AXARM: An Extensible Remote Assistance and Monitoring Tool for ND Telerehabilitation

Antonio Bueno[1] , Jose L. Marzo[1], and Xavier Vallejo[1]

[1] Institute of Informatics and Applications, University of Girona, Girona, Spain
antonio.bueno@udg.edu, joseluis.marzo@udg.edu, xvallejo@atc.udg.edu

Abstract. AXARM is a multimedia tool for rehabilitation specialists that allow remote assistance and monitoring of patients activities. This tool is the evolution of the work done in 2005-06 between the BCDS research group of UdG and the Multiple Sclerosis Foundation (FEM in Spanish) in Girona under the TRiEM project. Multiple Sclerosis (MS) is a neurodegenerative disease (ND) that can provoke significant exhaustion in patients even just by going to the medical centre for rehabilitation or regular checking visits. The tool presented in this paper allows the medical staff to remotely carry on patient consults and activities from their home, minimizing the displacements to medical consulting. AXARM has a hybrid P2P architecture and consists essentially of a cross-platform videoconference system, with audio/video recording capabilities. The system can easily be extended to include new capabilities like, among others, asynchronous activities whose result can later be analyzed by the medical personnel.

Keywords: Cross-platform, hybrid P2P, XMPP, MS, telerehabilitation, plug-in.

1 Introduction

This paper presents an application to assist medical personnel in telerehabilitation tasks using basic broadband Internet connections and readily available hardware. It is the continuation of the work carried out in the TRiEM project, a joint effort of the BCDS research group of UdG [1] and the Multiple Sclerosis Foundation (FEM in Spanish and Catalan) in Girona during 2005-06 to develop a tool for MS telerehabilitation.

1.1 The Illness

Multiple sclerosis (MS) is a chronic, inflammatory disease that affects the central nervous system. MS can cause a variety of symptoms, including changes in sensation,

[1] This work was supported in part by the MCYT (Spanish Science and Technology Ministry) research project TEC2006-03883/TCM and the Generalitat of Catalonia's research support program (SGR 00296).

visual problems, muscle weakness, depression, difficulties with coordination and speech, severe fatigue, cognitive impairment, problems with balance, overheating, and pain. MS will cause impaired mobility and disability in more severe cases [2]. The name "multiple sclerosis" refers to the multiple scars (or scleroses) on the myelin sheath (the fatty layer that surrounds and protects neurons, helping them carry electrical signals). This scarring causes symptoms which vary widely depending upon which signals are interrupted.

The prevalence of MS is significant [3]; in equatorial areas it is around 0.0001% while in southern Europe the rate is 6-14 times higher and in northern Europe it is 30-80 times higher. Numbers in North America are on the same range as in Europe.

1.2 TRiEM project

The prototype [4] developed in TRiEM (Catalan acronym for "telerehabilitation and MS") allowed medical specialists to carry out patient consults remotely, with the double objective of making a better use of the funds in order to provide longer and more continuous follow-ups and improving the patients' quality of life.

The system was also designed, at the outset, to use as far as possible standard available low-cost infrastructure; both in terms of computer equipment (CPU, graphic cards, webcams) or communications. For example the prototype can sustain good enough two-way audio and video quality with a basic broadband connection, which in Spain is currently of 1 Mpbs downstream and 300 kbps upstream.

The tool was built on top of an open source instant messaging application [5] and started as a standard videoconference application. In order to adapt it as far as possible to specialist's necessities and patient's characteristics, additional capabilities of audio/video recording and multimedia management were also included. The prototype used a hybrid P2P architecture (see figure 1 in section 2): a server-centric component for session management, information and remote control messages and a peer-to-peer component for the more bandwidth consuming live audio and video streams.

The application was written in Java and made extensive use of the XMPP open standard [6]. This protocol transmits lightweight, human-readable and easily understandable messages in XML form and it was easily extended for the additional features of the TRiEM prototype. For the multimedia part the Java Media Framework (JMF, [7]) and the Media4j framework [8] were used (the video is sent in H.263 format over RTP). These components allow isolating the application from any specific operating system interface, ensuring that the same application can work in Windows and almost any UNIX-like system.

Although in the end the tool worked as expected, several problems remained unsolved or solved just partially. For one, maintenance of the patient's computer ended up consuming a significant amount of time and effort. Added to the complexity of the installation and initial configuration, widespread and unassisted deployment of the tool was compromised. Also, while the tool not only covered the initially desired features but minor additions suggested during the testing period, the addition of new features was at least difficult, and in some cases impossible. These issues could not be addressed within the original project's framework and eventually became the motivation for the work presented in this paper.

As for related work, the field of telerehabilitation has generated abundant literature. For example, already in 2000, Lauderdale and Winters [9] explored the possibility of using then current teleconferencing products for telerehabilitation. In contrast, the American Occupational Therapy Association expressed in 2005 [10] concern about the lack of published research regarding the use of telerehabilitation methods for follow-up of specific occupational therapy services. Nowadays it is still difficult to find telerehabilitation tools that work on more than just one aspect of the rehabilitation process.

The paper is structured as follows: After this introduction, the development and status of the tool is discussed in Section 2 and section 3 is dedicated to the user interface. The paper concludes with the customary conclusions and future work.

2 AXARM: beyond TRiEM

As mentioned before, adding new features to the original prototype was difficult. Taking into account numerous ideas that arose during the original project it was decided that to achieve the full potential of what was becoming a platform, drastic measures were required. Taking as a starting point the basic plug-in features of the instant messaging application that was used as a base for the tool, we proceeded to a complete refactoring [11] of the application and the development of a full plug-in system.

2.1 Refactoring

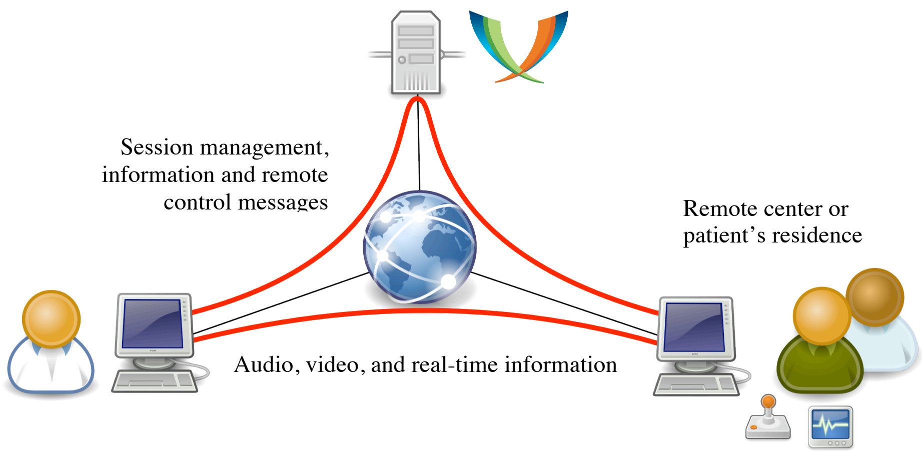

Fig. 1 TRiEM/AXARM common hybrid P2P architecture

The idea of refactoring is to modify the software to make it more amenable to changes without losing or adding new features. As such, a user may not be initially aware of any change between the original tool and the refactored one. Because of that AXARM shares with TRiEM its hybrid architecture (see figure 1 above): The XMPP server allows sessions and passes messages between the parties, while using direct (P2P) connections for more bandwidth intensive data streams (like two-way audio and video streaming).

The refactoring process is progressive: once familiarised with the code it is necessary to identify which parts are to be part of the core of the application and which ones will end in a plug-in. Once a feature is part of the core, it is available for all current and future plug-ins. In the end we were able to considerably simplify the development of plug-ins and their access to the application core. As of this writing all of the original prototype's features have been converted to plug-ins (frequently two of them: one version for the patient and another for the medical specialist) and we are working on adding new ones. In the meantime the application has been completely functional with every refactoring improvement and some new plug-ins have been developed. In parallel the application core has been also improved to offer some additional technical features to make plug-in development easier, and the installation and auto-configuration have been simplified as much as possible.

2.2 Plugins

Essentially a plug-in is a file with the extension "jar" that includes the code to run and any additional resources that are necessary (for example, images and sounds). Once a plug-in is part of the system (installing it consists just in putting the file in the application's "plugins" folder) it will be initiated by the application (see figure 2 below). When the plug-in is running it has control of an area of the graphical interface (GUI) as well as access to any XMPP message the application receives (see figure 3 for an example of a message). It is the responsibility of the plug-in to react to those messages it considers its own. This broadcasting behaviour allows different plug-ins to react to general application messages. On the other hand, plug-ins can also use the application core to easily send messages of their own to their counterpart in the application running in another computer.

At first it may seem the plug-in is limited to what the application offers but in fact it can even add its own preference panel to the application "preferences" section to allow easy access to the plug-in parameters. At the same time programming a plug-in results in a much more simple process thanks to the available features via the application core, and only a limited knowledge of the complete application architecture is needed, so the learning curve is quite smooth.

Frequently the plug-ins must be developed in pairs to express the differences between the actions and interfaces available to each role (patient or medical staff). Once a plug-in is ready for deployment, it appears in the general plug-in repository (they can be filtered according to user role) and it can be installed in the application. In fact one of the first steps to implement the plug-in system was to put together our own plug-in repository as the original one (the public repository of the base instant messaging client) is not available anymore.

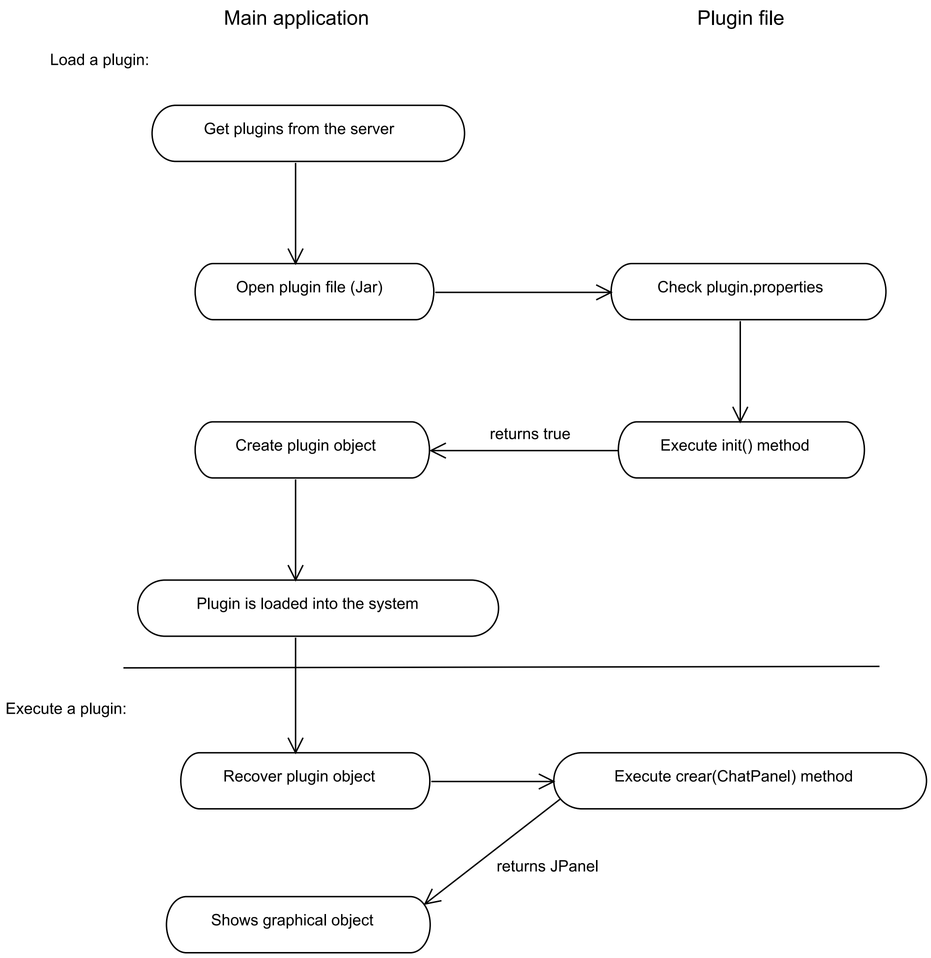

Fig. 2 Interaction between the application core and a plug-in

Currently the AXARM application has plug-ins for almost all of the original features of the TRiEM prototype: *chat*, *recordable videoconference*, *multimedia manager* (to manage A/V recordings), and *notepad* (a simple document viewer/editor). It also has plug-ins that implement new features: *photos* (higher-resolution frame grabbing), *joystick game* (a simple game of motion and coordination control where the specialist sets a difficulty level and see what the patient does together with the standard A/V stream) and *offline recording* (where the patient and the specialist can record themselves in order to transfer the recording at a later time).

Both the *joystick game* (figure 3 is an example of its messages) and the *offline recording* plug-ins represent two completely new fields of activities for the application. It is in our plans to explore the use of game control devices to implement new activities that can be carried on remotely. The rehabilitation specialists are especially interested in the use of *dancing pads* for lower-extremities exercises and the possibilities of the Bluetooth-enabled control devices of the Nintendo Wii gaming platform are exciting. The second field of activities is the development of asynchronous activities that the specialist can assign to a patient, the patient can do at the time of their convenience, and the results arrive to the specialist automatically.

```
<message id="zBrbf-7" to="lab1@bcds.udg.edu"
from="lab2@bcds.udg.edu/JBother">
       <xsow xmlns="joysticksession">
              <xdest>32</xdest>
              <ydest>104</ydest>
              <xcursor>123</xcursor>
              <ycursor>250</ycursor>
              <counter>11</counter>
              <squares>0</squares>
              <time>27</time>
              <button>0</button>
              <type>1</type>
              <difflevel>1</difflevel>
              <positions>1</positions>
              <ip>172.26.0.10</ip>
              <reset>0</reset>
       </xsow>
</message>
```

Fig. 3 Example of a XMPP message of the “Patient joystick” plug-in

3 The User Interface

When the application starts the user is presented with the familiar image of a contact roster. The aspect of reusing user previous knowledge and user-friendliness has been important since the beginning of the project. The difference is when a communication window is opened. Instead of just a chat dialog the user is presented with several panels that can be arranged at their convenience either in a single container or in separate windows.

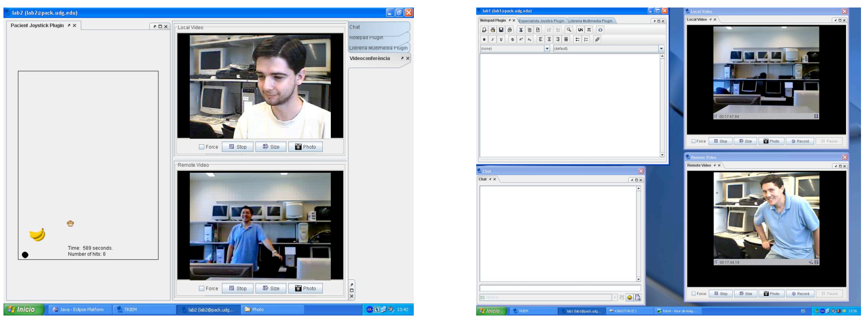

Fig. 4 Graphical interface: personalized panel arrangement

In figure 4 both video panels (local and remote) can be seen with their respective controls, but once deployed the patient's interface is simplified to the maximum while all the controls remain available to the rehabilitation specialist. The patient side of the *joystick game* plug-in can also be seen on the left of figure 4. Figure 5 shows the specialist side of the *joystick game* plug-in when defining the activity parameters and when the patient is carrying on the game.

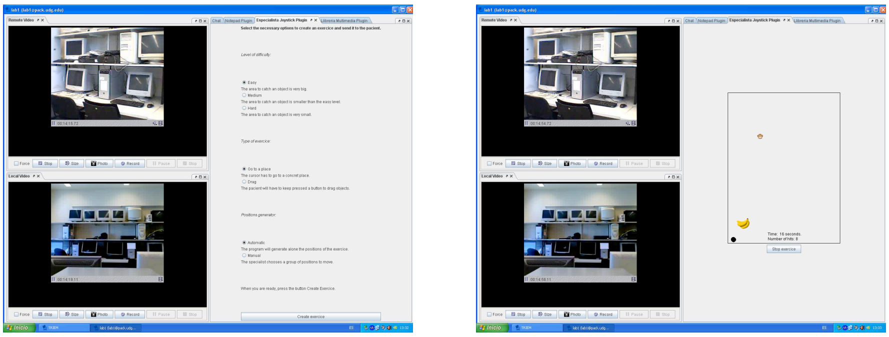

Fig. 5 Graphical interface: *joystick game* for the rehabilitation specialist

4 Conclusions and Future Work

This paper has presented the AXARM application, an extensible tool for remote assistance and monitoring for rehabilitation. A high-level description of the application development as well as specific technical details have been presented, with particular attention to the tool's extensibility and its GUI.

In the short term we plan on developing plug-ins for asynchronous activities, centralise and simplify user management (moving these functions from the application to the XMPP server). We are currently working on the core features to support the asynchronous activities plug-ins. We will also start small scale (10 patients) testing with real patients and specialists.

The incorporation of gaming control devices will follow as well as middle scale testing (up to a hundred patients, possibly involving more than one rehabilitation center) early in 2009, after processing the feedback from the first batch of tests.

As long-term objectives (still without date) we aim to extend the test to several rehabilitation centres along Catalonia. We also want to explore the possibility of packing the system in a standalone appliance for cases where no computer is available or where proper maintenance of the equipment is problematic. For now in these cases we can only install the application in a dedicated Linux partition where we keep administrative rights to ourselves.

Acknowledgments. The authors thank to the FEM and in particular all the staff at the *Hospital de Dia* of Girona for their ideas and unwavering collaboration.

References

1. Broadband Communications and Distributed Systems (BCDS) research group at the University of Girona (UdG), http://bcds.udg.edu/
2. “What is Multiple Sclerosis?”, http://dailystrength.org/news/Multiple-Sclerosis-MS
3. Blake, D.J., Bodine, C.: An overview of assistive technology for persons with multiple sclerosis, Journal of Rehabilitation Research and Development, vol. 39 Number 2, March/April 2002, p. 299-312
4. Bueno, A., Guadall, C., Marzo, J.L., Harle, D.: "TRiEM: An application for MS TeleRehabilitation. Proceedings of the First International Conference of E-medical Systems. October 2007
5. JBother, a Java Jabber Client, http://www.jbother.org/
6. XMPP Standards Foundation, http://www.xmpp.org/about/summary.shtml
7. Java SE Desktop Technologies, http://java.sun.com/javase/technologies/desktop/media/jmf/
8. Media4j, A framework to simplify the use of JMF, https://media4j.dev.java.net/
9. Lauderdale, D.E., and Winters, J.M.: Evaluation of nine IP teleconferencing products (2000). Proceedings of the RESNA 2000 Annual Conference: Technology for the New Millennium.
10. Wakeford, L., Wittman, P.P., White, M.W., Schmeler, M.R.: Telerehabilitation position paper, Am J Occup Ther 2005 Nov-Dec; 59(6): 656-60.
11. “What is Refactoring?”, http://c2.com/cgi/wiki?WhatIsRefactoring