

# Simulation using OMNet++ of the RAMPE system - an Interactive Auditive Machine helping blinds in Public Transports

J. Sayah, G. Baudoin, O. Venard, B. El Hassan

**Abstract** — this paper presents the simulation of the RAMPE system that assists and helps the blinds in public transportation. First the RAMPE system and the network protocols used in this system are described along with their limitations and possible enhancements. Then we describe the simulation of RAMPE using the network simulator OMNET++ useful for debugging, illustration purposes or performance evaluation and providing a collection of tools that help modifying or implementing new features in an easy fashion.

**Keywords** — Access Point, Association, Blind, Database PDA, Protocol, RAMPE, Station, WiFi

## I. INTRODUCTION

USING the bus, the metro, the tramway or any kind of public transportation, be aware and in time, of urgent notifications and messages, is difficult for normal people, so how is the case with blinds?

RAMPE [1,5,6] initially stands for “Référentiel d'assistance aux personnes Aveugles pour leur Mobilité dans les transports publics et les Pôles d'Echanges” and that can be translated to Reference frame of Assistance to the Blind people for their Mobility in public transport. The project started in January 2004. It is supported by the PREDIT3 which is a French Program of Research, in the field of land Transport, under the general theme of services for mobility and accessibility for persons with reduced mobility. It is conducted by three complementary partners: ESIEE- a center for higher education and research in information and communication technology, LUMIPLAN- a company designer of information products and services in transports, and LEI- an education and research laboratory of the university PARIS 5 in the field of ergonomics and computer sciences.

RAMPE has a single goal in mind: Helping and assisting blind People in using public transportation in an adaptable and practical way. The object is to conceive and develop, thanks to the use of new communication and information technologies, an active and interactive device of assistance and information to the passengers,

particularly intended for the blind people. It should be installed in bus (or tramway) stops.

This article is organized into 5 sections: section 1 is an introduction followed by, in section 2, a description of the RAMPE system. Protocols used in RAMPE are enumerated in section 3. Section 4 shows the simulation using OMNET++. Conclusion and perspective are presented at the end of this paper in section 5.

## II. RAMPE DESCRIPTION AND PRINCIPLE

The RAMPE system (Figure 1) is based on:

- A smart handheld device carried by the user. This device is a WiFi enabled PDA running the RAMPE application software. The Man Machine Interface is realized by Text to Speech Synthesis and the keys of the PDA.
- Fixed base-stations installed at the bus stops. They include a WiFi Access Point and a loudspeaker that can be remotely activated by the user. A client/server of information is also located in the station and connected to the access point WiFi.. The stations are connected to the available transport information system.



Fig. 1. RAMPE Architecture and components

### A. How do these components work together?

The principle is as follows:

The user has an electronic device, the PDA. On the PDA, the RAMPE application is installed and supervises the presence of transportation stations near it. The bus stations have an information system ready to communicate wirelessly with the user's PDA. Actually, each station is equipped with an Access Point (AP). This AP periodically sends its identifier or SSID (Service Set Identifier) whose syntax is specific to RAMPE. This syntax is the

J. Sayah is with Group ESIEE, 2, Blaise Pascal Bvd, BP 99, 93162, Noisy le Grand CEDEX (e-mail: jelsayah@etudiant.univ-mlv.fr; jinanesayah@hotmail.com)

G. Baudoin is with Group ESIEE, 2, Blaise Pascal Bvd, BP99, 93162 Noisy le Grand CEDEX (e-mail: g.baudoin@esiee.fr)

O. Venard is with Group ESIEE, 2, Blaise Pascal Bvd, BP 99, 93162, Noisy le Grand CEDEX (e-mail: ovenard@esiee.fr)

B. El Hassan is with the Lebanese University, Tripoli, Lebanon (e-mail: elhassan@ul.edu.lb)

“stopPointName” separated from the “direction” by a slash (RAMPEstopPointName/direction). The user device notifies this last when he arrives near one or more stations.

If the user wishes to move towards a precise station, the device gives him the possibility to ask this station to ring so he can locate its position in an auditive way. With this intention, it is necessary initially that the PDA associates with the access point WiFi and gets an IP address [2]. The station automatically rings a first time. Then the user can ask the station to ring a second or a third time by double-clicking a button of the PDA. Then the data base from the server of the station is automatically downloaded into the PDA. This database is structured using a XML framework specified either by a DTD (Data Type Definition) or by a XSD (XML Schema Definition).

Then and by the means of his PDA, the user can obtain detailed vocal information (served lines, routes, timetables) of the station which has just made a sound.

This information can change (updating of the database) or an urgent message can happen (accident, unforeseen disturbance, delay, arrival of the bus...). In this case, specific frames have to be sent by the AP to all the PDAs that are connected to it. In the case of an urgent message, the PDA receives the frame, stops navigating in the user interface and informs immediately of the urgent message using the text to speech synthesis. In the case of a the database updating, the PDA receives the corresponding frame, downloads the database again, and informs the user (by a short clearly audible sound) of the new information available.

### B. RAMPE Frames:

The frames are currently defined according to three categories:

- Refresh frames (R): these are always sent by the AP towards the PDA. This frame is produced in case of variation of schedules or exceptional event on a line. These changes are reflected in the base XML which the PDA must download further to a frame R.
- Frames of indication of coming vehicles (V) - it is an urgent message, always sent from the AP towards one or more PDA. This frame is produced when a bus is coming towards the stop.
- Identification frames (U): these frames are always sent from the PDA towards the AP. It makes it possible for the PDA to identify itself to the AP so he can be able to associate with it, then asking it to ring...etc.

### C. Navigation in the RAMPE interface

Once the database is downloaded, the user can navigate into the RAMPE interface with some “zooming” capabilities. This navigation was segmented into 3 levels:

1. Level 1: Inventory of the lines served by this station.
2. Level 2: Inventory of the Main stops (skeleton) or the stop stations.
3. Level 3: Inventory of all the stations that the bus

will pass by, starting at the main station and ending at the terminus of the line.

The blind can pass from level to level by using the buttons of his PDA.

Actually the PDA will have some “Silence” buttons, some “Acknowledgment” buttons and a “Stop” button as per the figure 2.

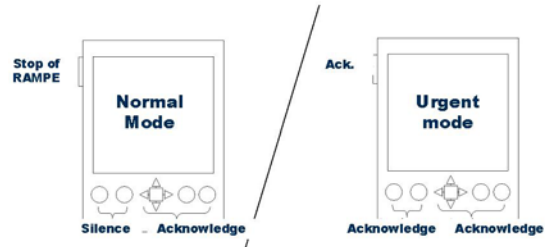


Fig. 2. PDA buttons and functions in RAMPE

We can see that in Normal mode, each button has a specific function. While in Urgent mode, all the buttons will let the user acknowledge. This will help the blind, when hearing an urgent message, not to be lost which button he has to press.

The “Acknowledgement” button is used to go into the next level or to acknowledge an asynchronous message. . The “Silence” button is used to put the application in pause : it becomes silent except if an urgent message is received. For the time being, and while in Level 3, the interface doesn’t evaluate at all. Either acknowledgment or silence, the user will get back to Level 2, then to Level 1 and can restart from this point. At a later stage, an acknowledgment sends the user back to Level 1.

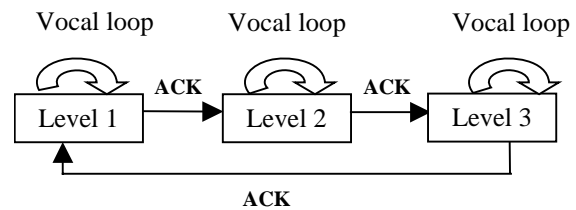


Fig. 3. Levels of the RAMPE application

## III. RAMPE NETWORK PROTOCOLS

RAMPE uses TCP, UDP, HTTP and DHCP protocols.

- TCP is a protocol of the transport layer, generally used above IP. It’s a connection oriented protocol used for reliable transmission of data. In RAMPE, once an AP has been successfully selected, the application proposes a guiding service using a TCP frame.
- UDP: Compared to TCP, UDP is a connectionless oriented protocol. It is used when sending short messages, not requiring an acknowledgment from the receiver. In RAMPE, urgent messages will be broadcasted using UDP.
- HTTP: This protocol is based on TCP via port 80.. In RAMPE, the XML database is downloaded using a HTTP-GET service. From this point, the user is able to browse through the details of the chosen station.

- DHCP: this protocol is based on UDP. It permits the dynamic assignment of IP address to a terminal, in our case to the PDA. When the PDA chooses an AP to associate with, it will obtain an IP address, permitting it to ring the AP and interacts with it.

#### IV. RAMPE SIMULATION USING OMNET++

OMNeT++ (Objective Modular Network Testbed in C++) [3,4] is a discrete Event Simulator. It is used by many research groups for communication network performance estimation.

Models and system can be simulated and shown in a graphical and interactive manner.

The utility of simulating the RAMPE application in OMNET++ is that this network simulator is useful for illustration, debugging, evaluation and performance improvement purposes. It provides a collection of tools that help modifying or implementing new features in an easy fashion. Actually in this article, we are simulating the system at the user interface and not on the network level.

##### A. OMNET++ Models

An OMNeT++ model consists of the following parts:

- NED (Network Description): it shows the components of the simulated network in a graphical manner. These are files with the .ned suffix.
- Message definitions: files with .msg suffix.
- Simple modules implementations and other C++ code, in .cc files (or .cpp, on Windows)
- Configuration files indicating the network to be simulated and the value of some variables or parameters.

In the following, we will be describing the above files used in our simulation of the RAMPE system.

NED description: In our simulation, the NED file shows actually the blind communicating directly with his PDA where the RAMPE application is installed. The PDA then communicates with the access point located in the bus station using a wireless connection. The AP can access the database server as shown in figure 4.

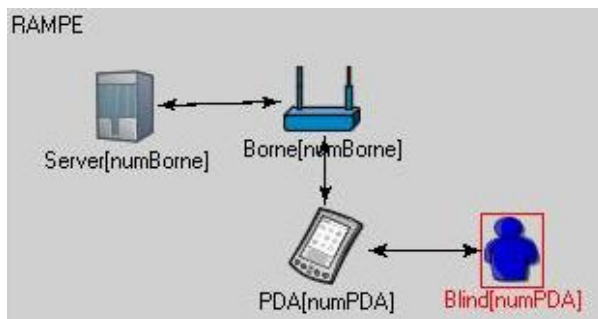


Fig. 4. RAMPE Architecture in NED

The PDA is a compound module with multiple simple modules which are its buttons that are used by the blind:

An “Acknowledgement” button, a “Silence” button and a “Stop” button (fig. 5). As described above, these buttons

permit the blind to evaluate in the application especially when consulting the database and the available information.

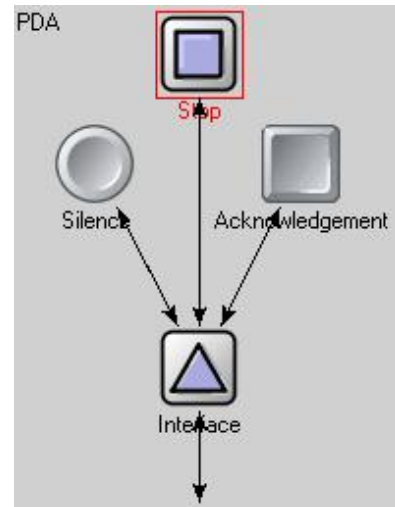


Fig. 5. PDA buttons in NED

It is possible to simulate the case of many Stations and blinds (Fig. 6)

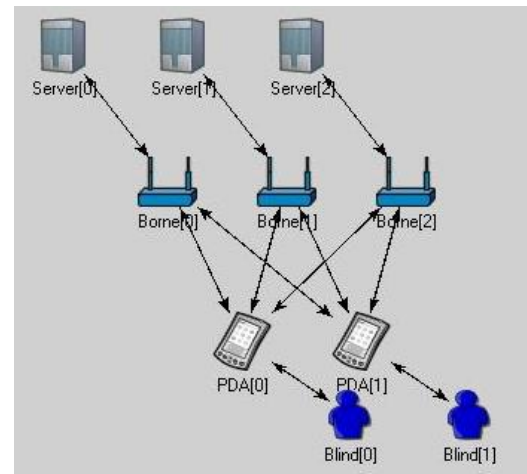


Fig. 6. Case of several stop points and blinds

##### B. Message files

The message file show the different message types exchanged between all the components. There is an association message sent between the PDA and the Access Point when trying to associate and when getting the IP address, a Database message exchanged between the AP and the database server when downloading the database, and the urgent message sent by the server to the AP and then to the PDA when an urgent event happens.

##### C. C++ files

In the C++ file, we define all the simple modules used and we simulate the working of the RAMPE project: First, the number of the access points (i.e the bus stations) will be determined using the input function that asks interactively the user to enter the desired number of stations he wants to simulate. By default this number is 3. Also the number of blinds (i.e the number of PDAs) is to be determined by the user. By default this number is 2.

At startup, the Access points send continuously their SSID in beacon frames. The RAMPE application is up and running on each PDA. The interface is red. Messages are sent back and forth between the different components to simulate RAMPE in an animated way. As explained before, once the PDA is associated with a certain access point, it downloads the database and the blind can begin surfing into it to get more detailed information.

In Level 1 main served lines are enumerated (and the number of minutes before the next bus). IOnce the blind acknowledges by clicking during the enunciation of a line name, he will get into Level 2 that enumerates the main stop points of the chosen line (skeleton). If the user does not acknowledge the application stays in a vocal loop repeating the information. Once in Level 2 and he acknowledges by clicking on a station name, he gets into Level 3 enumerating all the stations served by this selected station. Once in Level 3, if he acknowledges, he gets to Level 1. If he doesn't, he stays in the vocal loop of Level 3. Note that the blind can at anytime press the silence button of his PDA, making silent the RAMPE application except if an urgent event happens and an urgent message is being announced. In this case, whatever was the pressed button, the message is sent by the RAMPE application to the blind and the database must be redownloaded to reflect the changes, and the whole process restarts.

Figures 7, 8 and 9 show some screen shots of the simulation:

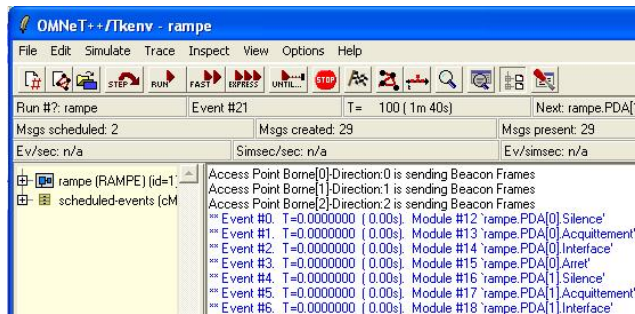


Fig. 7. progression of OMNeT++ simulation

```

Application 0 receives the Beacon Frames
Application 0 is asking Blind 0 if he wants to connect to station Borne[0]-Direction:0
*** Event #11. T=0.0000000 ( 0.00s). Module #18 `rampe.PDA[1].Interface'
Application 1 receives the Beacon Frames
Application 1 is asking Blind 1 if he wants to connect to station Borne[0]-Direction:0
*** Event #12. T=0.0000000 ( 0.00s). Module #14 `rampe.PDA[0].Interface'
*** Event #13. T=0.0000000 ( 0.00s). Module #18 `rampe.PDA[1].Interface'
*** Event #14. T=0.0000000 ( 0.00s). Module #14 `rampe.PDA[0].Interface'
*** Event #15. T=0.0000000 ( 0.00s). Module #18 `rampe.PDA[1].Interface'
*** Event #16. T=0.0000000 ( 0.00s). Module #10 `rampe.Blind[0]'
Blind 0 pressed the Acknowledge button
*** Event #17. T=0.0000000 ( 0.00s). Module #11 `rampe.Blind[1]'
Blind 1 pressed the Silence button
*** Event #18. T= 5 ( 5.00s). Module #10 `rampe.Blind[0]'
*** Event #19. T= 5 ( 5.00s). Module #11 `rampe.Blind[1]'
*** Event #20. T= 100 ( 1m 40s). Module #14 `rampe.PDA[0].Interface'
Blind 0 acknowledged the connection to stationBorne[0]-Direction:0
PDA 0 sent an Association Request to the stationBorne[0]-Direction:0

```

Fig. 8. Logging of the RAMPE simulation

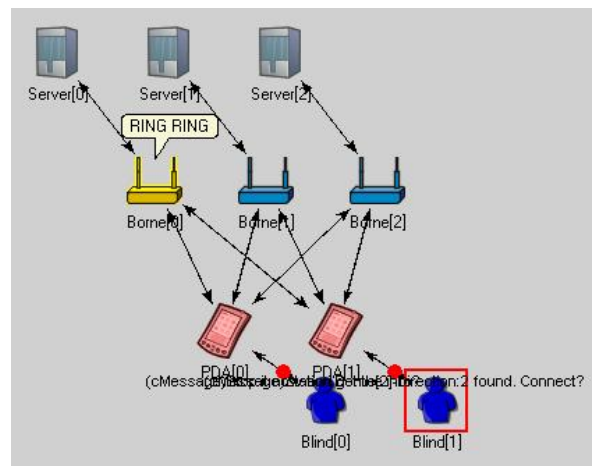


Fig. 9. One stop point is ringing and messages are exchanged between components

## V. CONCLUSION AND PERSPECTIVE

In this paper, we introduce the RAMPE project and we simulate it in an easy and friendly manner using OMNeT++.

The limitation of the current version of the RAMPE system is in using a communication protocol between the access point and the PDA:

We will assume that an urgent message happened at a certain stop point (delay in arrival, accident, arrival of the bus...). The blind must be urgently informed of this type of messages. Actually all the blinds in the bus station must be informed. Some problems may arise:

If using UDP (the connectionless protocol) which doesn't require any acknowledgement how can we be sure that the urgent message reaches all the blinds in the station?!!

And if using TCP (the connection oriented protocol) the connection may suffer from a latency and delay, especially if we are considering the case where we have a concentration of blinds (a school for example) and they want to get out simultaneously taking the bus?

The goal in the future is to find out a protocol more reliable than UDP but at the same time more fast than TCP. This constitutes an interesting continuation of this article.

## REFERENCES

- [1] ESIEE, Lumiplan, Université René Descartes, "Projet RAMPE", final report PREDIT 3, June 2005.
- [2] IEEE Std 802.11-1999, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications, 1999
- [3] Andras Varga. OMNeT++ Discrete Event Simulation System. User Manual. March 29<sup>th</sup>, 2005.
- [4] Jeroen Idserda. TCP/IP modelling in OMNeT++. July 6<sup>th</sup>, 2004
- [5] <http://www.esiee.fr/~rampe/>
- [6] G. Baudoin, O. Venard, et al, "The RAMPE Project: Interactive, Auditive Information System for the Mobility of Blind People in Public Transports", Proc. Of the 5<sup>th</sup> international conference on ITS Telecommunications ITST 2005, Brest France, June 2005.