Internet-Based Calibration of Measurement Instruments

Yong En Haur
Z. Roziati
Z. Abidin A. Rashid

National Metrology Laboratory, SIRIM Bhd.,
Lot PT 4803, B.B. Salak Tinggi, 43900 Sepang, Malaysia

Faculty of Computer Science & Information Technology,
University of Malaya, 50603 Kuala Lumpur, Malaysia

***Abstract*-In a conventional calibration service, clients send their instruments to a laboratory for routine calibration, during which the clients are not able to use their instruments due to downtime. Some of their instruments are huge and heavy such as power source transformer which is difficult to be transported.**

A solution to these problems is to send an engineer together with measurement instrument to a client site to perform the calibrations. Another way of solving the problem is to send only the measurement instrument to a client site where the instrument and the unit under test are controlled by a computer that links with the laboratory computer over the internet. The calibration engineer can monitor the calibration progress remotely from his laboratory without going to the site. This can improve the calibration service and bring satisfaction to the clients.

The research conducted here is to develop a calibration system that allows the client instruments to be calibrated at the remote location. The existing internet facility and its advancement pave the way for development of internet calibration system. The methodology of internet calibration is based on the technology of client-server applications. It is distinguished by the connected online calibration service that it provides.

Customers from various industries and services will benefit from this implementation. Their instruments performances are directly traceable to national laboratory. Experts from national laboratory can share their skills and knowledge with the customers.

I. INTRODUCTION

A. Calibration Overview

Calibration is part of the metrology activities. Metrology deals with designing measurement, conducting measurement, analyzing measurement results and calculating uncertainty. In a simple sentence, metrology is the science of measurement. The ultimate purpose of metrology is to ensure that the measurement is accurate. Calibration is the comparison between a measurement instrument against a standard measurement device, which has equal or better accuracy [1].

B. Statement of Problems

Calibration is a routine activity. Industries which comply with ISO standards and involve in trading legislation must have their measurement instruments calibrated [2]. All the instruments regularly used by industries and calibration laboratories require re-calibration after a period of time.

Traditionally, instruments have to be transported to a laboratory for calibration; usually involving a significant amount of downtime. The users will not be able to utilize the devices until they are returned. Sometimes users request calibration service from a nearest commercial calibration laboratory to reduce the instruments downtime. It is common that eventually the laboratory can only provide less accurate measurements compared to a national laboratory. Users also have limited communication with the staffs of national laboratory in this conventional calibration service.

C. Proposed Solutions

An alternative way of reducing the user downtime is to transport the standard instrument from national laboratory such as National Metrology Laboratory (NML, SIRIM) to client site where the devices are calibrated. The internet plays a role of a communication medium to address the dissemination of calibration values from NML.

Most of the instruments are equipped with communication interfaces such as RS-232 port, network interface or IEEE-488 interface [3]. When these instruments are connected to the computer, their operations can be controlled by a computer. The software system has to extend its capability over the internet to gather the calibration information from the national laboratory. In other words, the national laboratory provides a server system that contains the calibration information pending for client's computer to connect from a remote area. Upon the connection, client's computer is able to automate the calibration via the instruments communication interfaces. The internet calibration software will perform the automated calibration after the user has done the initial setup such as powering up the instruments and setting up the terminals connections. The calibration data is then sent back to system server for process.

D. *Previous Research Outcome*

National Physical Laboratory (NPL, UK) has taken a lead in internet calibration with two demonstration projects completed in July 2000 [4]. In mid 2001, Dudley and Ridler from NPL have successfully developed internet calibration for automatic network analyzers [5]. They utilized several programming languages such as JAVA, ActiveX and VB script to develop the software which allows two-way communication between an NPL server and a remote laboratory, while maintaining data security, overcoming company firewalls, running at an acceptable speed, and presented significant challenges.

National Institute of Standards and Technology (NIST, USA) has also developed an internet-based calibration for multifunction calibrator in year 2000 [6]. NIST collaborated with Sandia National Laboratories to setup and demonstrate a remote calibration of multifunction calibrators using this internet-based technique, which expands present calibration capabilities and become more interactive. Internet calibration is capable to provide real-time audio, video and data exchange, consultation and training, as well as web-accessible calibration procedures, software and reports.

Netherlands Meetinstituut (NMi, Netherlands) has similar internet calibration development as NPL and NIST [7]. The software is simpler and only concentrates on calibration of Fluke® 5700A calibrator. 5700A has powerful feature such as self-calibration, self-reporting tools, 2-year full verification and internal calibration check to accomplish the calibration task [8].

Steinbeis Transfer Centre for Quality Assurance and Quality Metrology, Germany is a government organization that encourages the use of web-based collaboration in a customer-oriented internet Metrology [9]. It aims to give a big chance especially for small and medium sized enterprises (SME) to shorten their gap in quality management in competition with large enterprises. The quality capability of SME will be improved [10]. Added value is increased comfort, better functionality and affordability through application of information society technologies in SME.

All these conclude that calibration work using internet is expanding rapidly in countries like UK, USA, Germany, Japan, Netherlands and Korea. International workshops on "Internet Measurement and Self Calibration" and "The Impact of IT in Metrology" held in Delft and NPL in year 2002 provided profound discussions on this issue.

II. REQUIREMENTS OF INTERNET-BASED CALIBRATION OF MEASUREMENT INSTRUMENTS

The internet-based calibration of measurement instruments requires IEEE-488 facility to control multiple instruments using a computer. The advantage of Client/server model in distributed system is taken to establish the internet-based calibration system.

A. *IEEE-488 Standard*

Measurement system based on the IEEE-488 standard is controlled by a computer, which provides both process control and data processing [3]. In the common system, the control unit sends to the concrete instrument a command for setting instrument function and measurement range and executing the measurement. The measuring instrument acts on the command, sets its function and range, executes the measurement, and sends to the address of the control unit the measured value. The control unit accepts the string and continues in controlling the process. Each instrument has its own set of instructions, which it "understands" and by means of which it communicates with the control unit [11].

The measurement system can be programmed either in some of higher-order programming languages (C/C++, Pascal, BASIC), or some specialized development environment (making programming of the system easier) can be used [12]. GPIB commands in a program consist of three layers. The inner layer is the device specific command that an instrument needs to have. For example:

```
*RST, *IDN? or SYST:COMM:SERIAL:BAUD 9600
```

If there is a driver for a given instrument at the disposal in the system, the user needs not to know the commands, and only the function of the instrument has to be selected.

B. *Client/Server Application*

Most of the internet-based calibration works using client/server architecture. It is a conventional and scalable architecture, whereby server acts to serve client when there is a request. Sometimes the term server or client may refer to the software rather than the computer.

Client/server application contains built-in synchronization and sharing engines [13]. Client/server also embodies the concepts of user accounts and sharing of resources the system must separate and keep track of different users' files and applications during a user session, then free up those resources for another user session.

The functionality of the application was separated logically into two parts: the processes requiring the majority of the computing horsepower were put on the server, and the user interface and less processor-intensive processes were put on the client. An RPC system runs over a Request-Reply protocol and can be integrated relatively transparently with a programming language [14].

C. *Calibration System*

A more accurate instrument is used to verify the performance of another instrument which is also called unit under test (UUT). Calibration is also carried out to obtain the correction factors for the UUT. In electrical laboratory, the involved physical quantities are voltage, current, frequency, resistance, etc. Digital multimeter (DMM) and multifunction calibrator (MFC) as shown in Figure 3.1 and Figure 3.2 are instruments that provide several functions to measure and stimulus source respectively. DMM and MFC are also called measuring meter and source generator respectively. Hence in a calibration system, one must generate source (output) and the other measure (input) the source. The common functions, ranges, and resolution for a DMM and a MFC are listed in Table I.

Both DMM and MFC have their own indicating values at the displays. For instance, an MFC generates 10.00000 V dc source and a DMM measures the source with 10.00034 V of reading as shown in Fig. 1. This shows that there is an error in the system. If the MFC is the calibration standard (more accurate), this means that the DMM (also a UUT) has reported 0.00034 V of error. By using manufacturer specification, the performance of the DMM can be verified. If the specification states the 10 V dc function is ±0.00100 V, the tolerance error is 34%. A tolerance error within 100% indicates that the test point is passed.

TABLE I
COMMON FUNCTIONS, RANGES AND RESOLUTIONS FOR A TYPICAL DMM OR MFC

Function	**Range**	**Resolution**
Direct Voltage 0 to ± 1100 V	100 mV, 1 V, 10 V, 100 V, 1000 V	10 nV on 100 mV range
Direct Current 0 to ± 1 A	100 μA, 1 mA, 10 mA, 100 mA, 1 A	1 nA on 100 μA range
Alternating Voltage 1 μV to 1000 Vat 10 Hz to 1 MHz	1 mV, 10 mV, 100 mV, 1 V, 10 V, 100 V, 1000 V	1 nV on 2 mV range
Alternating Current 1 μA to 1 A at 10 Hz to 10 kHz	100 μA, 1 mA, 10 mA, 100 mA, 1 A	1 nA on 100 μA range
Resistance 0 to 100 MΩ	10 Ω, 100 Ω, 1 kΩ, 10 kΩ, 100 kΩ, 1 MΩ, 10 MΩ, 100 MΩ	1 μΩ on 10 Ω range
Capacitance 10 nF to 100 mF	10 nF, 100 nF, 1 μF, 10 μF, 100 μF, 1 mF, 10 mF, 100 mF	10 nF on 1 mF range
Frequency 1 Hz to 1 MHz	10 Hz, 100 Hz, 1 kHz, 10 kHz, 100 kHz, 1 MHz	1 mHz on 10 Hz range

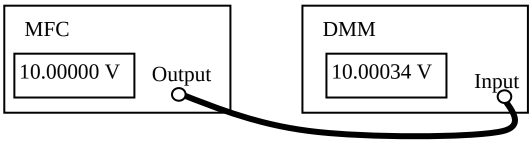


Fig. 1. Calibration system illustration.

D. *Calibration Engine*

Calibration engine remotely controls the instruments settings, output (stimulus source), data acquisition (retrieve readings), command translations, mathematic calculations and results evaluations.

1) *Command translations*: The first requirement in the calibration engine is to unify all the instrument IEEE-488 commands to become generic or common. Two different instrument models can set to a similar function using different IEEE-488 command syntax. See the example in Table II.

TABLE II
GENERAL FORMAT COMMAND FOR DIFFERENT TYPE OF INSTRUMENTS

Instrument Model (Manufacturer)	**Function**	**IEEE-488 Command (Manufacturer)**	**General Format (Unify)**
3458A multimeter (Agilent)	AC voltage at 10 V range	FUNC ACV, 10	ACV
	DC voltage at 10 V range	FUNC DCV, 10	DCV
5700A calibrator (Fluke)	AC voltage 10 V @ 1 kHz	OUT 10V, 1KHZ	ACV
	DC voltage 10 V @ 1 kHz	OUT 10V, 0HZ	DCV

The application shall be able to unify different models of instruments to a general format command to represent particular function. 3458A requires “FUNC ACV, 10” command to be sent via IEEE-488 to set a function of ac voltage at 10 V range. 5700A requires “OUT 10V, 1KHZ” command to be sent via IEEE-488 to set a function of ac voltage 10 V @ 1 kHz. Both instruments have similar functions but different IEEE-488 commands. The calibration engine shall introduce descriptions for general commands to unify different IEEE-488 commands. The example of general command in Table II is ACV.

The purpose of the general format commands is to ease and simplify the design of calibration procedure. Users only need to apply the general command instead of applying the long and confusing IEEE-488 syntax.

The general command described above is an analogy to a word in a language. A set of words will form a sentence and a set of sentences will form a passage. Similarly, a calibration procedure is formed by a set of instruments commands. For example: ACV (3, 1E6) command is to send to a source generator to output ac voltage function of 3 V at 1 MHz. Next ACV (10) and MEAS (6) commands are sent to a measuring meter to set to ac voltage function at 10 V range and then start measuring the output six times. These are simple steps of calibrating a single test point. When there are many test points in a procedure, many steps and commands will be involved. The second calibration engine requirement is to develop the calibration procedure.

2) *Command header*: The calibration engine shall recognize whether the header is a model identifier by checking through the Instrument Command relation (command translation). The other header types and their functionalities are shown in Table III.

TABLE III
KEY FUNCTION FOR HEADER AND ITS CONTENT

Header	**Content / Description**
IEEE	*<Address><IEEE command>* If the *Instrument Command* relation does not provide the required general function, use this header to send IEEE command directly to the address inside the square bracket.
RNG	*<Number>* Set the range of the UUT

TOL	*<Number><Unit* or *PPM* or *Percentage><UUT Indicated Value* or *UUT Range>* Set the tolerance limit of the UUT and then the calibration engine shall evaluate the UUT indicated value with the standard value.
MSG	*<Text>* A message box shall popup with the text.
INPUT	*<Text>* An input box shall popup to request input.
MATH	*<VB Script>* MATH shall execute the content in the form of *VB Script*.
PIC	*<Picture filename and path>* The picture of the selected file shall be shown.
DESC	*<Text>* Description of the test point shall be stored in the *Result* relation.

III. SYSTEM SOFTWARE DESIGN

The system software consists of the following components:

1) *Calibration database*: storing all the applications data.

2) *Server application*: provides services to the client such as user login, calibration procedures, instrument inventory, instruments commands and calibration result storing.

3) *Client application*: login to server application and retrieve necessary information to perform site calibration.

4) *Calibration engine*: instrument command that communicate with different types of instrument models to perform calibration procedures that consist of setting up instruments, retrieve readings from instruments, calculation, error of measurement and generating results.

A. Design Overview

In Fig. 2, the database is located at the server computer. The database design will be shown in a later topic on the entity-relationship diagram. The server application connects to database to read, update, add or delete. The server application is also capable of performing remote procedure calling. The remote procedures allow clients connected to network or internet to call the procedure remotely. In this way, the client computer is able to communicate with the server computer. The client application retrieves necessary data from server and performs the calibration.

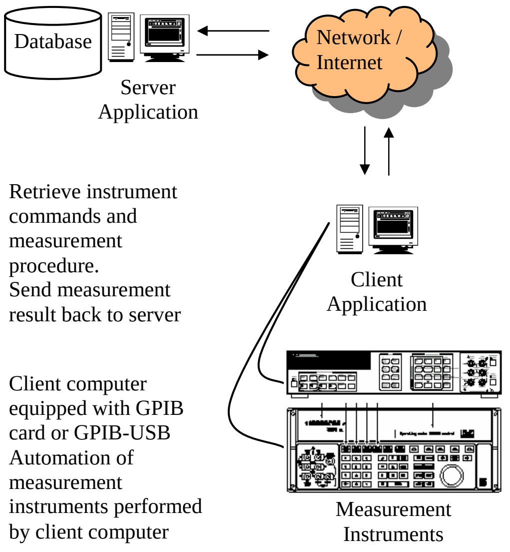


Fig. 2. Overview of internet-based calibration system

B. Calibration Database

Fig. 3 shows that a user selects an instrument as an item to be calibrated. This item is then calibrated by calibration procedure. Since a calibration procedure is only meant for an instrument, and an item may have more than one calibration procedure, hence the calibration item and calibration procedure are a many-to-one relationship.

The calibration procedure then extracts the contents from the calibration step. The calibration steps are the details of procedure that show the details of the measurement flow. A calibration step only belongs to a calibration procedure but a procedure may consist of many steps resulting in the relationship of one-to-many.

The calibration procedure also needs to obtain what instruments are used in the measurement. The relationship is one-to-many due to a procedure requiring several instruments. Next the instruments used in the procedure maps the instruments commands. Their relationship is one-to-many.

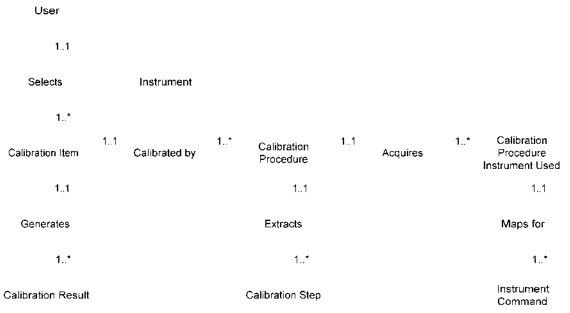


Fig. 3. Entity-relationship diagram

Notice that the entity “Calibration Procedure Instrument Used” is not found in the requirements, but the calibration procedure need to know what equipments are involved in the calibration and which commands need to be mapped. Even though the calibration step has an attribute called “Header” that sometimes is made up of instrument model identifier, yet it still needs the entity “Calibration Procedure Instrument Used” to summarize the instruments required in a procedure.

Finally the calibration item will generate the calibration result after the calibration. An item calibrated at a time may consist of a lot of test point results. Hence they have one-to-many relationship.

C. Server and Client Applications

Node instance shape in Fig. 4 is a run-time physical object that represents a processing resource. The server node and the client node represent computers whereby the instruments node represents the instruments.

There are two components in the server node namely server application and database. The server application establishes connection with the database. Communication between server application and client application across different nodes through LAN or internet can be accomplished using Microsoft Distributed COM (DCOM) [15]. DCOM extends the Component Object Model (COM) to support communication among objects on different machines. Once the connection has been established, client can call the methods from the server. To materialize the communication process, the internet or network connection between server node and client node must exist. This project is focused on the application layer of the network.

The client node needs an IEEE-488 GPIB interface card to be installed. The client application interfaces with the driver provided by the vendor of the GPIB hardware in order to communicate with the instruments. The GPIB interface in Fig. 4 is physical GPIB connectors. The GPIB driver needs no further design details because it is already properly implemented and tested by vendor. Lastly the instruments node is the machine that performs the measurement.

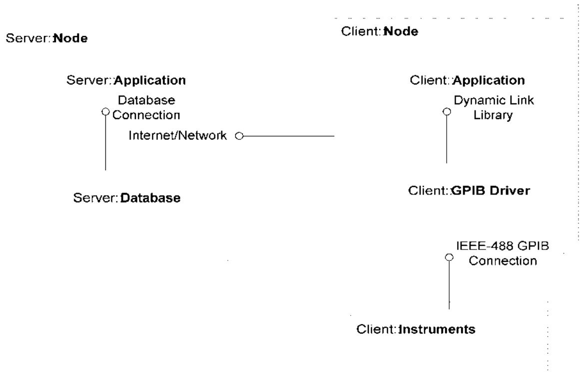


Fig. 4. UML component diagram for client/server

D. *Client Application*

Fig. 5 shows the activities diagram of the client application. The initialization starts with the creation of server application object. The objects refer to the previous topic, DeServer. The creation of the DeServer object allows client application to use the operations. Using the DeServer operations across different nodes is also known as remote procedure call. The login activity calls DeServer→Login operation to verify the user. Upon returning of false value indicates that the login has failed, resulting from end of runtime.

The next activity "Load UUT Inventory" calls DeServer→InstUut to return the instruments records that belong to the login user. These instruments imply those with UUT status. "Load Serial Number" is a local operation that extracts serial number from the instruments records. The serial number is also equivalent to the instrument identifier. After the selection of instrument id, the calibration procedure is loaded according to the id. "Start Calibration" activity is shown in Fig. 6.

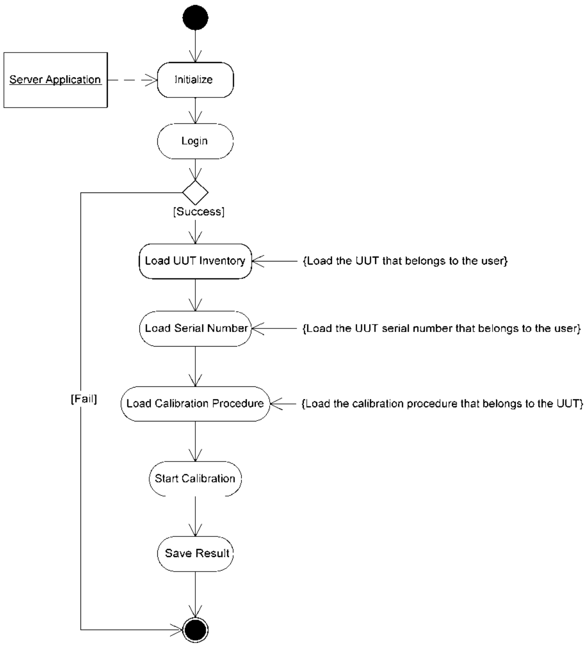


Fig. 5. Activity diagram for start Calibration

E. *Calibration Engine*

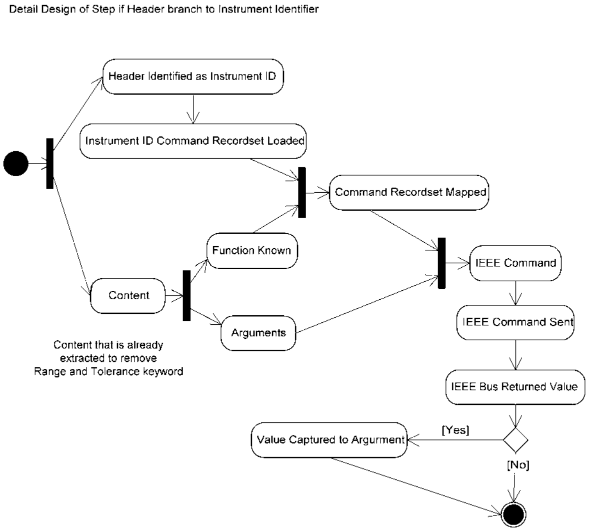


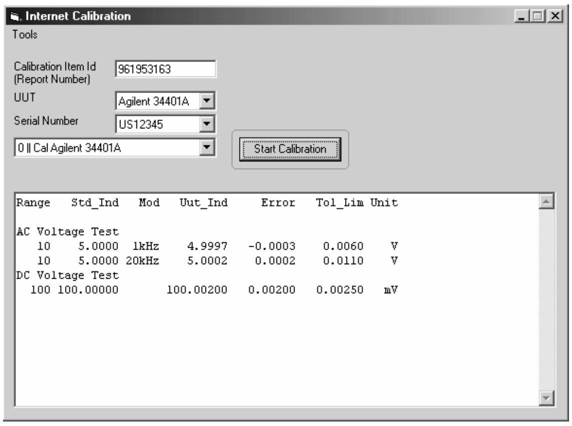
Fig. 6. UML activity diagram for calibration engine

IV. GRAPHICAL USER INTERFACE AND IMPLEMENTATION RESULTS

The designs of GUIs are kept simple and user friendly for practical use and efficient accessibility. The client application GUI is as shown in Fig. 7.

A user is required to insert the name and the password in order to access the calibration system. After user login at the first level, the user will be directed to the interface shown Fig. 7. The server GUI also logs the client activities in the server computer (Fig. 8).

The content of the calibration GUI is made up of two textboxes, and three dropdown list boxes and a button. Calibration Item Id is the calibration report id that allows user to insert any value. The UUT list box is the concatenation of manufacturer and model retrieved from instruments relation. The selection of UUT will affect the loading of serial number list and the calibration procedure list. On completion of data inputs, clicking the Start Calibration button leads to the process of calibration and the result will be displayed in the textbox and also return to the server application (Fig. 9).



Calibration Item Id (Report Number): 961953163

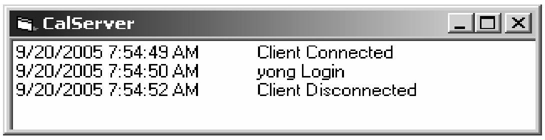
UUT: Agilent 34401A

Serial Number: US12345

0 || Cal Agilent 34401A

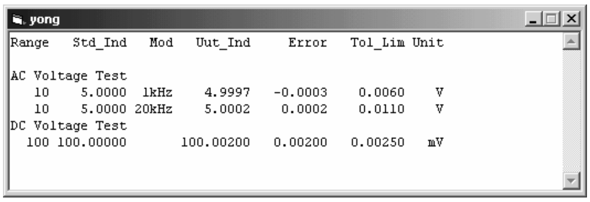
```
Range   Std_Ind   Mod   Uut_Ind     Error   Tol_Lim Unit

AC Voltage Test
   10    5.0000  1kHz    4.9997   -0.0003    0.0060    V
   10    5.0000 20kHz    5.0002    0.0002    0.0110    V
DC Voltage Test
  100 100.00000       100.00200   0.00200   0.00250   mV
```

Fig. 7. Calibration interface



```
9/20/2005 7:54:49 AM        Client Connected
9/20/2005 7:54:50 AM        yong Login
9/20/2005 7:54:52 AM        Client Disconnected
```

Fig. 8. Server log



```
Range   Std_Ind   Mod   Uut_Ind     Error   Tol_Lim Unit

AC Voltage Test
   10    5.0000  1kHz    4.9997   -0.0003    0.0060    V
   10    5.0000 20kHz    5.0002    0.0002    0.0110    V
DC Voltage Test
  100 100.00000       100.00200   0.00200   0.00250   mV
```

Fig. 9. Server results

V. CONCLUSION

The server application that resides at the national laboratory computer provides calibration procedures, instrument inventory, instruments commands and user information services. The server application grants permission to client application. With the permission, client is able to retrieve data using DCOM remote procedures. The server application displays the login user identity and the instant calibration results.

The client application is also developed to establish DCOM connection with server application in order to acquire services. Users are able to select inventory, calibration procedures and perform on-site calibration.

ACKNOWLEDGMENT

I would like to express my deepest gratitude to my supervisor, Assoc. Prof. Dr. Roziati, who has indeed supported and guided me throughout the whole project.

REFERENCES

[1] “International vocabulary of basic and general terms in metrology, (VIM),” 2nd ed., *ISO, Geneva*, 1993.
[2] “Traceability of measuring and test equipment to national standards,” *European Cooperation for Accreditation of Laboratories*, November 1995.
[3] K. Salminen, “Introduction to IEEE-488,” *HP-VEE Work 0.5 c.u.*, 1997.
[4] R.A. Dudley, “Handbook of measuring system design: internet calibration,” *John Wiley & Sons, Ltd., UK*, 2005.
[5] R.A. Dudley, and N.M. Ridler, “Internet calibration direct to national measurement standards for automatic network analysers,” *Centre for Electromagnetic and Time Metrology National Physical Laboratory, Teddington, Middlesex, TW11 0LW, UK*, May 2001.
[6] NIST, “Internet-assisted remote calibration,” *http://www.eeel.nist.gov/, National Institute of Standards and Technology, USA*, 2000.
[7] J. Korthout, “Industrial use of self-calibration and internet measurement,” *Workshop on Internet Measurement & Self-Calibration, NMi, Nederlands Meetinstituut*, May 2002.
[8] R. Gert, “Internet metrology at NMi”, *Workshop on Internet Measurement & Self-Calibration, NMi, Nederlands Meetinstituut*, May 2002.
[9] K. Guckelsberger, “Overview of e-measurement projects at PTB,” *Workshop on Internet Measurement & Self-Calibration, PTB-Physikalisch-Technische Bundesanstalt*, 2002.
[10] H. Dietrich, and L. Gerhard, “Challenges and chances of internet metrology,” *Measurement Science Review, Volume 3, Section 1*, 2003.
[11] ICS, “A tutorial about GPIB bus,” *IEEE-488 application bulletin, Pleasanton, CA. ICS Electronics*, 2003.
[12] D. Jernigan, and T. Dehne, 1999. “Programming with NI-488.2™ Software,” *Application Note 002, National Instruments Corporation, USA*, 1999.
[13] A.F. Behrouz, “Data communication and networking, 2nd ed.,” *McGraw Hill, New York*, 2002.
[14] G. Coulouris, J. Dollimore, and T. Kindberg, “Distributed systems: concepts and design, 2nd ed.,” *Addison-Wesley, England*, 1994.
[15] R. Lhotka, “Professional Visual Basic 6 Distributed Objects,” *Canada, Wrox Press*, 1999.