

Experiments in Communication Protocol for Remote System

Bruna S. F. Pereira (EM), Evilyn S. Lemes (EM), Gustavo P. Moraes (EM), André F. de Angelis (PQ)

Abstract

This project aimed to demonstrate the development of a simple application protocol and its implementation in an object oriented language. It was proposed the case of a distributed client/server system planned to remotely supervise a set of sensors connected to a TCP/IP network. The target protocol was designed with the help of finite automata concepts. Two software modules were written in Java: the client to simulate the sensors in replacement of the actual hardware and the server to monitor several remote sensors in a central locality. The protocol was also implemented in Java and it was proved to work correctly over the Internet.

Key words: *networks, protocols, java programming*

Introduction

Network protocols are communication standards that permit that computers to exchange data. Most of them are designed grouping sets of functions in layers, ending up with an architecture composed by several layers, as the models ISO/OSI and TCP/IP. The layer nearest to the user program itself is the application layer. The communication of this specific layer is managed by an *application protocol*.

This project focused in the design and implementation of an application protocol aimed to communicate remote sensors with a central monitor. Moreover, a client/sever graphical system was written in Java to operate the developed protocol.

The server provides the screen in which the data of all sensors are displayed, whereas the client simulates the actual hardware of the remote sensors.

A distributed application was written using graphical user interface and other common program facilities. The client side simulates an scalar remote configurable sensor (Fig. 2).

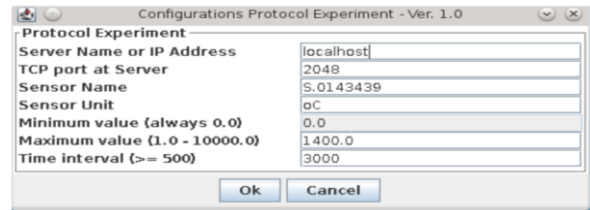


Fig. 2: Client configuration screen.

The server side simultaneously monitors the activity of a set of sensors. Figure 3 presents the server running with a single sensor connected to itself.



Fig. 3: Server monitoring one single sensor.

Results and Discussion

The first result was an application protocol design, expressed by a series of finite automata, as shown in Figure 1.

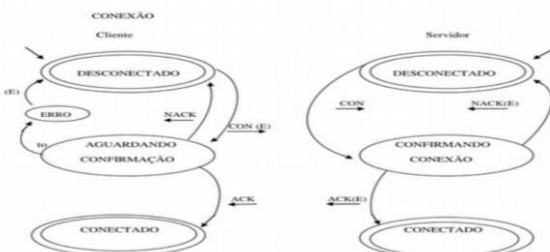


Fig. 1: Finite automata corresponding to the CONNECTION protocol primitive.

Conclusions

All the proposed objectives were reached. The protocol was developed, implemented, and has proven to work correctly.

Acknowledgement

We thanks CNPq and Unicamp for the scholarships; the Concrete Mathematics Laboratory of the School of Technology for the offered resources.