Research on Web-Based Networked Virtual Instrument System

B P Tang, C Xu, Q Y He and D Lu

Department of Mechatronics, College of Mechanical Engineering, Chongqing University, Chongqing, 400030, China

E-mail: baopingtang@hotmail.com

Abstract. The web-based networked virtual instrument (NVI) system is designed by using the object oriented methodology (OOM). The architecture of the NVI system consists of two major parts: client—web server interaction and instrument server—virtual instrument (VI) communication. The web server communicates with the instrument server and the clients connected to it over the Internet, and it handles identifying the user’s name, managing the connection between the user and the instrument server, adding, removing and configuring VI’s information. The instrument server handles setting the parameters of VI, confirming the condition of VI and saving the VI’s condition information into the database. The NVI system is required to be a general-purpose measurement system that is easy to maintain, adapt and extend. Virtual instruments are connected to the instrument server and clients can remotely configure and operate these virtual instruments. An application of The NVI system is given in the end of the paper.

1. Introduction
Among the existing virtual instrument (VI) system, it is very popular to use the single-bus-based VI, but there are not many the networked virtual instrument (NVI) systems with the multi-bus-based VI connected by the internet. The rapid growth, diffusion and reduction in cost of network related resources, e.g. bandwidth and standard protocols, make it possible to develop and deploy the NVI systems [1]. It is very significant to integrate the PXI-based VI, the VXI-based VI, the GPIB-based instrument and the PC-based VI into the NVI system which can accomplish the measuring and monitoring task. The Web-based networked virtual instrument (NVI) system will be designed in this paper by using the object oriented methodology (OOM).

2. General Design of NVI System
The NVI system is required to be a general-purpose measurement system that is easy to maintain, adapt and extend. Virtual instruments are connected to the instrument server and clients can remotely configure and operate these virtual instruments. The NVI system allows multiple clients to operate multiple virtual instruments concurrently subject to the condition that each instrument can only be configured and operated by one client at any given time. However, multiple clients can view data from the same instrument simultaneously. The general architecture of the NVI system, which is shown in figure 1, consists of two major parts: client—the web server interaction and the instrument server—virtual instrument communication [2]. The web server communicates with the instrument server and the clients connected to it over the Internet, and it handles identifying the user’s name, managing the
connection between the user and the instrument server, adding, removing and configuring VI’s information. It validates client identities before allowing a connection to the system and disconnects a client after the operation time is reached or an error in system operation occurs. The virtual instrument parameters and information, processed data, transaction records, and client information are stored at the server side in a database. The database is used to maintain consistency of data and to handle concurrent transactions. Concurrency issues arise because of possible simultaneous requests for the same instrument by more than one user.

The instrument server acts as a mediator between the web server and the virtual instruments, and it handles the server-instrument communication. The responsibilities of the instrument interface includes initializing the selected instruments with the parameter values supplied by the client and retrieving the processed data from the instrument to either store in files or display in graphical form at the client's side. The instrument server can share a master computer with the web server, and also be deployed independently. Thanks to security, it suggests that the instrument server had better be deployed independently.

The implementation of the NVI system utilizes technologies that support OOM as well as communication over the Internet such as JAVA Applets, Servlets, and so on [3]. Database management is implemented using Microsoft Access and Java Database Connectivity (JDBC). The communication between the server and instruments is implemented using VC++.

3. Object oriented analysis of the NVI system [2,4]

The main objects of the NVI system can be identified from the system description introduced in Section 2. The main objects’ classes include the web server class, the client class, the instrument server class, VI class and database class.

The identified objects perform certain tasks based on the messages they get from other objects or from within the same object. Thus, based on the description of the NVI system, the main object responsibilities can be summarized as follows.

Web server class: The web server class is the most important class in the system. Some of the responsibilities that the web server carries include: identifying connected virtual instruments, configuring the virtual instrument information and its parameters, configuring and handling the addition or removal of virtual instruments, sending the list of connected virtual instruments to clients when requests are made, preparing test plans that contains a list of user-selected virtual instruments and the operation time, obtaining the required parameter names, types, sizes, and default values for the
selected virtual instruments, releasing an virtual instrument after the specified time of operation is reached and storing the output data obtained from virtual instruments in an indexed database.

Client class: A client is responsible for connecting to the web server, receiving a list of virtual instruments connected to the server. There are two types of clients, a controller client and a viewer client. A controller client can create a test plan by selecting virtual instruments from the list supplied by the web server and supplying operation time, parameters' values for the selected virtual instruments. A client obtains the processed data from the virtual instruments, displays it in a graphical form or gets the processed data file after the specified operation time for that virtual instrument is reached. A viewer client cannot create a test plan, but can only view the data obtained from a set of virtual instruments according to the test plan decided by a controller client. It is obvious that one client can setup a test plan, but more than one client can observe the data obtained from the virtual instruments included in that test plan.

Instrument server class: The instrument server acts as a messenger between the web server and virtual instruments, its main responsibilities include obtaining the required parameters values supplied by the client from the web server and initializing the virtual instruments accordingly. The instrument server operates the virtual instruments for time interval specified by the client, collects data from the virtual instruments and sends the processed data to the web server.

Virtual instrument class: This is the class that represents the physical instruments in the networked measurement system. It handles information regarding the virtual instruments and the virtual instrument parameters. Virtual instruments have to be programmable and could be connected to the instrument server through any standard interface such as serial, GPIB, or VXI.

Database class: The database is used to store all the information concerning instruments' information and parameters, processed data files generated for clients, client information, and transaction records at the server side.

The next step after identifying classes of the NVI system is to identify the links and association. Any dependency between two or more classes is an association. Links and association can be extracted from the above definition and responsibilities of objects, and can be also extracted from figure 1.

4. Implementation of the NVI system

The implementation of the system requires the integration of object oriented codes, network communication protocols, database management systems, and instrumentation interface software. The NVI system is implemented using an integration of JAVA for code development and network communication support, Microsoft Access for database management and VC++ for instrumentation interface. The implementation of the NVI system can be broken into four modules: system administration, client interaction, database management, and instrument management.

(1) System administration. System administration handles all the operations related to clients and virtual instruments. It is responsible for maintaining records for the currently logged in clients, currently used and available virtual instruments, addition, deletion and updating of the instrument information and client information available at the web server. It also manages the connection and disconnection of clients according to the test plans and the notification of clients via electronic mail when test plans are completed. It can also obtain summary of information pertinent to a specific user such as a listing of the virtual instruments used, the number of access times, and total duration of time of use of each instrument.

(2) Client interaction. A client needs to register with the NVI system to connect to the system and operate virtual instruments. The NVI system processes, validates and stores the information in the database. If the information is invalid, the client is prompted to reenter the information. A registered user connects to the web server by supplying valid security parameters namely the user id and password. Once a user is connected to the NVI system, the web server sends a list of the currently connected virtual instruments. If a virtual instrument is in use, information regarding their status, the client using it, and the time remaining until it is free is also displayed. A user creates the test plan by supplying the web server with desired values of instrument parameters and operation time. The test
plan is then validated and passed on to the virtual instrument through the instrument server. The processed data coming from virtual instruments are stored in the database for future reference by the client. The web server notifies the user when a test plan is completed.

(3) Database management. A relational database management system is utilized in the NVI system to organize and store the system information. The database is used to store information related to user, connected virtual instruments, and processed data. The database is divided into various tables that store this information. The user's information is stored in a table that holds terminal identification of the user's computer, username, and password along with other related details. Another database table where complete information of the instruments is available contains entity integrity on instrument id. The database program automatically generates this unique id when an instrument is added. This table also contains other details of the connected instruments such as company name, interface to which it is connected, brief description of its functionality, number of parameters it requires, and number of outputs it generates. The details of required parameters such as name, type, size, and default value for each instrument is stored in the available-instruments table. The output information is stored in a separate table called instrument-output that has a referential constraint on instrument id column.

(4) Instrument management. The instrument server is the part where communication between the server and the connected instruments takes place. The functionality of the instrument server requires minimal modifications of objects when there is a change in structure while instruments are added, removed, or configured. It starts with the identification of connected instruments and their parameters. It recognizes the instruments selected by the users and starts executing their test plans. It extracts information such as instrument Id, operation time for each instrument, their parameter values, and file to be used for storing processed data generated by instruments from the test plan. It opens a reference to the selected instrument and initializes the instrument with the parameter values supplied by the user after appropriately typecasting them. After initializing the instruments, the instrument server starts operating them for the specified operating time. The processed data from the instrument are collected after the operation is completed and are stored in the files. These data are also sent to the web server to be presented in a pictorial form.

5. Application example
According to the proposed method in this paper, the web server, the instrument server and database are shared in the same computer connected the internet, PCI-based output control card of virtual instrument is wired to actuator, USB-based data acquisition card of virtual instrument is wired to two displacement transducers fixed on the different position in the vibration experiment device, then remote measurement and control of the vibration experiment device can be implemented. The structure of experiment system is shown in figure 2. Virtual instrument interacts with the outer device such as the actuator and the displacement transducers. The remote user accesses the server by the internet, and selects the virtual signal generator and oscillograph to control the vibration experiment device. The experiment result indicates that the NVI system works well.

6. Conclusion
With the development of virtual instrument and network technology, it is possible to deploy the networked virtual instrument system. The web-based networked virtual instrument system is proposed in this paper, and the experiment indicates that the NVI system works well and is easy to maintain, adapt and extend.
Acknowledgments
This work is supported by National Natural Science Foundation of China (No.50405009) and supported by Program for New Century Excellent Talents in University (NCET-04-0849).

References
[1] Franco Cicirelli, Angelo Furfaro and Domenico Grimaldi etc 2006 A software architecture for the management of networked measurement services Computer Standards & Interfaces 28 386
[2] Mohamed Abdelrahman and Abdul Rasheed 2000 A methodology for development of configurable remote access measurement system ISA Transactions 39 441
[3] G Canfora, P Daponte and S Rapuano 2004 Remotely accessible laboratory for electronic measurement teaching Computer Standards & Interfaces 26 489
[4] Mihaela M Albu, Alessandro Ferrero and Florin Mihai 2005 Remote Calibration Using Mobile Multiagent Technology IEEE TRANSACTIONS ON INSTRUMENTATION AND MEASUREMENT 54 24
[5] Jiao huiqin and liu junhua 2001 Web-based remote measurement and control network model Chinese journal of science instrument 22 130
[6] He linsong and Zhang Rong 2002 Web-based virtual instrument technology and its application Chinese mechanical engineering 13 759