond 3G". The overall aim of MIND is to facilitate the rapid creation of broadband multimedia services and applications that are fully supported and customized when accessed by future mobile users from a wide range of wireless access technologies. MIND is a follow up of the IST project BRAIN[2] (Broadband Radio Access for IP based Networks).

The technical approach of MIND takes as a starting point the concept of an IP core, accessed by a variety of technologies. Research is conducted in the areas of services and applications, the access network architecture and the air interface. The project follows a top down approach. First, the scenarios that need to be supported by future wireless networks are detailed. Then from these, the requirements on the network and air interface are derived and potential solutions studied.

MIND is intended to complement other access systems (e.g., UMTS). To this avail, vertical handover (handover between different access technologies) is studied and multi-homing terminal concepts are analysed.

Besides the conceptual work detailed above, the project is conducting trials on key concepts studied in the predecessor project BRAIN. Actually, the MIND demonstrations will implement selected key concepts of the architecture defined in the BRAIN project.

In this paper we focus on BRENTA[3] –which is the BRAIN End Terminal Architecture– functionalities. BRENTA design is split in two major planes: the usual data networking plane, and the QoS- and resource-management one. The former deals with data handling through the use of multimedia components (e.g. codecs) whilst in the latter BRENTA manages and coordinates local and network resources. One of the key points in BRENTA is the idea of auto-adaptation when QoS violations occur.

The demonstration of some BRENTA functionalities is reported in this paper, which is organised as follows: Section II describes the service concepts that are going to be tested within MIND. Section III comments on the different ways in which a real-time multimedia application can improve the user-perceived QoS. Section IV describes some QoS layer approaches to aid the application. Finally Section V shows some initial results.

II. MIND SERVICE CONCEPTS

During MIND, some service concepts have been studied and selected for demonstration in the trials inside the project. They deal mostly with the adaptive capabilities of the videoconference application that is going to be used, as well as some interfaces (mainly for QoS support), currently under development.

In the following sections some of the key points of the adaptability will be described.

A. Terminal Adaptability

In order to demonstrate in our trials the benefits obtained from the interaction between service and lower layers, the new architecture represented in Figure 1 has been designed. Several layers below the application can interact with the so-called "central control" entity, which possesses all the intelligence to make the application react to changes in its network environment. This 'middleware' includes the mobility protocol, tools like Traffic Control, or Control Protocol of RTP, which is used for the data flows. Also, through some interface, the application can enforce a set of parameters, as it will be described in the sections below.

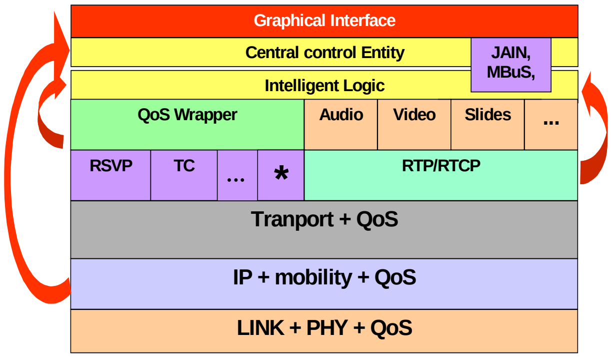

Figure 1: End-terminal architecture

This architecture is a simplification of BRENTA, designed for project BRAIN, and while covering only a subset of its functionalities, it will nevertheless help demonstrate some interesting concepts.

B. QoS Support

Quality of Service (QoS) as defined in ITU-T recommendation E.800, ITU-E.800[4], is the collective effect of service performance, which determines the degree of satisfaction of a user of a service. It is characterised by a combination of service performance factors such as operability, accessibility, retainability and integrity.
Our trials will handle QoS in a subset of the terms used above, as they play the most important role in subjective perception of quality of final user:

- **Throughput information** from the end to end data flow is essential to the application so that it can offer a coherent data rate to lower layers with minimum packet loss, thus aiding the achievement a requested QoS. Ways to achieve this go from switching application codec to increasing or decreasing encoding rate or picture size.
- **Delay** awareness in any point of the transmission can be used to resize outgoing and incoming buffers, and provide a continuous stream communication, needed for example in real-time videoconference applications.

However, important though it may seem for the application to receive feedback on those parameters, true QoS support needs of the capability to enforce in lower layers. This covers, for example, bandwidth reservation, or service type marking (Best Effort, Controlled Load or Guaranteed), as it is covered in Section IV.

C. End-to-end signalling

For our trials, we will consider the application working in a peer-to-peer fashion. It is obvious that any change in the network conditions that may trigger some adaptive behaviour must be informed in both transmitting and receiving sides, especially if it involves a reduction of available bandwidth due to a vertical handover.
Several ways to do this can be envisioned, from advanced SIP signalling to RTCP payload type detection. The last one will be the selected at a first stage for our tests.

III. ISABEL ADAPTATION CAPABILITIES

The ISABEL application[5] is being extended within the framework of the MIND project so that it can be used as a multimedia terminal being able to show most of the BRAIN service concepts. In this paper we focus on auto-adaptation capabilities as a means of achieving a better user-perceived QoS. That is, we have extended the application so that it is able to change (on the fly) the behaviour of its components in the case of some network problems or bandwidth limitation coming up.
The selected parameters to be changed in a possible auto-adaptation condition are as follows:

- **Codecs used for audio and video.** This allows the application to decrease the bandwidth consumption. Usually less bandwidth means less quality. However in these environments the user prefers less quality audio or video than packet losses, which may cause the audio and video tools not to be useful.
- **Audio sampling rate.** The sampling rate is proportional to the number of packets to be sent out to the network. Higher sampling rates mean better quality and higher bandwidth consumption.
- **Video size.** The bigger the video size, the higher the bandwidth consumption. In scarce bandwidth environments the user will prefer seeing smaller videos than bad quality ones in which most of the frames are lost.
- **Frames per second.** Transmitting at a lower frame rate means saving bandwidth. In most situations the application does not need to reach the optimum number of 25 fps to make the user 'feel' a fair quality. In addition, practical limitations from video cameras make the subjective better sensation be achieved by using 24 fps dividers: 1, 2, 3, 4, 6, 8, 12 and 24 fps. This means for example that the user perceives better quality when switching from 12 to 8 fps instead of changing from 12 to 10 and changing the codec as well.
- **Buffering.** Intelligent and dynamically adaptable buffers may help offering a better quality in adverse network conditions.
- **Components to use.** In very constrained bandwidth scenarios in which even using the lowest bandwidth consumption approaches, the user may prefer using some components instead of using all of them with a poor quality. For example, in a 40 Kbps scenario the user may prefer just receiving a GSM audio flow without any losses rather than receiving a

video with very poor quality and an unintelligible audio stream.

IV. QoS INTERACTIONS

The Internet supports a large number of users with a simple, best effort network, which ensures that all users receive a share of total bandwidth. However, Quality of Service (QoS) is needed if it is to successfully carry a significant quantity of real time services such as audio or video-conferencing; some packets are treated preferentially to achieve, for example, low transfer delay. Parameters that can be controlled include packet delay, packet loss, packet errors, available bandwidth and inter-packet delay (jitter).

Weaknesses in this solution appear when it is evaluated against requirements if both wireless access and mobile users are assumed. Wireless issues revolve around the restricted bandwidths and increased error rates of wireless links (multi-path fading, interference, etc.), and how techniques to overcome these problems may interact with higher layer QoS mechanisms.

Many services can tolerate some decrease in network performance, as in the case of file-transfer and web browsing. But real-time applications need of some mechanisms to maintain user satisfaction, like the ones described in Section III. Such adaptive behaviour, however, requires support from the network layer. At the very least, the network must be capable of reporting the current QoS, therefore providing the application with feedback; in addition, it is desirable that some traffic parameters can be enforced.

A. Reservation based and packet oriented services with IntServ and DiffServ

QoS may be achieved through per flow reservation. Here an application queries the network to discover if the QoS requirements can be achieved. Reservations make best use of resources allowing better planning of the network usage, and giving a more reliable QoS. However, there is a large overhead associated with this, as signalling messages are required and there is a delay before applications can start to send data. In the alternative prioritisation model, clients mark their packets to indicate a “premium” service requirement. The service may be used at any time, but the performance provided will be less predictable and may suffer from network congestion. Prioritisation is typically used with Service Level Agreements (SLAs) that may be general, or defined on a per-user basis. The solution chosen provides both reservation based and prioritisation services.

Per-flow traffic management means that the application's traffic is granted resources and protected from the effects of traffic from other users in the network. This enhances the quality of the service experienced by the application, but also imposes a burden on the network, which needs to maintain state for each flow and to apply independent processing for each one. In the core of large networks, where it may become necessary to support millions of flows simultaneously, per-flow traffic handling is not practical. Alternatively, when traffic is handled in aggregates the state maintenance and processing burden on devices in the core of a large network is reduced significantly. However, the quality of service is no longer independent of the effects of traffic from other sources. Over-provisioning of resources to the aggregate QoS traffic can offset this effect. However, this approach tends to reduce the efficiency with which network resources are used. Whilst we do not require any specific network internal QoS mechanism, the DiffServ aggregate mechanism is considered highly suitable because of its scalability and potential simplicity.

B. Priority classes

For audio transmission, we use only the expedited forwarding classes to ensure the Audio capabilities even in case of congestion of the wireless link. For the Video transmission two encodings are used: “Simple Video Encoding” and “Advanced Video Encoding”. The last one splits the stream into 1, 2 or 3 layers having descending priorities and may therefore have different DiffServ classes and probability of packet loss. It is now possible to do an error correction by replacing missing data of one Video channel by the data of the other channel. The IP packets are marked with IPv4 TOS/IPv6 Traffic Class byte given by the QoS management to support DiffServ usage. Using IPv6, flow labels are also used to support bandwidth reservation.

V. TRIALS

Some trials have been envisioned over a standard testbed, so that the adaptive characteristics of the application can be demonstrated, and also the benefits of such behaviour in contrast with the one found in usual applications.

The audio part of the ISABEL videoconference application has been selected as a fair sample of a possible variable bandwidth service over RTP[6]. The results can therefore be extrapolated to a case where more components (audio, video, text o a blackboard) are used, just by considering each one a separate data flow.

A. Hardware infrastructure

The testbed, as shown in Figure 2, consists on a number of standard desktop computers that provide all the routing functionalities, and a Laptop, which is used as the mobile node (MN).

Network links are mostly Ethernet, and an ISDN PPP link (64 Kbps), which serves the purpose of limiting the effective bandwidth in one of the branches of the testbed.

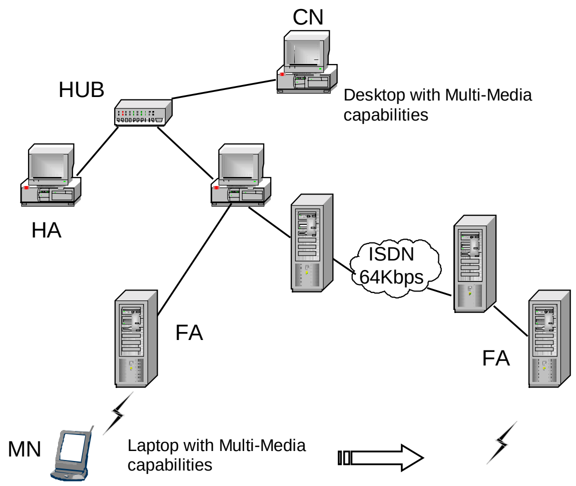

Figure 2: Testbed infrastructure

The MN will be forced to perform a handover between the two lowermost hosts, which act as access points for the two separate networks. This resembles a typical scenario of switching from a broadband network to one with low resources.

B. Software

All the systems run Linux Kernel 2.4.17, and have a dual IPv4/IPv6 stack. Currently only Hierarchical Mobile IP (HMIP[7]) in installed, but the same testbed will perform in the future when other Macro/Micro-mobility protocols, both over IPv4/6 are deployed. Roles for each node are noted in Figure 2.
The adaptable audioconference application to be used is an extended version of ISABEL, which is installed both in the MN and Correspondent Node (CN), so that communication can be established between both at any moment.
Among the whole set of adaptation techniques formerly described, changing the audio Codec has been elected as the most feasible to be done easily and in a short period of time. Two codecs have been used, one for high bandwidth and one for low bandwidth, as shown in Table 1:

Codec	S. Rate	S. Depth	Bit rate (mono)
PCM	16000 Hz	16 bit	~256 Kbps
GSM	16000 Hz	16 bit	~38 Kbps

Table 1 Used codecs

Data showed in the following tests has been acquired from the RTCP statistics collected at the MN side.

C. Operation without adaptation

The first set of tests covers the functioning of a simple application with no adaptive capabilities. Figure 3 shows the effect that a handover to the ISDN bandwidth limited network from the broader Ethernet has in the transmission:

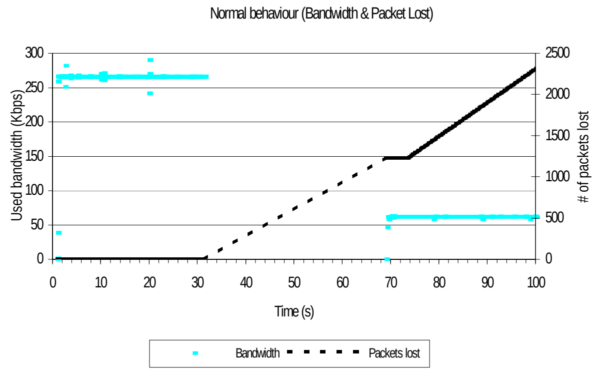

Figure 3: Bandwidth without adaptation

The handover occurs between seconds 30 – 70, so all the packets are lost in that period. That high-latency is only ascribed to the particular settings in this HMIP installation (mainly due to long agent advertisements periods), and cannot be considered as the usual performance of a tuned mobility protocol. However, for the purposes of these tests, which do not deal explicitly with mobility procedures, a strong disruption in communication helps in isolating the behaviour of the application after it is restored and therefore highlights the capabilities at service level.
It can be easily observed that, while maintaining the codec to PCM (and thus, the data rate sent from the CN to the MN at 256 Kbps), after the handover the limit is imposed by the 64 Kbps ISDN. Therefore, many packets in excess are lost (after an initial transitory, while the UDP buffers are being emptied). Packet loss is critical in real-time applications, and the effect is disturbing the audio quality experienced by both end users.
The transitory can be explained through the graphic of the interarrival Jitter (in Figure 4):

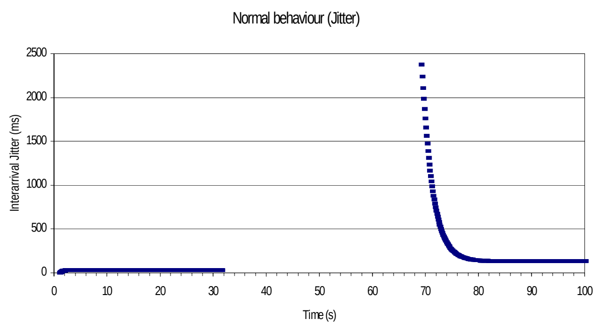

Figure 4: Jitter without adaptation

Immediately after the handover, packets stored in UDP buffer are retrieved by the application, but being old packets, their timestamp affect the jitter calculation in the same amount of time stored in the buffer (i.e., about 2.5 seconds). It is also noticeable that the jitter is increased to about 130 ms after the handover (while being 30 ms before it), and this fact worsens the audio experience.

D. Operation with adaptation

The adaptive behaviour of the application is triggered when it senses, by any way, that actual bandwidth has changed. In the simplest case, which is the one tested, the reaction consists only in the selection of the lowest bandwidth codec available (GSM), but a more complex logic could manage the selection of one among a showcase of different codecs, each one with particular characteristics. Figure 5 shows the expected behaviour.

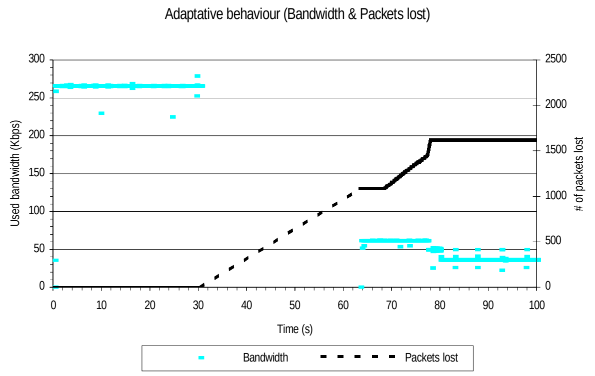

Figure 5: Bandwidth with adaptation

The first seconds after the handover ends are similar to the situation explained in Subsection C. The adaptation is ordered to the sender a few moments afterwards (around second 77 in the graph), and almost immediately the new codec is selected and the bandwidth consumption is reduced to less than 40 Kbps, enabling all the packets to go though the ISDN with no packet loss (flat part of the graph). This way, audio quality is considerably better than in the first case, when no adaptation took place. Jitter is shown in Figure 6.

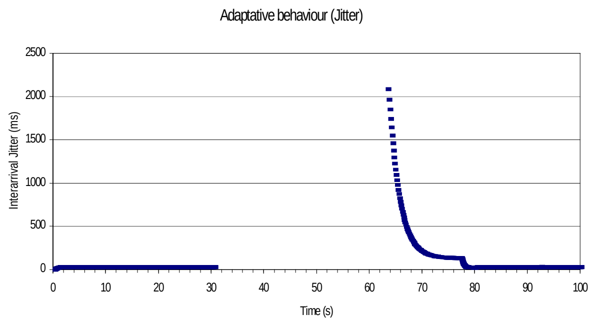

Figure 6: Jitter with adaptation

It is quite similar to the former jitter image, but it is quite noticeable that the changing of the codec allows a reduction of the jitter to the minimum value (30 ms): all packet traffic is processed in its proper time.

VI. CONCLUSIONS AND FUTURE WORK

In the context of the IST MIND project, some mechanisms of demonstration of service concepts developed in its precursor BRAIN have been envisioned. To that point, a versatile multi-media application has been used and adapted to be able to handle, and even enforce, the changes in its environment. Through a sample testbed and some simple experiments, this paper has demonstrated the benefits that such a technology can provide to enhance the experience of a end user in a typical changing conditions scenario, guaranteeing an adequate service level even in dramatic bandwidth changes.

Moreover, the described testbed is flexible enough to allow further experiments including handover with micro-mobility protocols, such as the BRAIN Candidate Mobility Protocol[8] (BCMP), and also tests over IPv6. Improvements in reaction are related to the implementation of some logic of codec selection from a wide pool, each one for a concrete situation. Total integration with lower layer QoS control facilities, as described here, is the final objective to reach a quite fair level of service quality in the scope of the trials of the MIND project.

ACKNOWLEDGEMENT

This work has been performed in the framework of the IST project IST-2000-28584 MIND, which is partly funded by the European Union. The authors would like to acknowledge the contributions of their colleagues from Siemens AG, British Telecommunications PLC, Agora Systems S.A., Ericsson Radio Systems AB, France Télécom S.A., King's College London, Nokia Corporation, NTT DoCoMo Inc, Sony International (Europe) GmbH, T-Systems Nova GmbH, University of Madrid, and Infineon Technologies AG.

REFERENCES

[1] IST-2000-28584 MIND, www.ist-mind.org.
[2] IST-1999-10050 BRAIN, www.ist-brain.org.
[3] A. Kassler et al. *BRENTA - Supporting Mobility and Quality of Service for Adaptable Multimedia Communication*, in Proceedings of IST Mobile Summit 2000, Galway, Ireland, October 2000
[4] ITU-T Recommendation E.800 (08/94) "*Terms and definitions related to quality of service and network performance including dependability*"
[5] ISABEL CSCW application http://agoratechnologies.com/productos/isabel.html
[6] H. Schulzrinne, S. Casner, R. Frederik, V. Jacobson, *RTP: A Transport Protocol for Real Time Applications*, IETF, RFC 1889, January 1996
[7] H-MIP code from http://www.cs.hut.fi/Research/Dynamics/
[8] K. Ceszei, N. Georganopoulos, Z. Turanyi, A. Valko, *Evaluation of the BRAIN Candidate Mobility Management Protocol*, Proceeding IST Summit '01, September 2001.