NETLAB: A Framework for Remote Network Experiences

Stiubiener,I.; Ruggiero,W.V.; Silveira,R.M.;Korolkovas,I.; Skopp,S.; Meiler,C..

(itana,wilson,regina,ian,skoop,cmeiler,)@larc.usp.br

Abstract— **This work presents some results in the construction of a framework to provide distant network experiences. This framework, called NetLAb, will offer all network experiences of our regular undergraduate course at de Computer Engineering course at Politechnic School of Sao Paulo University in a virtual way to regular students and students of other correlated courses**.

Index Terms— *distance-learning, electronic learning, network experiences, virtual laboratories.*

I. INTRODUCTION

"Learning engineering is practicing it". This basic principle is true for several areas of knowledge and it is the root of the constructivist learning theory. A fundamental activity of the engineering learning process is the possibility of having real life practical experiments, constructing and making operational real life engines. In a regular engineering curriculum special disciplines of laboratory practice are always present. In particular, in the computer network engineering curriculum a special discipline of network laboratory plays an important role in the construction of the network engineer's knowledge. To the engineer apprentice it introduces the real life experience including the network and computer equipments, its techniques and protocols.

One of the biggest challenges in an engineering school is to provide students with enough available laboratories. these laboratories are very expensive to provide and maintain with state of the art equipments and technologies. At the same time, such infrastructure is available only to a few students, under certain restricted circumstances due to security and scheduled constraints.

Other similar projects of remote laboratories focuses on a single network experiments, this paper shows a framework that can be used to create an unlimited number of experiments using a set of hardware and software provided to the teacher [4], [5].

As we can observe in the infrastructure proposed by Braun,Steinemann and Zimmerli , they are not proposing a complete framework but only a procedure to reserve resources and equipments (BRAUN, STEINEMANN ZIMMERLI, 2003]. In the paper presented by Mérida,Jové,Massaguer and Fàbrega they implemented an infrastructure for IP Networks but the topology of the network is fixed differently of our

proposal where each group create and implement its own topology [6].

The objective of this paper is to present the concepts adopted in the construction of this framework to provide remote network experiences and present some results observed with the implementation of a network experience.   This framework will improve our network labs offerings, making their use easier and less expensive and making them available to a larger number of students. In this software and hardware framework we intend to create learning actions to carry out virtual network practical experiments using equipments and resources from several distinct remote laboratories, interconnected by a high-speed network. This framework would integrate this multi laboratory scenario providing in a common view the remote student with the perception of a unified experimental environment.

This framework, called NetLAb, will offer all experiments in our regular network laboratory disciplines in the Computer Engineering course at Polytechnic School of University of Sao Paulo in a virtual way to anyone, anytime, anywhere. Furthermore, we intend to integrate our laboratory resources with the ones available in other places and schools connected by the advanced Internet. The developed framework will be integrated with TIDIA-AE's environment, an e-learning platform that is being developed in São Paulo State in Brazil, and will able to make use of all the tools available through this e-learning environment [1].

In Section II of this paper we will present the NetLab framework architecture we are developing to offer all network experiments in a virtual way, in Section III we present a case-study we developed to observe and validate the configuration of the framework, in Section IV we present some future experiments we intend to virtualized and finally, in Section V we present some conclusions we observed in order to improve NetLAb framework.

II.   FRAMEWORK ARCHITECTURE

The NetLab framework architecture contains the components needed to make a remote experiment possible. The Figure 1 below shows some tools, organized in groups, which are provided to a professor to create an experiment. By using and combining different sets of these tools it is possible to create a wide array of network experiments. The framework also

provides a configurable and extensible group of experiments scripts, called Experiment Templates - ET, which consists of procedures, hardware and software

tools which must be present in an experiment. These ETs enable a professor to create a new experiment based on a previous script created by another professor. The idea is not to restrict the professors in their experiments projects but provide them with an initial base to start from, minimizing the effort required for this task, and allowing enough flexibility to design the experiment, as they desire.
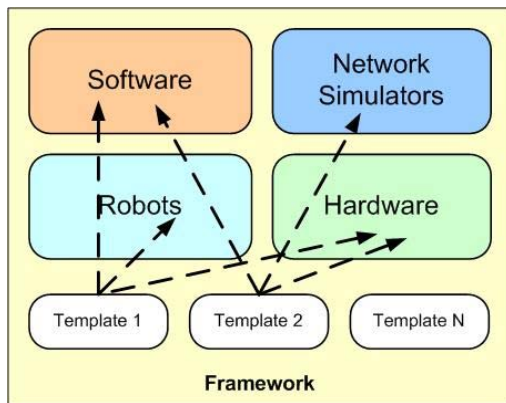

Figure 1 – NetLab Framework

Bellow we present a description of each part that compose the NetLAb framework:

A. Software

The NetLab framework will present a list of possible software tools that can be used by a professor in an experiment. This list will point to a pool of tools that can be developed by our laboratory or obtained from other frameworks and other sources. The framework will be extensible and configurable in a way that any new software tool considered interesting to be offered in an experiment will be easily added to NetLAb framework.

Some of the software functionalities are explained below:

1) Network analyzer

This software can capture frames from a live network connection, or read from a capture file. Live data can be read from Ethernet, FDDI, PPP, Token-Ring and others [3]. The Network analyzer could be used for a wide range of experiments; the student can measure the data contained in the frame headers and payload e.g. sequence numbers, protocol used, packet length, etc.

2) Performance Monitors

It allows monitoring of local as well as remote hosts. The graphical front end uses so-called sensors to retrieve the information it displays. For each type of information, one or more displays are provided. Using these software students can monitor, for example, CPU load, memory usage, network bitrates and packet loss.

3) Traffic Generator

A traffic generator can be used to measure various aspects of networking performance. Its primary focus is on bulk data transfer and request/response performance using either TCP or UDP and the Berkeley Sockets interface [5].

4) Network Management Tool

A SNMP (Simple Network Management Protocol) agent configured in some network hosts and routers can give important information about the network performance, e.g. traffic in/out, number of packages, CPU Load, Disk usage.

Students could retrieve though a SNMP request information about all kind of equipments which supports this management protocol.

B. Network Simulators

As the implementation of an experience can be very time demanding and as we are in a distributed framework, with students far from real equipments, we consider that it is very advisable for students to simulate some parts of the experience before implementing and executing it. For instance, it is important to confirm that a certain topology will converge, that is, that the interconnection protocols will recognize all network elements. Through the use of a simulator, students can make some observations and tests based on a topology previously designed by them before connecting network equipments. If it network presents the expected behavior, they can ask the laboratory technician to physically configure the desired projected network.

The NetLab framework will provide a group of simulator tools that can be used and configured in an experience.

Until now we have in NetLAbs' the simulator group of tools composed by one simulator, the ColLab tool, a network simulator developed at LARC - PCS – EPUSP, which is a collaborative software simulator consisting of an environment developed in Java platform, that allows two different students to create and simulate a network topology and, simultaneously, choose routers, links, protocols and simulate the OSPF behavior [2].
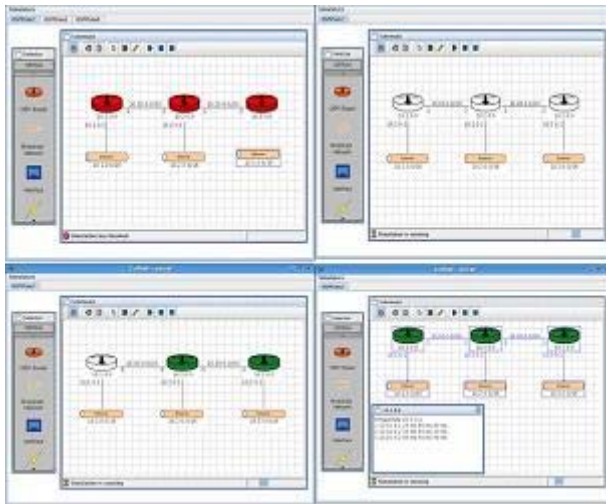
Figure 4 – ColLab Screenshot

C. Robots and Cameras

The students will have a set of tools to capture images from the physical equipments. Cameras can be positioned to have a complete vision of the hack. Also a robot with a video camera attached can be controlled to obtain more details of the cables connections.

Two types of robots can be used, moving robots with video cameras attached and armed robots. The first type of robot can be remote controlled to have a better view of the equipments and cable connections of the experiment and the second type, the armed robots, can enable the students from a control interface to change, for example, a cable connection using its arms.

D. Hardware

To make the experiment real a set of network equipments such as routers, hubs, switches, cables, hosts, etc. are provided

These equipments are connected, according to the experiment demands and can be accessed by a student using an Internet browser.

In Figure 5 its possible to see NetLab topology. The student can access the simulator, the robot video streaming and each host configured. The student connects through the Internet to a Web Server containing the WebGUI (Web Based Graphic User Interface) which consists the web page that students will access to make the laboratory experiments. The WebGUI is composed by the components presented following:
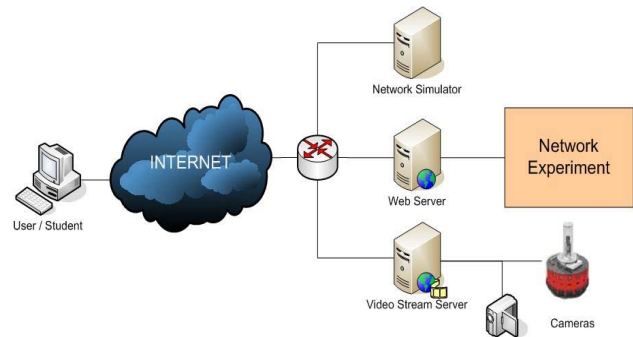

Figure 5 – NetLab Network Topology

A. VNC Client

VNC (Virtual Network Computing) software makes it possible to view and fully-interact with one computer from any other computer or mobile device anywhere on the Internet. VNC software is cross-platform, allowing remote control between different types of computer. For ultimate simplicity, there is even a Java viewer, so that any desktop can be controlled remotely from within a browser without having to install software[http://www.realvnc.com/].

B. Embedded Video Player

The Embedded Video Player allows students to validate and change physical connections of the network experiments. A robot with a camera broadcasts a video stream in real time of the rack, hosts and network equipments.

C. Menu Buttons

Through the menu buttons, students can switch hosts or obtain video stream of the robot. Depending on the number of hosts needed for the experiment, these can be accessed clicking on the respective button.

D. Robot Controller Buttons

At the left side of the WebGUI, there are buttons to control the robot, using this feature, the students can verify the physical connections made by the Laboratory technician or change some connections of the equipments.

Figure 6 – NetLab WebGUI

III. DESCRIPTION OF A CASE STUDY: AN EXPERIMENT IN OBSERVING A TRAFFIC NETWORK LOAD

In this study case we intended to produce a generic learning action for a virtual network experiment, to be offered by a Learning Management System - LMS, in our case TIDIA-Ae LMS [1].

As a study case we developed a network experiment to observe a network behavior under different traffic loads. To implement this experience we developed a first Experiment Template, the ET1, which consists of four parts: network design and planning, network simulation, remote tests in a real environment and report elaboration. The four parts are accessed remotely and can be developed collaboratively by a group of students.

Bellow we present the four phases of our study case:

Phase1: In this first phase students received, each one at his/her site, the script of the experience and all material needed to understand and execute the experience as: texts with theoretical concepts, interesting and correlated links and any other material considered relevant;

Phase2: In phase 2 students projected the experience, to make the measures and observations planned in the experience. To elaborate the network project students had disposable many collaborative tools as chat, e-mail, blackboard and a portfolio, offered by TIDIA-AE's environment platform [1].

With the topology projected they verified the network convergence by simulating it. They executed this simulation in the ColLab [2], shown in Figure 4, with the intent of simulating the network and avoiding students sending the physical lab a topology with problems and errors.

After this simulation students sent to the lab, by e-mail, all connections that should be done to implement their

experience. At the same time, they scheduled a date they connected from their home/site to the lab. When they connected to the lab they found their topology implemented manually by a technician.

Phase 3: Before executing the experience, students first checked if all projected connections were correct. To do this they had the possibility of conducting remotely a robot with a video camera attached and, by sending commands to this robot, they checked all connections and made eventual corrections.

After checking all connections and making sure that the network was correct, students started the experience, causing and provoking some events, for instance, they provoked a load causing a high traffic on the network and observing the network performance, making the necessary measures by using disposable software;

Phase 4: In phase 4 students compiled a report with all observations they have done, all measures they did, relevant conclusions and sent this report by e-mail to the responsible of the experience.

This experiment was presented in TIDIA-Ae Workshop, in November 2005 and will be applied to our network laboratory regular course in the second semester, 2006 observing students' behavior and all the process involved in the experience execution.
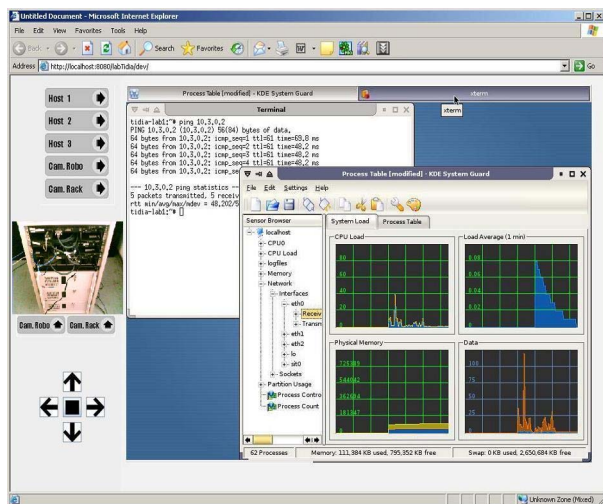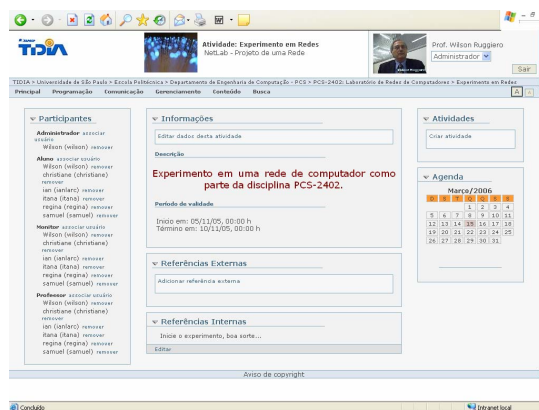

Figure 7.1 – The experiment script provided in TIDIA-Ae´s LMS

Figure 7.2 – The experiment script provided in TIDIA-Ae´s LMS

In the second phase of NetLab project we want to construct a virtual panel (a virtual switch panel) that will do virtually all connections planned by students'.

We intend also to transform all experiences of our regular course in virtual experiences and add one new experience that will use resources of others distant laboratories. Some of these experiments are presented in the next Section.

VI. FUTURE NETWORK EXPERIMENTS

Most of the experiments offered to our undergraduate students can be easily changed to fit NetLab. To do so, a professor needs to provide the needed hardware and software to the students. Below are some examples of experiments and the configuration that can be selected by the professor.

For the experiments below, only a single network experiment topology is necessary to accomplish the exercise, this topology is presented in the figure 8.
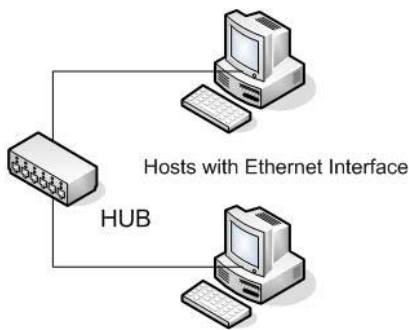

Figure 8 – Network Experiment Topology

Note that all hosts must have two network interfaces in order to the interface network traffic not intervene with the experiment network traffic.

E. Ethernet

This experiment allows students to familiarize with the Ethernet Protocol, based on IEEE 802.3 pattern [7] - and ARP – Address Resolution Protocol . The hardware and software needed are listed below:

Hardware Requirements:
- 2 hosts;
- 1 Hub or Switch;
- 2 Ethernet Adapters.

Software Requirements:
- Terminal Console;
- Network Analyzer (Sniffer).

The student will be able to access two distinct hosts connected by a hub or switch, and using a terminal console the student can send Ethernet Datagrams through the network. Moreover a Network Analyzer e.g. Ethereal can capture these datagrams and show the content of headers and payload. The students will be able to measure header fields values and compare with the IEEE 802.3 standard.

F. Client – Server

The major objective of this experiment is to learn the most important concepts of the client server architecture. For that, a server application using TCP/IP sockets will be implemented.

Hardware Requirements:
- 2 hosts;
- 1 Hub or Switch;
- 2 Ethernet Adapters.

Software Requirements:
- Terminal Console;
- Java SDK;
- Internet Browser

G. Java Server Pages

This experience allow students to make dinamic web pages based on the JSP tecnology.

Hardware Requirements:
- 2 hosts;
- 1 Hub or Switch;
- 2 Ethernet Adapters.

Software Requirements:
- Terminal Console;
- Web Browser;
- Java Tomcat;
- SQL Server;
- Netbeans.

H. Basic Servers

The objective is to learn basic servers configuration. The students will configure a DHCP server, HTTP server, FTP server and file sharing.

Hardware Requirements:
- 2 hosts;
- 1 Hub or Switch;
- 2 Ethernet Adapters.

Software Requirements:
- Terminal Console;
- Web Browser;
- DHCP server;
- Apache Web Server;
- Samba

A. IP Routing

The IP Routing experiment main objective is the use of IPv4 and ICMPv4. These protocols are called routing protocols. They are based on datagrams that contains the destination host on its header.

Hardware Requirements:
- 10 hosts with Ethernet adapters;
- 3 Hub or Switch;
- 4 Routers
- 2 Ethernet Adapters.

This experiment is an example of advanced network topology, as seen, its required a complex set of network equipments.

Software Requirements:
- Terminal Console;
- Network Analyser (Sniffer)

Using these tools, students can send IPv4 and ICMPv4 datagrams through the network. Based on measurements the students can discover the network topology without seen the connections of hosts and network equipments.

V. CONCLUSION

This distributed network framework to provide a virtual laboratory structure will help to enrich the offer of the network experiments, using more adequate equipments, and will provide an enhanced experience facilitating the work that deals with real life situations in a more affordable way.

The learning activity composed by these experiments provides students with a collaborative networks experience in which they will interact to perform data communication network experiments. The objective of the learning activity is to allow the students to develop network experiments remotely and collaboratively access real equipments. In this way the laboratory, that was accessible only to students in presential courses, now can be shared remotely with other students.

The NetLab environment does not take into consideration the student profile in this early implementation, especially because our Learning Management – LMS, the TIDIA – AE environment cannot consider a user profile yet [1]. In next versions of the LMS, when it will be possible to consider a student profile, we intend to consider in our remote experiments student's preferences, as: preference for working remotely or in a present way, preference to work in a group or individually (remote experiments permit a student to work alone), which schedule a student prefer to work,etc.

An activity implemented as a case study of NetLAb Framework is now available in TIDIA-AE´s distance learning platform and will be offered in our regular network laboratory discipline in the Computer Engineering course at Polytechnic School of University of Sao Paulo in a virtual way to anyone, anytime, anywhere [2].

REFERENCES

[1] Tidia-Ae Platform. Accessed on March 10. Disposable at http://tidia-ae.usp.br

[2] Collab . Vilcachagua, O. D.; "Um ambiente colaborativo para simulação de redes de computadores", Dissertação de Mestrado, biblioteca da Escola Politécnica, São Paulo, 2002.

[3] IEEE802.3 Pattern. Accessed on March, 10. Disposable at http://grouper.ieee.org/groups/802/3/

[4] Harms,J.;Berqia,A.; Diop,A,; "**Remote Real Laboratory: Linux Installation and Configuration**", International Conference on Internet Computing- (IC) 2003, Las Vegas, Nevada, USA.

[5] Weyland,A.; Kurt,E.; Braun,T.;Baumgartner,F; "**Virtual Routers: A Tool for Networking Research and Education**"; ACM Computer Communication Review, 33(3):145-151, July 2003.

[6] Fàbrega,L.; Massaguer,J.; Jové,T.; Mérida,D.; "**A Virtual Network Laboratory for Learning IP Networking**"; ACM SIGCSE Bulletin, Volume 34 , Pages: 161 - 164 Issue 3,September 2002.