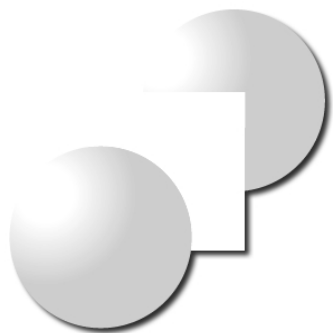# ICERI 2008

PROCEEDINGS CD

International Conference of Education, Research and Innovation

17th-19th November 2008, Madrid

CONFERENCE PROCEEDINGS

ICERI2008 Proceedings
Edited by
L. Gómez Chova, D. Martí Belenguer, I. Candel Torres
International Association of Technology, Education and Development
IATED, Valencia

Book cover designed by
J. L. Bernat Tomás

Technical Program

Open Search

ORAL SESSIONS MONDAY

Session: Education: Enhancing learning and the undergraduate experience (I)
Session chair: Margaret Farrell. Dublin Institute of Technology (IRELAND)

COMBINING COOPERATIVE AND COMPETITIVE LEARNING TO FACILITATE KNOWLEDGE TRANSFER BETWEEN PEERS
M. Arevalillo-Herráez, J. M. Claver

DEVELOPING SKILLS IN THE CLASSROOM: A SIMPLE GAME TO DESIGN A MANUFACTURING PROCESS
M. Jimenez, A. Romaniello, L.M. Arroyo

SUPPORT FOR LEARNING: AN INNOVATIVE APPROACH IN AN AUSTRALIAN TEACHER PREPARATION PROGRAM
K. McCluskey

THE ANALYSIS OF ASSESSMENT AND LEARNING STRATEGIES FOR A LOGISTICS UNDERGRADUATE EDUCATION PROGRAMME
M. Farrell

STUDENT AND TUTOR PERCEPTIONS OF COMPUTER ASSISTED LEARNING IN CLINICAL AND HEALTH STUDIES
P. Fletcher-Cook

USING COMPETITIVE STRATEGIES TO ENCOURAGE AN EVEN PARTICIPATION OF GROUP MEMBERS IN COLLABORATIVE TASKS
M. Arevalillo-Herráez

Session: New Challenges in the Higher Education Area (I)
Session chair: Bill Byers. University of Ulster (UNITED KINGDOM)

"PROFESIONAL.ES - EMPRENDEDOR.ES." DISCOVERY OF MOTIVATION, SKILLS AND THE VALUE OF KNOWLEDGE LOOKING AT THE PROFESSIONAL FUTURE. A COLLECTIVE EXPERIENCE WITH THE STUDENTS OF FIRST YEAR OF CHEMISTRY
A. Domingo, A. M. Bajo, A. Chiloeches, V. García

THE EUROPEAN CHEMISTRY THEMATIC NETWORK (ECTN), COOPERATING TO DEVELOP A KNOWLEDGE BASED ECONOMY
B. Byers

HIGHER ENVIRONMENTAL EDUCATION: A RESPONSE TO THE CHALLENGES OF THE 21ST CENTURY
M. Novo, M.A. Murga, M. Melendro, M.J. Bautista-Cerro

COMPETITIVE INTELLIGENCE NETWORKS FOR HIGHER EDUCATION
A. López-Borrull, E. Ortoll--Espinet

INTERNATIONAL COLLABORATION STUDIES: SUCCESSES AND CONSTRAINTS
D. Ojinnaka

KOREAN EXCHANGE STUDENTS' INTERACTIONS WITH OTHER STUDENTS IN SWEDEN
S. Ahn

Session: Curriculum Design (I)
Session chair: Radu Eugen Breaz. Lucian Blaga University of Sibiu (ROMANIA)

INCORPORATION OF A STRATEGIC FRAMEWORK, INTO A DESIGN OF AN ACADEMIC CURRICULUM PROGRAM: THE IES-UDEM CASE
J. Sillero, F. González Aleu

MARKET-, SKILLS- AND COMPETENCES-BASED CURRICULUM DEVELOPMENT
A. Pausits, K. Prager

A NEW METHOD FOR CONCEPTION AND OPTIMIZATION OF THE CURRICULAR AREA FOR THE INDUSTRIAL ENGINEERING DOMAIN BASED UPON FUZZY LOGIC TECHNIQUES
W. Beck, P. Schupp, O. Bologa, R. Breaz, F. Ionescou

COMMUNITY ORGANIZATION IN INDIA: INNOVATIONS IN PEDAGOGY AND TOWARDS TRANSDISCIPLINARY PERSPECTIVE
J. Andharia, M. Vyas

RESPONDING TO THE DEMANDS OF A CONTEMPORARY LABOUR MARKET; THE RELEVANCE OF CURRICULUM DESIGN AND ASSESSMENT IN HIGHER EDUCATION
A. Morgan, L. Long, E. Lloyd-Parkes

ESTABLISHING THE OPTIMAL RATIO BETWEEN THE CATEGORIES OF DISCIPLINES FOR ENGINEERING EDUCATION IN MULTIDISCIPLINARY SPECIALIZATIONS
P. Schupp, W. Beck, O. Bologa, R.E. Breaz, G. Racz

Session: Quality Assurance and Accreditation (I)
Session chair: Albert Tsang. The Hong Kong Polytechnic University (HONG KONG)

THE INTEGRATION OF QUALITY ASSURANCE IN EDUCATIONAL PROGRAMS. A UNIVERSITY COLLEGE'S SOLUTIONS TO DIVERGING NEEDS OF DIFFERENT STAKEHOLDERS IN EDUCATION
F. van Dijk

UTILIZING TECHNOLOGY FOR ACCREDITATION: ALIGNING PROGRAMS WITH A CONCEPTUAL FRAMEWORK, NATIONAL AND PROFESSIONAL ORGANIZATION STANDARDS
K. Hyatt, W. Barone, J. Kush, R. Mautino

SUB-DEGREE EDUCATION IN HONG KONG: SUCCESSES AND SHORTCOMINGS
A. Dubey

QUALITY ISSUES IN HIGHER EDUCATION: ASSESSMENT AND ACCREDITATION IN INDIA
G. Choudhary

SASSO. A SYSTEM TO ASSESS AND SCORE A TRAINEES ACHIEVEMENT. METHOD AND POSSIBILITIES
L. Simoens

PRODUCT IMPROVEMENT PRIORITIES EVALUATION FOR TECHNOLOGY MANAGEMENT
S.F. Wong, H.C. Tsang

Session: Global Issues: Education, Research and Globalization (I)
Session chair: Michael Tully. Dublin Institute of Technology (IRELAND)

RESEARCH METHODOLOGIES: FROM ACADEME TO COMMUNITY OUTREACH
E. Flordelis, E. Yabut, T. Branzuela, G. Suliguin, P. Katigbak

FINAL PROJECTS IN MECHANICAL ENGINEERING PROGRAMS IN CO-OPERATION WITH INDUSTRY AND INTERNATIONAL UNIVERSITIES -SOME EXPERIENCES AND REFLECTIONS
S. Khoshaba, F. Rovira-Más

INTERNATIONAL EXCHANGE OF INSTRUCTORS FOR ENGINEERING PROJECTS SUPERVISION
F. Rovira-Más, S. Khoshaba

HOW IRELAND HAS INCREASED THE RELEVANCE OF MATHEMATICS AND SCIENCE ON SECOND LEVEL, THIRD LEVEL AND STATE TRAINING PROGRAMMES THROUGH THE USE OF THE INFINITY PROJECT©
M. Tully

POSTER SESSIONS TUESDAY

Session: Poster Session1. E-learning and Virtual Innovations

ADAPTIVE ASSESSMENT IN B-LEARNING: AN EXPERIENCE IN THE ACTUARIAL SCIENCES DEGREE
M.C. Mayorga-Toledano

AN ON-LINE WEBLAB EXPERIENCE IN CONTROL AND ELECTRONICS
L. Castedo, L. Dávila, R. González, S. López, D. Rodríguez-Losada, P. San Segundo

APPLETS JAVA TO CONTRIBUTE TO THE SELF-LEARNING OF STATISTICS AND STATISTIC SAMPLING IN MANAGEMENT AND PUBLIC ADMINISTRATION FOR UNDERGRADUATES
C. Roldan, N. Ruiz Fuentes, M.V. Alba

ARTFUL THINKING IN SECOND LIFE
A. Lim, G. Ma, S. Coelho

ASSESSMENT OF STUDY GUIDES OF NURSING PROGRAMMES PRESENTING IN DISTANCE EDUCATION
U. Chikte

CHANGES OF LEARNING STYLES IN E-LEARNING COURSE AT FACULTY OF SPORTS STUDIES
M. Sebera, L. Berankova, M. Zvonar

DEVELOPMENT OF A WEB-BASED SURVEY SYSTEM FOR SOCIAL SCIENCE RESEARCH INCORPORATING EXPERIMENTAL DESIGN CONSTRUCTS
V. Abraham

DEVELOPMENT OF VIRTUAL CLASSROOM FOR VOCATIONAL TRAINING OF HEALTHCARE PROFESSIONALS – WWW.MEDITOPEU.COM
B. Pazar, SZ. Gergely, L. Csaky

E- LEARNING IN NURSE TRAINING IN SOUTH AFRICA
U. Chikte

EDUCATIONAL APPLICATIONS OF HUMOS: AN INTERACTIVE ON-LINE SIMULATOR OF HUMAN MORTALITY
A. Fernandez-Morales

EFFECT OF INDUCTIVE AND DEDUCTIVE TEACHING WITH DIFFERENCE PRACTICE FORMATS IN WEB-BASED INSTRUCTION ON PHYSICS UPON CRITICAL THINKING OF MATTAYOM SUKSA IV STUDENTS
T. Sranamkam

LEARNING CHEMICAL ENGINEERING WITH MOODLE WORKSHOPS AND OTHER VIRTUAL TOOLS TO ENHANCE CONTINUOUS COLLABORATIVE WORK: A PRACTICAL EXPERIENCE
C. Gomez-Lahoz, F. Garcia-Herruzo, A. García Rubio, JM. Rodriguez Maroto, C. Vereda Alonso

LEARNING CHEMICAL ENGINEERING WITH MOODLE WORKSHOPS AND OTHER VIRTUAL TOOLS TO ENHANCE CONTINUOUS COLLABORATIVE WORK: INTERNAL EVALUATION OF THE REULTS
C. Gomez-Lahoz, F. Garcia Herruzo, A. Garcia Rubio, JM. Rodriguez Maroto, C. Vereda Alonso

OPEN SOURCE LEARNING CONTENT MANAGEMENT SYSTEM: PATHWAY TO DIGITAL EDUCATION DEVELOPMENT IN NIGERIA
I. Ivongbe, E. Nnebe, A. Akhator

REMOTE LABORATORY FOR MOBILE ROBOTICS TRAINING
A. Sánchez, L. Payá, O. Reinoso, A. Gil, D. Úbeda, M. Juliá

THE EMERGENCE OF THE SENSE OF PRESENCE AND ENJOYMENT IN AN E-LEARNING ENVIRONMENT: THEIR BENEFITS ON PERCEIVED LEARNING AMONG STUDENTS
S. Agut, R. Peris, A. Grandío

THE EXPERIENCE OF SOCIAL PRESENCE AS AN INSTRUMENT TO ENHANCE LEARNING PROCESSES IN E-LEARNING ENVIRONMENTS
S. Agut, A. Grandío, R. Peris

SECOND LIFE AND EDUCATION: POSSIBILITIES AND LIMITS OF PROPRIETARY VIRTUAL WORLDS
J. Oceja

TEACHING MIDDLE-AGED STUDENTS: A NEW WAY TO LEARN MATH BY USING AN EDUCATIONAL SOFTWARE
R. Costa, A. Thereza, R. Gomide, E. Damasceno, S. Souza

THE CLASS BLOG : A NEW OPPORTUNITY FOR CHILDREN WITH LEARNING DISABILITIES
H. Vilas-Boas, A. Ramos

THE EFFECT OF THREE TYPES OF VISUALIZATION ON A CHEMISTRY MULTIMEDIA APPLICATION FOR 12TH GRADE STUDENTS
G. Korakakis, E. Pavlatou, J. Palyvos, N. Spyrellis

USING ICT FOR THE LEARNING AND DISSEMINATION OF THE PORTUGUESE LANGUAGE AND CULTURE ABROAD
H. Freitas, A. Ramos

REMOTE LABORATORY FOR MOBILE ROBOTICS TRAINING

L. Payá, A. Sánchez, O. Reinoso, D. Úbeda, M. Juliá
Departamento de Ingeniería de Sistemas Industriales.
Miguel Hernández University. Avda. de la Universidad, s/n
03202 Elche (Alicante), Spain
lpaya@umh.es, alex@emotica.net, o.reinoso@umh.es

Abstract

This work presents a distributed platform to carry out practical lessons with mobile robots through Internet. The main goal consists in the development of an architecture that allows the student to access in a transparent way to a set of mobile robots, independently of the operative system the student uses and with a minimum additional installation in his computer. This paper details all the processes used for the implementation of the system. Also, we present the model of practices proposed so that the students can deepen in some important concepts of mobile robotics and computer vision, taking into account the advantages the access through Internet supposes. At last, some safety measures have been taken to guarantee an adequate use of the system.

Keywords
Remote laboratory, Internet, Mobile robots, Distributed platform, Computer vision.

1. INTRODUCTION

Nowadays, Internet has become a powerful tool in the field of education. The students are highly motivated to use the resources available in remote environments through Internet. Besides, thanks to the remote laboratories, the student can get used to working with real equipments, and he can access them from his house, and with a very flexible timetable. This concept of remote laboratory acquires a special importance in view of the change of scenario the education is undergoing in our country, with the adaptation of the education to the European Space of Higher Education. The new system is centred in the learning of the student and tries to foster his autonomy and reduce the time the teacher uses to present the theoretical concepts at the lecture rooms. This is expected to be a less rigid system with regards to the lectures and timetables. The remote laboratories are definitely going to help to the establishment of this new system in the technical studies.

The information and communication technologies have been put into practice recently in the field of control systems teaching very successfully [1], [2]. Remote laboratories are one of the possible applications. They provide a distributed environment that permits the students to access in a remote way through Internet to the equipments available in the laboratory. This work presents a remote laboratory that has been developed so that the students can make use of the mobile robots available in the laboratory through Internet. In the area of robotics, we find several works that present different ways to operate robots in a remote environment. Candelas et al. [3] present a virtual laboratory to carry out robotics practical lessons. This platform is capable to simulate a robot arm and also, students can teleoperate the equivalent real robot. Thamma et al. [4] describe a client/server architecture to remotely control a manipulator robot, making use of Java programming and using TCP and sockets. The users can access the server through any web browser. The work in [5] presents a remote laboratory that allows controlling a Lego mobile robot through a web browser. The student can make the design of several kinds of controllers using the MATLAB/Simulink environment. The objective is to carry out a path planning and to test them on a real robot. Khamis et al. have developed the 'Developer' architecture to teleoperate and control the B21r robot [6]. At last, Siegwart and Saucy present and discuss some applications that use mobile robots through Internet in EPFL, Lausanne [7].

This work presents a distributed platform that allows monitoring and control a team of mobile robots through Internet. With this goal, we have designed a communication protocol between the different members of the system using CORBA (Control Object Request Broker Architecture) as a reference model. The robustness and efficiency of this standard for the implementation of distributed applications has been demonstrated in previous works [8], [9], [10]. Our objective in the present work

is to develop a platform that allows the students to operate the mobile robots in a transparent way, without needing a deep knowledge of the internal architecture of the robot and offering access to different kinds of robots using the same graphical interface.

2. ARCHITECTURE OF THE SYSTEM

The main objective of the work consists in designing and implementing a platform so that the students can test their control algorithms through a simple graphical interface, using high-level languages to write the program and without necessity of a deep knowledge of the architecture of the robot used. With this goal, we have created three different working environments that allow the simultaneous access of three students to carry out the practices. This structure of environments is shown on fig. 1.
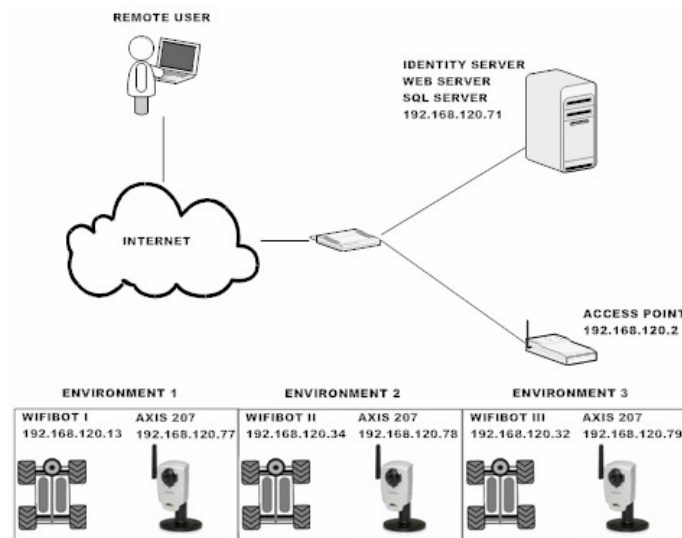

Fig. 1. Working environments that the students can access in a remote way.

Each working environment is composed by a WifiBot robot that has a wireless connection to the access point, and an Axis 207W camera, placed in the ceiling, that provides us with a view of the working environment. Thanks to this camera, the position and the orientation of the robot can be computed using an artificial mark we place on each robot. Each WifiBot robot carries a colour camera on its top, two infrared sensors to measure the distance to the obstacles and the walls, an odometer system and four independent motors. Also, they have been equipped with an onboard PC.

Besides, there is a PC that is connected to the network of the laboratory where the next services run. These services are described in great detail in the next sections.

- Web server. In this server, the client application is available so that the students can download it and install it in their computers.

- Identity server. This is the base of the CORBA communications.

- SQL server. This server is used to implement several safety mechanisms such as the restriction to the access and a record of all the actions each user has carried out with each robot.

A point to emphasize is that to allow the students to access the platform from any location (not only from the computers in the laboratory) it is necessary to have total access to the router to open and redirect the necessary ports. Also, it must be taken into account that not only the necessary ports for the HTTP, SSH and MySQL protocols must be available. Also, the CORBA information will go through the network and it must not be filtered by the firewall [11].

3. FLOW OF COMMUNICATIONS

Before defining how the communications flow between the components of the system, it is necessary to distinguish between the two types of communication that are present in the system.

3.1 CORBA Communications

The basic communication between the different members of the system is based in the CORBA protocol [12]. The main components of the communications network are described in the next paragraphs.

Onboard Servers. Each robot carries a PC onboard where the necessary servers to access to the sensors and actuators of the robot run (video cameras, infrared sensors, odometer, motors, etc.). These servers provide us with several services that are implemented using CORBA interfaces. More specifically, each interface is implemented through a C++ object that access the robot sensors or that controls the robot movements. The CORBA standard allows accessing each service through a text stream known as IOR (Inter-operable Object Reference). Currently, the following interfaces are available:

- <u>RobotBasics Interface</u>. It implements the basic movement actions (to start movement, to set up an advance/steering speed and to stop the robot) that the robot can carry out. It also includes the value of the infrared sensors and the odometer of the robot.

- <u>SingleCamera Interface</u>. It allows the client to obtain snapshots from the camera each robot has on its top. It also allows obtaining images from the camera of the working environment. Thanks to these images, this interface also provides us with information of the position of the robot in the environment thanks to the artificial mark each robot carries on its top.

- <u>StereoCamera Interface</u>. It runs only in the robots that have a stereo pair.

- <u>LaserServer Interface</u>. It offers data from a laser distance server. It is available only in the robots that have this kind of sensor.

Identity Server. It runs in a PC that is connected to the laboratory network. The IP of this PC must be known by all the elements of the system. This is the only piece of information we must configure to establish the communication between all the components. This server has two main functions. The first one consists in keeping a list of all the robots and the cameras of the environment that are active at each moment in the system. This way, this server stores a string that is different for ach robot (Wifi1, Wifi2, etc.) and a list of the services this robot offers (camera, infrared sensors, sonar, etc). Each service is also stored as a text string and an IOR reference. The second function is to carry out an access control from the client applications to the robots. This means that when a student accesses one working environment to carry out the practices, he has the exclusive rights to operate with the robot. None of the other students can access this environment until the current user leaves it. Besides, to avoid that only a student can monopolize the environment during a long period of time, a reservation system has been established. Through it, each student can make a reservation during a time slot in a timetable (the maximum period of time is limited to one hour in the current implementation). During this time slot, only the student who has made the reservation can access the working environment. Taking into account that, in this moment, there are three independent environments available, three different students can access the system simultaneously to test their algorithms.

Client Application. It can run on any computer, independently of its architecture and operative system. This application permits the access to the robot environments to carry out the practical, independently of the location it runs from. The student can access from the computers in the laboratory, but also from any computer that has Internet access.

This client application has been entirely developed using the Java programming language, because it provides the independency we need both from architecture and from operative system. Also, Java supports CORBA in a native way, without necessity of using other foreign libraries. The CORBA support in Java is defined by the OMG (Object Management Group), and it permits the interaction with

the CORBA services developed using the C++ language, both those offered by the robots and those offered by the environment cameras and the identity server.

The implementation of the client application has been carried out through Java Web Start. It is the reference implementation model of the JNLP (Java Networking Launching Protocol) specification, and it has been developed by Sun Microsystems. With this technology, it is possible to start desktop Java applications that reside in a web server, checking first if the version of the application the client has installed is the latest one. If not, it downloads the new version and the application will run in local mode. The starting of such applications can be carried out from a web page or through a link in the client desktop. Thanks to this technology, we can be sure that an application is always distributed with its most up to date version.

The information about the location of the application, the version and other details can be found in a configuration file whose extension is '.jnlp'. This way, it is necessary to put at the disposal of the students the file '.jnlp' in a web page so that they can download it and run it after. Also, any person who wants to make use of the application must install in his computer the last distribution of JRE (Java Runtime Environment). We must take into account that currently, Java Web Start is included in the JRE.

In the first step of the project, we considered the possibility of integrating the application into a Java Applet. This way, the students would only have had to access a web page to use the application. However, the security restrictions that these Applets present, made it complicated working with files in the computer of the user (reading and writing) and carrying out bidirectional communication flows using CORBA with the services the robots provide. That is the reason why Java Web Start has been chosen in the latest version of the platform.

Communications Flow. At first, it is necessary that the Identity Server is running in the network. After this, the onboard servers of the robots must be activated. When one of these servers starts, it publishes the name of the robot and the interfaces it implements (with their corresponding IORs) in the Identity Server.

When a client application wants to access the services a robot offers, it first makes a request to the Identity Server, indicating the symbolic name of the robot and the interfaces it wants to access. If the robot is available at that moment, the Identity Server provides the client application with the references to the interface the robot implements. From that moment, the communication is carried out in a bidirectional way between the client application and the onboard server of the robot. In a typical mode of operation, the client application sends some requests of data capture from the sensors and movement commands to the motors. All this process is sum up, step by step, on fig. 2(a) and 2(b).
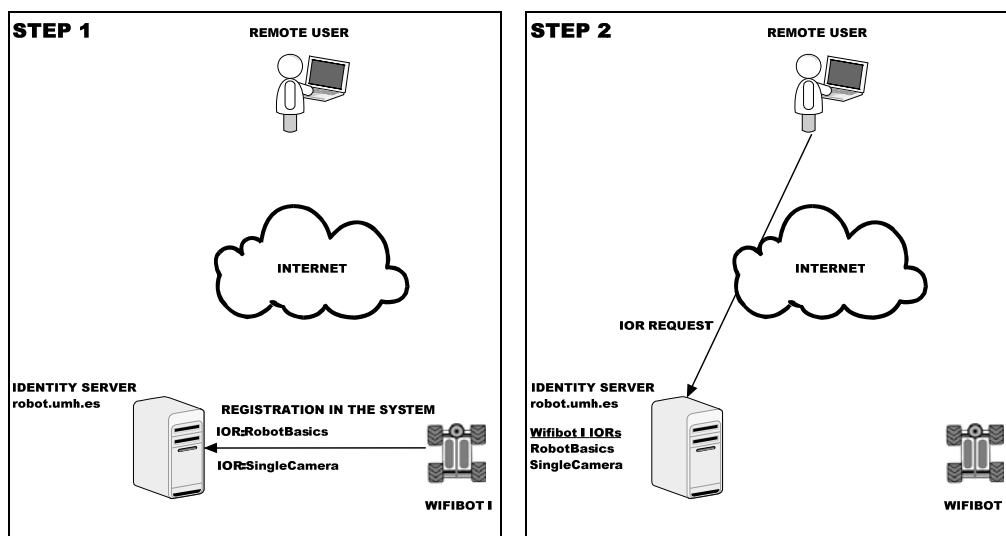

Fig. 2(a). CORBA communications flow in the remote laboratory. Steps 1 and 2.
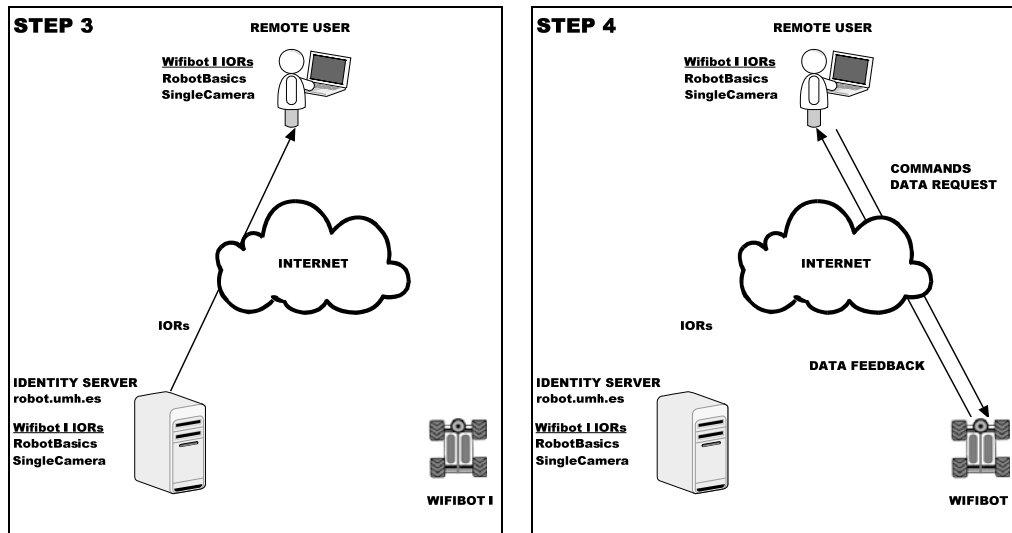
Fig. 2(b). CORBA communications flow in the remote laboratory. Steps 3 and 4.

Fig. 3 shows the appearance of the client application. From the main window of this application, the student can continuously see the images that both the camera of the ceiling and the camera of the robot capture. This window also shows the advance and the steering speeds of the robot. In addition, using the arrow buttons, the student can operate the robot with basic movements. Using the top menus, the user can decide which data he wants to save while the application is running and the name of the files to save these data.
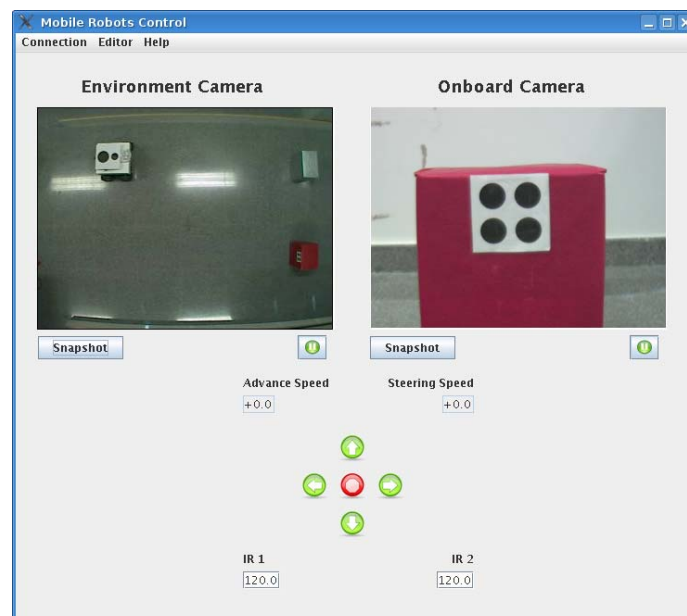

Fig. 3. Appearance of the graphical interface of the client application.

3.2 SSH Communications

This level of communication is used with the objective that the students can send the algorithms they have implemented to control the WifiBots. The main goal is to allow the student to send the program he has created to the robot he is operating, to compile and to run it. The implementation of this communication flow has been made with JSch, a Java library developed by the company JCraft under GPL license, which implements the SSH2 protocol. Thanks to this library, the functionality of the SSH2 protocol can be integrated in our program.

SSH gives support for a reliable remote access, reliable file transfer and reliable TCP/IP readdressing. It can encrypt, authenticate and compress the data transmitted automatically.

In this work, Jsch has been used to establish remote connections with a SSH daemon that has been installed in the robots with the objective of accepting incoming connections. The platform makes use of the following JSch functions:

- Scpto: This function is used to send the file that contains the source code from the user's computer to the computer in the laboratory where it will be compiled and run.

- Exec: It permits running commands in a remote way. In this platform, it has been used to compile the source code using the command 'makefile' in the remote machine and to run it later. Besides, this function permits showing on the user's screen the standard output ('stdout') and the standard error output ('stderr'). If the standard error does not activate, it means that the compilation has been carried out successfully. Otherwise, the student should correct the source code and send it again.
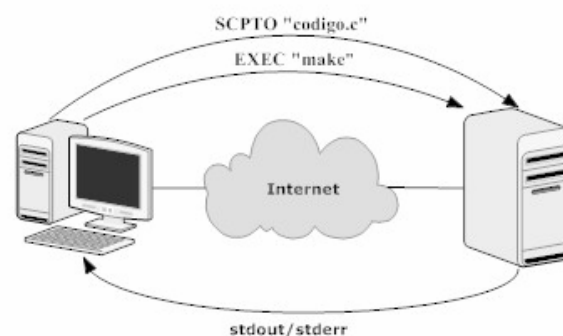

Fig. 4. SSH Communications flow in the remote laboratory.

The advantage of this method to make the practical is that the student does not need to install any library in his computer. However, the depuration of the possible errors during the compilation of the source code becomes a slow process, because the source code must be sent several times until all the possible errors have been corrected. This communication process is shown on fig. 4. Fig. 5 shows the appearance of the window where the student can edit the source code, send it to the server, compile it and check if there has been any compilation error. This edition window is accessible from the main window of the client application.

Fig. 5. Appearance of the Editor window. The student can edit the program and the text box at the bottom shows the status of the process.

4. PRACTICAL SESSIONS PROPOSED

Up to this moment, a practical lesson with two exercises has been implemented. Through these exercises, the student can put into practice his knowledge about the sensors and the actuators of the robot. Also, they deepen in the knowledge of the robot kinematics and the reactive control of mobile robots using a competitive coordination of behaviours. Next, the objectives of both exercises are described.

4.1 Exercise 1

The student must create a program so that the robot moves forward, describing a straight line. The robot must avoid the obstacles it finds in its way, and it must stop when it has travelled a distance of three meters. The task has to be solved making use of the infrared sensors to detect the presence of an obstacle, and the odometer data must be continuously read to compute the position of the robot and the distance it has travelled. Besides, the student must use the information of the ceiling camera to know the position of the robot at each moment and compare it with the results computed using the odometer data.

In general, the robot has to move straight, and it has to describe a 90º arc when it fins an obstacle. The robot must turn to the side where the obstacle is farthest from, and it must stop when the distance it has travelled is higher than three meters. To compute this distance, the student has to make use of a function that integrates the odometer data and that returns the coordinates of the robot with respect to the initial position. This function is available in the RobotBasics interface.

The problem must be solved using a competitive coordination of two behaviours, 'Go straight' and 'Turn'.

After the simulation, the student has to make a graphical representation of the path the robot has followed, using the odometer data. After this, the student must draw the same trajectory, but computed from the information provided by the external camera. A comparison and a critical analysis of the results must be carried out.

4.2 Exercise 2

The student must make the necessary changes in the program of the exercise one so that the robot moves from its initial position to a target point, trying to avoid the obstacles it finds. In the target point, there is an artificial mark that can be viewed from the ceiling camera.

As in the previous exercise, the robot must move straight to the target point, turning when it finds an obstacle. Once the obstacle has been surpassed, the robot must orientate again to the target point and move straight to reach it. The current position of the robot must be computed using the function that makes uses of the information of the ceiling camera, and the same for the target point.

The problem has to be solved through the implementation of two behaviours: 'Go to destination' and 'Avoid obstacle'. At each moment, only one of these behaviours must be active, depending on the reading of the infrared sensors (presence of an obstacle). Once the program has finished its running, the student must draw the path the robot has followed, using the data of the ceiling camera.

4.3 Safety measures

The fist measure that has been implemented to preserve the application from unknown users is a SQL database (MySQL) that runs in the same computer where the Web server and the Identity server run. This database has two purposes. First, it restricts the access to the application to those users that have been previously included in a database, through a login and a password. This verification is carried out before checking which robots are available in that moment. Second, this database also registers all the actions the users have carried out in the system, from the verification of the student until he leaves the system and the environment is released.

To develop the system, a MySQL server must be installed in any computer of the intranet. Two tables are implemented. The first one contains three fields with the data of the user: name, identity number

and the password, which is codified using SHA1 format. The second table stores the record of actions that every user has carried out while they have been using the system. This table stores the date, time, IP and a code to identify the action. Taking into account that the client application has to write data on these tables, it has been necessary to use the MySQL connector to Java ('MySQL Connector/J') to implement the application.

Apart from this safety measure, another prevention mechanism has been implemented to avoid malfunction of the robots. To do it, an independent thread has been programmed. This thread runs concurrently with the user program and it is in charge of monitoring the advance and steering speeds of the robot and the infrared sensors. In case that any speed is over a threshold or the robot has a wall or obstacle very near, this thread can stop the robot and reinitialize the algorithm.

5. CONCLUSION

To conclude, we must emphasize the fact that we have implemented a platform that permits that the students carry out the practical lessons in mobile robotics, in an absolutely free timetable and from their own home. With this platform, the students can test the algorithms they have programmed, over real robots, and they can access all the services these robots offer. Thank to this platform, the students can deepen in the knowledge of basic sensors in robotics and basic control strategies.

Several servers have been implemented so that the students can operate the robots in a transparent way. In spite of the fact that all the services presented in this paper have been implemented in a specific type of robot (WifiBot), all the interfaces would be the same for any kind of robot. This way, we could add new environments or change one or more robots without necessity of modifying neither the Identity Server nor the Client Application. In conclusion, the platform permits the remote control of a team of heterogeneous robot in a transparent way.

At the moment, there are three different environments available. Each environment is composed by a WifiBot mobile robot and a camera that offers a general view of the environment where the robot can move. Each robot is equipped with a colour camera, two infrared sensors, odometer system and four independent motors. This way, three different students can access the system simultaneously to test their algorithms.

The students have at their disposal the information about the available robots and the services each robot offers. Also, they have several templates that show how they can use the methods implemented (read sensors, capture images or move the robot). The client application can run on any computer, independently of its architecture and operative system, and it has been implemented using the reference model of Java Web Start. This technology permits running Java applications that reside in a web server. Through this application, the students are able to connect to the robot, move it, monitor the state of the sensors, upload the program they have developed to control the robot, compile and run it in the remote environment.

Thanks to this platform, the students are able to carry out some practices on mobile robotics in a real environment. They can study robots with different kinematics, try to avoid obstacles using the infrared measures, perform a visual localization or navigation, study the properties of the odometer data and also test algorithms that imply collaboration between robots. Besides they can visualize the evolution of the task thanks to the cameras that are installed along the environment. Thanks to the use of the Java Web Start technology, the client application has the following advantages:

- Independence from the architecture and operative system of the student's computer.

- The user does not need any installation in his computer. Also, as the compilation is carried out in the remote computer, the user does not need any additional library.

- The last version of the client application is always available and continuously updated in the user's computer.

Acknowledgements

References

[1] Sánchez, J., (2001) Un Nuevo enfoque metodológico para la enseñanza a distancia de asignaturas experimentales: análisis, diseño y desarrollo de un laboratorio virtual y remoto para el studio de la automática a través de Internet, Tesis Doctoral, UNED.

[2] Dormido, S., (2004) Control learning: present and future, Annual Reviews in Control, 28(1), pp. 115-136.

[3] Candelas, F. A., Puente, S. R., Torres, F., Ortiz, F. Gil, P., Pomares, J. (2003). A virtual laboratory for teaching robotics. *International Journal of Engineering Education,* vol. 19, No 3. pp. 363-370.

[4] Thamma, R., Huang, L. H., Lou, S. J., Diez, C. R. (2004). Controlling robot through Internet using Java. *Journal of Industrial Technology,* vol. 20, No 3.

[5] Carusi, F., Casini, M., Prattichizzo, D., Vicino A., (2004) Remote control of a Lego mobile robot through the web. Workshop of Internet Based Control Education, Grenoble, France.

[6] Khamis, A., Rivero, D. M., Rodríguez, F., Salichs, M. (2003). Pattern-based architecture for building mobile robotics remote laboratories. *Proceedings of the IEEE International Conference on Robotics and Automation,* vol. 3, pp 3284-3289.

[7] Siegwart, R. and Saucy, P. (1999). Interacting mobile robots on the Web. *Workshop Proceedings of the IEEE International Conference on Robotics and Automation.*

[8] Payá, L., Gil, A., Reinoso, O., Riera, L., Jiménez, L.M. (2006). Distributed platform for the control of the WiFiBot robot through Internet. *ACE 2006. 7th IFAC Symposium on Advances in Control Education.*

[9] Payá, L., Juliá, M., Reinoso, O., Gil, A., Jiménez, L.M. (2006). Behaviour-based multi-robot formations using computer vision. Aceptado para publicación en *6th IASTED International Conference on Visualization, Imaging and Image Processing.*

[10] Utz, H., Stulp, F., Moeld, A. (2004). Sharing belief in teams of heterogeneous robot. *Proc. of RoboCup-2004 symposium,* pp. 508-515.

[11] Theophilo, A., Endler, M., Cerqueira, R., (2005) Evaluation of Three Approaches for CORBA Firewall/NAT Traversal. *International Symposium in Distributed Objects and Applications*, Cyprus.

[12] OMG, Object Management Group, Common Object Request Broker: Architecture and Specification. Revision 2.0, 1995.