Design and Development of Several Mobile Communication Systems for People with Hearing Disabilities

Jose L. Martín[1], Sira E. Palazuelos[1], Jerónimo Arenas[2], Javier Macías[3], and Santiago Aguilera[4]

[1] Department of Electronics, Polytechnic School, University of Alcalá, Campus universitario s/n, 28805 Alcalá de Henares, Madrid, Spain
{jlmartin, sira}@depeca.alcala.es
http://www.depeca.uah.es

[2] Department of Communication Technologies, Polytechnic School, Carlos III University
jarenas@ieee.org

[3] Speech Technology Group, Department of Electronics Engineering, E.T.S.I. de Telecomunicación, Technical University of Madrid
macias@die.upm.es

[4] Department of Photonics, E.T.S.I. de Telecomunicación, Technical University of Madrid Madrid, Spain
aguilera@tfo.upm.es

Abstract. During the last decade we have attended to an impressive development of mobile communications which, unfortunately, deaf and hearing-impaired community cannot (in principle) take advantage of. In order to favour this people to take part in the Information Society, we have designed and developed some PC-based systems and applications which will provide several text-based services to them, such as real time and text mode communication between mobile text telephones and those connected to the PSTN or direct accessibility to Urgency Call Centers. We emphasize in this article the advantages of the software design methodology followed, which has led to the implementation of two systems which have shown to be robust and versatile in operation.

1 Introduction

The invention of the telephone in 1876 by Graham Bell, improved the social welfare of millions of people, by allowing the communication between distant places. At first, this communication system could not be used by the hearing-impaired people; however, the development of the first text telephone terminals in the seventies changed this scenario. In these devices the microphone and receiver of a conventional telephone are substituted for both a display and a keyboard, allowing real time and text mode conversations. Although there are different models of text telephones (we will refer in the rest of this article to the DTS, the most widespread terminal in Spain), all of them present a common drawback. As

K. Miesenberger et al. (Eds.): ICCHP 2006, LNCS 4061, pp. 675–682, 2006.

they don't use standard communication protocols, only communications between terminals of the same kind are feasible.

Moreover, with the appearance and increasing significance of mobile communications one more challenge joins in the final objective of a global communication system completely integrated for the deaf community.

Bearing these ideas in mind, a standard mobile telephone (Nokia Communicator) was provided with a software application in order to allow deaf users to maintain a text mode conversation between two of these terminals, or one of these and a PC connected to the standard PSTN network via modem.

However, there were still many facilities which could be implemented to further improve the telecommunication capabilities of deaf people. In this direction, we have face two new aims that will be described in this paper:

– making the communication between a mobile terminal and the conventional DTS possible
– and providing Urgency Call Centers with an integrated communication system to receive calls from Nokia Communicators.

In order to implement these functionalities, we decided to develop two different and separated solutions: an Automatic Interworking System (AIS) and a dynamic library (SUMMA061). **AIS** is a complex PC based system with the functionality needed to establish real time and text mode communication between Nokia Communicators and DTS terminals from end to end (see Fig. 1). **SUMMA061** is an specific dynamic library designed to be integrated in the emergency Call Center standard software to attend Nokia communicator calls (see Fig. 2).

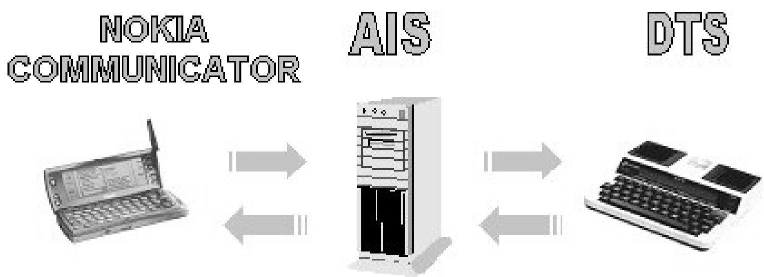

Fig. 1. Communication between Nokia Communicator and DTS, through the AIS

Though the Nokia Communicator-DTS communication was at first our main concern in the conception of what AIS should be, we soon realized that this system could host many more communication services for deaf people. By using an structured analysis and design methodology ([1]) (summarized in Table 1) we have tried to develop both systems in which new functionalities can be added in an easy and modular way.

Fig. 2. Communication between Nokia Communicator and emergency Call Center, through the AIS

Table 1. Structured Analysis and Design Methodology

Step	Description
1.	User Requirements Analysis
2.	Hardware Design
3.	Software Design (leading to a modular software)
4.	Implementation of the System
5.	Proof of the different modules
6.	Proof of the whole system by the users

2 System Requirements

Requirements imposed by users for the communication between the DTS and the Nokia communicator were:

– AIS must provide text conversations between DTS and Nokia Communicator without modifying (unless strictly necessary) the actual user interfaces (so the user doesn't need to get used to new protocols for making calls). Any modifications made, must be very easy to understand and manage.
– Telephone calls might be originated and released by any of the sides of the conversation.
– AIS must provide resources for up to 30 simultaneous conversations in real time.

For the addition of new communication services to AIS:

– AIS must have a modular software design which facilitates the incorporation of new services offered to the hearing-impaired community.
– The addition of new facilities must be carried out in a systematic way.

And finally, requirements for the SUMMA061 library were:

– The application must be absolutely robust to attend any kind of emergency call without hanging up.
– SUMMA061 should be fully integrated in the standard call center software (that keeps medical data, personal treatment information, etc. depending on the kind of emergencies attended at the call center).

– The application should allow encoded recording of every call, storing the number phone, length and content of the conversation for later analysis.
– Application access should be protected by several access levels permissions.
– Lost calls must be easily recover.

3 AIS Development

3.1 Hardware Design

AIS hardware architecture must supply, at least, means of communication with terminals both connected to the GSM and PSTN/ISDN networks. However, as we will discuss, a more general solution has been adopted. Mobile telephones will access AIS through TCP/IP sockets, which provide them with a connection oriented path, guaranteeing the real time requirements at this side of the conversation. This was not the only available solution, but it has the advantages of cheapness (just a LAN board with an IP address are needed), capacity (there is not a limit in the number of mobile terminals which can be connected to the AIS at a time), simplicity (lots of programming tools with sockets support are available, even at zero cost) and versatility (the Internet connection can also be used for other purposes).

For communication with DTS, AIS will use an ISDN primary access (E1), which allows up to 30 simultaneous calls. This interface will be managed by a Dialogic D/300-PCI board. Dialogic API implements useful high level routines ([2]) for dialing a number, putting the line on hook / off hook, playing a file, or recording the sound received, simplifying enormously the programming tasks.

Fig. 3 shows a non exhaustive communication scenario resulting from the decisions discussed above. It represents not only the terminals involved in a DTS-Nokia 9110 conversation, but also other terminals that can get to the AIS through the telecommunication networks depicted (for future communication services).

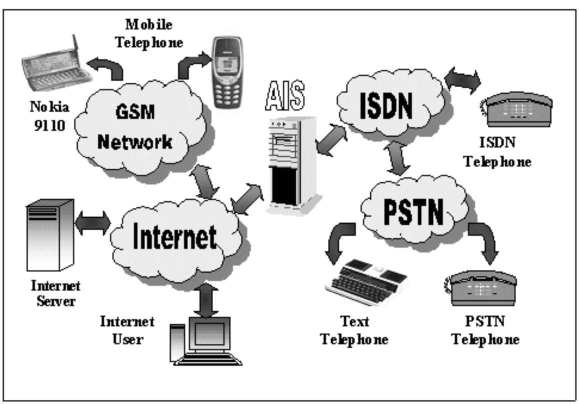

Fig. 3. Communication possibilities of AIS

3.2 Software Design

AIS has been designed using a descendent splitting scheme: in a first step we split the functionality we want the AIS to implement, into well differentiated units. Next, we divide each unit into smaller ones until we get modules small enough to easily program them. The first splitting results in the three blocks:

The Control Unit is in charge of managing all the resources of AIS (memory, B channels on ISDN interface, TCP ports, etc). In principle, the AIS is to attend up to 30 end to end communications of any type (DTS-Nokia Communicator must be treated just as a possibility at this point of design), each of then requiring (at most) a B channel, a TCP port, and some memory space for its exclusive use. As a consequence of this, we have further divided the Control Unit into three modules:

– Channel Information Table (TCYP), which will keep information about the 30 communication channels: their availability (free / in use), telephone numbers of the users, and any other relevant information.
– The Buffer Space (TB), is a table with 30 blocks of memory, one for each communication channel.
– Tables Manager Module (GT). It must maintain the coherence in the system at any time, by dynamically assigning or releasing the resources. It is the only module which can access the TCYP, and therefore, the most critical module in the AIS.

The Mobile Interface Manager Unit is made up of three kinds of modules:

– Communication Request Manager (S255). This module is permanently waiting for a new user to arrive at the Internet side of AIS. When this happens, it informs the GT about the event, and the service the user has demanded.
– Service Dependent TCP/IP Thread (HT). These threads are the ones that implement the functionalities required by the communication service demanded by the user. There will always exist one of them for each communication taking place at the system.
– TCP/IP Threads Manager (GHT) will dynamically create and destroy the HT threads.

The ISDN Interface Manager Unit is similar. The names we have given to its modules are listed next:

– Incoming Calls Manager (GLLE).
– Software Dependent ISDN Threads (HR).
– ISDN Threads Manager (GHR).

Fig. 4 shows the full splitting of the software, and the relations between the modules in it.

We have chosen C++ to implement the system. The reason is that we needed a language with the following characteristics ([3]):

– multithread programming: in AIS several tasks must run simultaneously, and each module in the architecture will be implemented with its own thread.

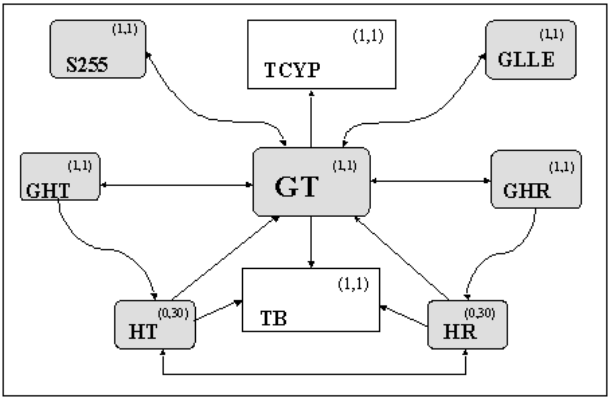

Fig. 4. Full AIS software architecture

- object oriented programming: so each module can be implemented as a different object with its own variables and methods.
- messages communication: the modules interchange messages in order to cooperate asynchronously.

3.3 Implementing DTS-Nokia Communicator Communication

As an example of the utility of AIS, we briefly discuss how it has been used to make it possible the communication between DTS and Nokia Communicator terminals. In order to implement it, we have developed two instances of HT and HR, called HTO and HRO, respectively.

An HTO thread simply receives characters from the socket that keeps it in communication with a mobile telephone. Next, the HTO writes this characters in the space of the TB assigned to the conversation. At the same time the HTO is checking the TB for the availability of characters received from the DTS by the HRO.

The HROs are more complex in the sense that they must implement routines to demodulate the characters received from the DTS terminals, and modulate the characters the Nokia Communicator has sent to the DTS.

Finally we want to remark that we have already added a new service to AIS which allows the sending of synthesized messages written by deaf people (using Nokia Communicator) to non deaf people (using standard telephones). The messages are synthesized in AIS by HRC, one of the two new kinds of threads which had to be designed and implemented in order to create this service (also operative at present).

4 SUMMA061 Development

4.1 Hardware Design

SUMMA061 is installed in the Call Center PC's and only needs a standard modem to be connected to the ISDN network.

4.2 Software Design

Software design of SUMMA061 is simpler than AIS since this dll will only manage one call. Even so, different concurrent threads will be necessary to handle the user interface and modem issues simultaneously.

Fig. 5 shows principle threads involved in the application:

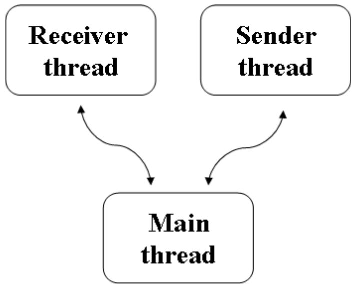

Fig. 5. Simplified SUMMA061 software architecture

- Main thread is in charge of coordinating user actions, user interface appearance (see Fig. 6) and information received or sent to receiver and sender threads.
- Receiver and sender threads communicate with the modem and controls receive and send buffers.

These last two threads are kept asleep, saving memory and CPU resources, until any event is produced in the modem or by the main thread. Communication between those treads and the main one is implemented by means of messages as it is in the AIS model.

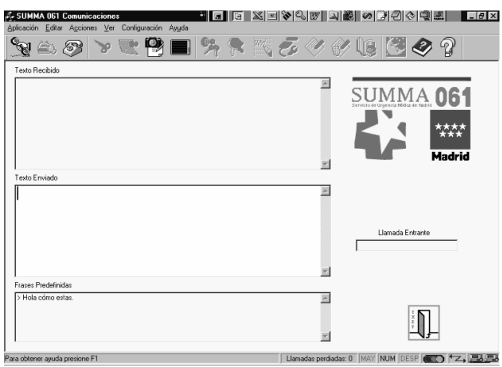

Fig. 6. SUMMA061 appearance

5 Conclusions and Future Work

The systems we have presented have already been tested and have shown to be very robust in operation. We have used AIS for providing DTS- Nokia Communicator communication, and the possibility that deaf people send synthesized messages to non deaf ones. The addition of this second service has shown that our objective of creating a platform for hosting communication services for deaf people has been successfully reached. SUMMA061 is currenty being used in the Medical Emergency Call Center of Madrid.

Due to the characteristics of the design that have been introduced, AIS and SUMMA061 are good frameworks for developing new communication services by creating the adequate modules, and maybe defining some new messages or events. No further changes would be needed.

We expect that in a short period of time deaf people enjoy new possibilities of communication, finding them useful. Furthermore, we are working on getting a complete communication between deaf and non deaf people in a future. However, difficult tasks involving speech recognition and synthesis (among others) must be faced in the way.

References

1. J.A. Cerrada, M. Collado. "Introduccin a la Ingeniera Software". UNED, 1995.
2. GlobalCall ISDN Technology Users Guide for Unix and Windows NT, Dialogic, 1998.
3. MSDN Library-October 1998 Release. Microsoft Corporation. 1995-1998.
4. Jernimo Arenas-Garca "Sistema de comunicacin en modo texto entre telfonos mviles Nokia 9110 y telfonos fijos EDT para personas sordas". (PFC). Madrid, 2000.
5. Survey of Text Telephones and Relay Services in Europe , EUR 14242 EN, CEC, Luxembourg 1992, ISBN 92-826-4395-6.