Remote Monitoring of Network Element Sites Using LabVIEW System

by Ralph Dick, NVSI © 1999

The Challenge:

Remote monitoring via Internet and Intranet of High Speed Fibre Optic Network Elements for transmission quality and remote troubleshooting.

The Solution:

Developing a Transmission Network Management System using Windows NT 4.0, GPIB-ENET Controllers over Internet and Enternet, integrating LabVIEW with Microsoft SQL Server and Internet Information Server.

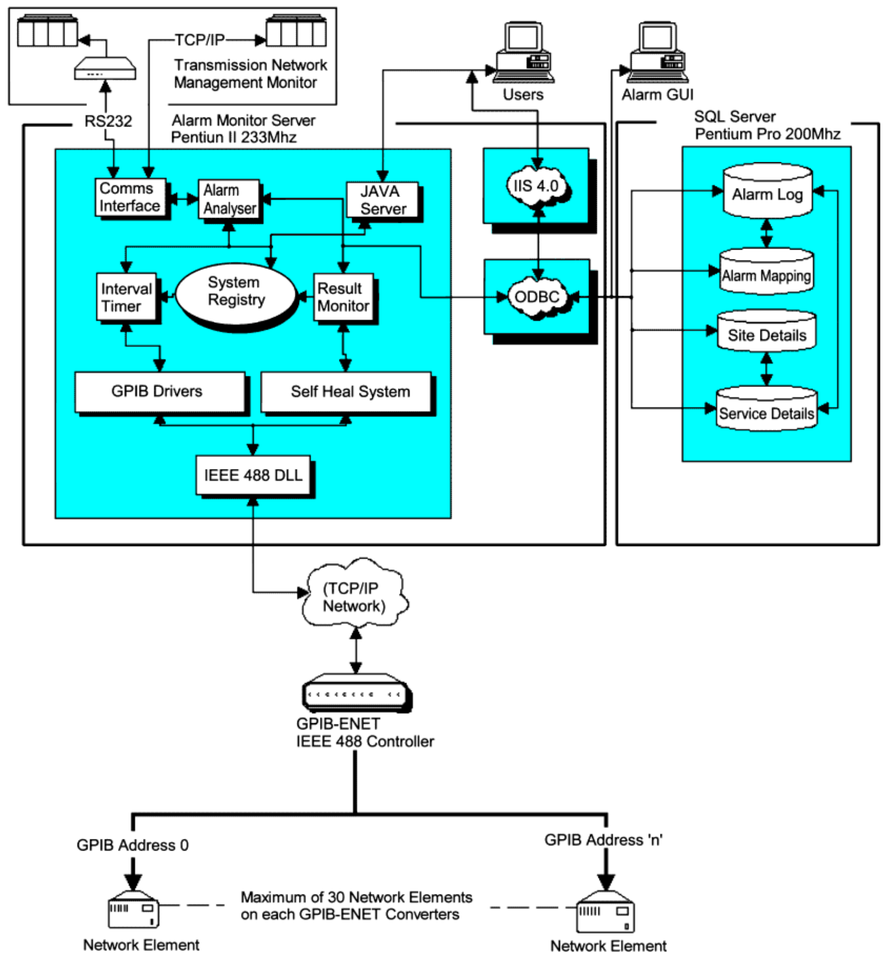

Figure 1 System Overview

Creating a system that is able to monitor and diagnose Network Element Devices (NED) to update Transmission Network Management Systems (TNM), Web Browser Diagnostic System and Independent Alarm Enquiry System (IAE).

This system is based around LabVIEW and Microsoft NT 4.0 operating system, SQL version 6.5, IIS 4.0 and Access 97. The objective of the system is to enable engineers and technicians to monitor and diagnose the high-speed data that is used in a Fibre Optic Network. The other main objective is that the system must be able to recover from any fault condition in the TNM system without losing any critical information.

The heart of this system is two NT 4.0 servers, the first with LabVIEW control system and the web server. The second is the SQL server. The information from this system is sent to a legacy TNM system, web browsers and IAE system (Figure 1 System Overview).

System Features

The system monitors remote NED across or between countries, wherever there is an Internet connection with conductivity to the Enternet system where the NED are connected. This system collects data that is passed to TNM as conditions change in any NED and handles any requests that the TNM requires. The data is passed to the system by two methods, using RS 232 and the TCP/IP network.

The IAE is used to acquire active alarms, cleared alarms and all alarms. The system uses the TCP/IP network to retrieve data from the SQL server. Only the system administration team uses this application.

The web browser application selects individual sites to retrieve several types of information. One displays any active alarms (Figure 3 Alarm Monitor Web Page) and another is the diagnostic mode (Figure 4 Diagnostic Web Page). The active alarm page indicates the type of alarm and when it occurred, the customer, and configuration information. The diagnostic mode displays the NED state which reports usage time, code violations (CV), loss of light, status of the laser, CV threshold and domain errors. Also displayed is the system operational status, which is indicated by a JAVA Applet heartbeat. There are two other indicators that indicate the status of the NED. These display the time of the last NED polling and the time of the last diagnostic command. A summary of the configuration information is also displayed with the key feature being a graphical representation of the customer's network showing the current operational status.

System Operation

The system is basically made up of eight functional components. These are as follows; the NED, GPIB-ENET network system (Figure 2 GPIB-ENET System), LabVIEW control system, SQL database, IIS web server, TNM systems, IAE system and operational user web browsers (Figure 1 System Overview).

The NED has a GPIB port and uses a proprietary control protocol, which is connected to the Internet using GPIB-ENET controllers. The system is set up to have a maximum of 30 NED per GPIB-ENET. The LabVIEW control system communicates to the NED using the Internet system. A custom multi-threaded GPIB DLL driver was used for communication with the NED. This program has nine major components that communicate with each other via the System Registry and cues.

The System Registry holds all critical data pertaining to the system. An Interval Timer communicates to each NED via its own separate GPIB driver, ensuring that there are no clashes with any other NED under normal conditions. The results from each NED are passed back to this GPIB driver and then fed into a Results Monitor queue where the data is processed for a number of conditions. If any error occurs in the return data it activates a Self Heal system, which remains in constant communication with the appropriate NED until the problem, has been resolved. While this is occurring, the appropriate register is set to ensure that the GPIB driver is taken offline and only

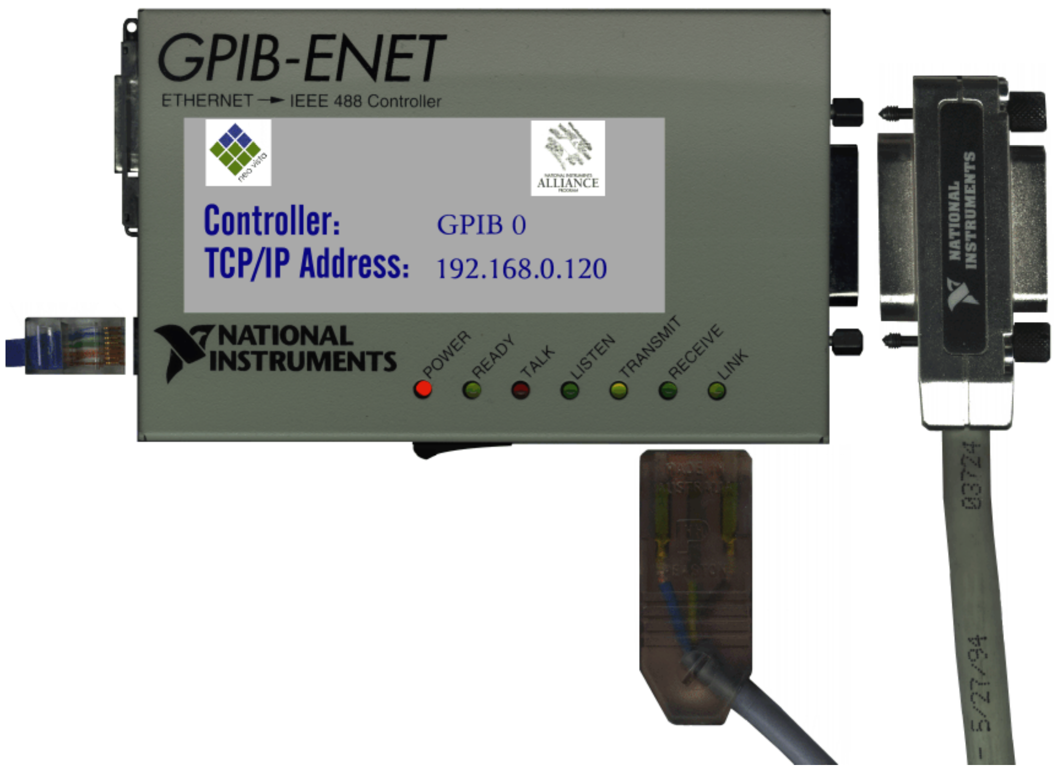

Figure 2 GPIB-ENET System

returned to online status when communication has been completely restored. Both systems use the custom GPIB DLL driver.

The Results Monitor updates the System Registry and processes information for the Alarm Analyser. The Alarm Analyser then passes this information to the Communication Interface, which in turn passes the information to the TNM system, and also to the SQL server via the ODBC communication layer. The Communication Interface also requests

information from the Alarm Analyser and retrieves it from the SQL server using the ODBC communication layer.
The web pages that the service personnel use initially get their information from the IIS web server, which requests the appropriate information from the SQL server. When the user accesses the diagnostics web page (Figure 4 Diagnostic Web Page) a JAVA Applet is executed. This JAVA Applet retrieves information from the System Registry giving the user all the necessary information about that particular NED. The Applet is also able to send command instructions to the NED that is being diagnosed.

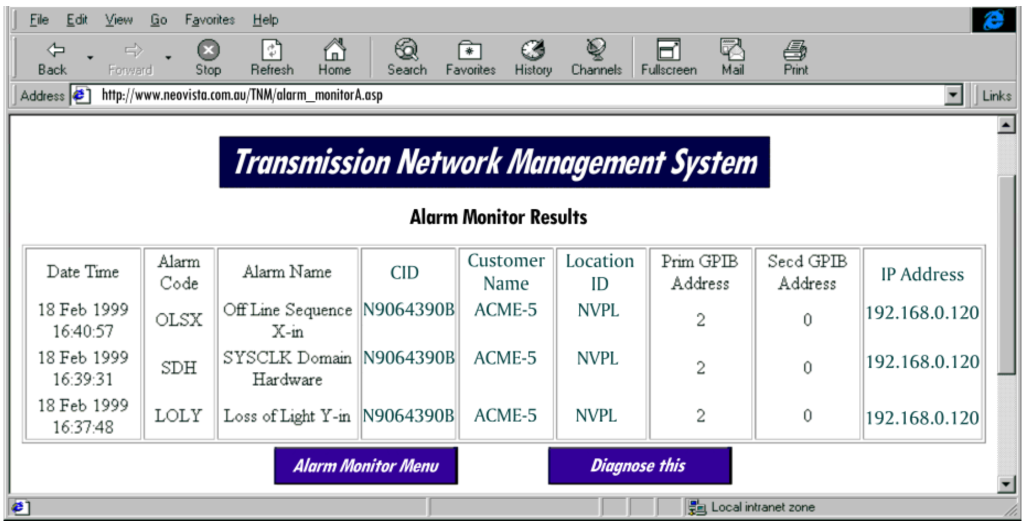

Figure 3 Alarm Monitor Web Page

The final system is the IAE, which is only used by the system administrator. This system is also a database that retrieves all of the alarm information from the sequel database server via the ODBC communications layer. None of the information can be altered with this information, it can only be displayed. The data can be displayed in a number of formats, i.e. active alarms, cleared alarms and all alarms.

Technological Challenges

There were five major technical challenges in this project. The NED, network conductivity, GPIB conductivity, the program fault tolerance/Self Heal system and the web browser system.
The first challenge was to reverse engineer the NED as there was very little available information. This was a legacy device that had limited documentation. Using the NI GPIB/Analyser PCMIA card the necessary functions had been determined resolving any anomalies in the documentation. It was found that this system uses a unique protocol for control and error checking. The LabVIEW GPIB Vis were unable to communicate to this device. To resolve this issue a custom GPIB DLL that met the requirements of the NED was used.
It was identified that the most probable failure mode in this system would be TCP/IP network. The GPIB-ENET had to be re-initialised using a specialised control sequence otherwise the device would go off line and could only be re-initialised by power reset, which would require a technician to travel to the remote site to re-initialise the device. The only way to detect the presence of a GPIB-ENET box is to open a TCP/IP port that is unique to that device.
The system is also required to detect a GPIB failure. The system must distinguish if there is loss of connection to the NED via the GPIB, this indicates that the GPIB bus has failed or the NED has power loss or is physically not present. The last failure fault mode to be detected is communication errors from the NED.
The Self Heal system must determine the failure mode of the GPIB system and diagnose which of the above failure modes is present then continue to poll the instrument until communication is reestablished. A higher level of the Self Heal system is required to handle any loss of communication to the sequel server. When communication has been restored to the sequel server all alarms logged while the server was offline must be must updated.
One of the requirements was the ability of the system to use a number of web browsers, i.e. IE 3.02 and up. Not all of these browsers met the requirements of the JAVA program. This made development of the JAVA Applet extremely complex. All functions needed to be put onto a single Applet instead of a number of Applets.

Conclusion

The final product is an extremely complex system made up of a number of modules and technologies. This made testing of the system a difficult task. To add to this complexity the system had to be Y2K compliant and able to exchange data with legacy systems so that all information was Y2K compliant.

Each of the technologies used presented a number of challenges. The final result was a combination of LabVIEW and Microsoft products resulting in a solution that met all of the project requirements. This type of project requires that the programmers involved have extensive knowledge not only of LabVIEW but also the necessary Microsoft products used.

To undertake this type of project requires top down design topology and low-level concept design confirmation of sub systems to ensure that the project can be successfully undertaken. The key factors in the success were the project management, weekly project meetings and strict adherence to the appropriate quality standards.

Further Information

Neo Vista Systems Integrators Pty Ltd
web: www.nvsi.com.au
email: info@nvsi.com.au
phone: +61 2 9809 7899
fax: +61 2 9809 7499

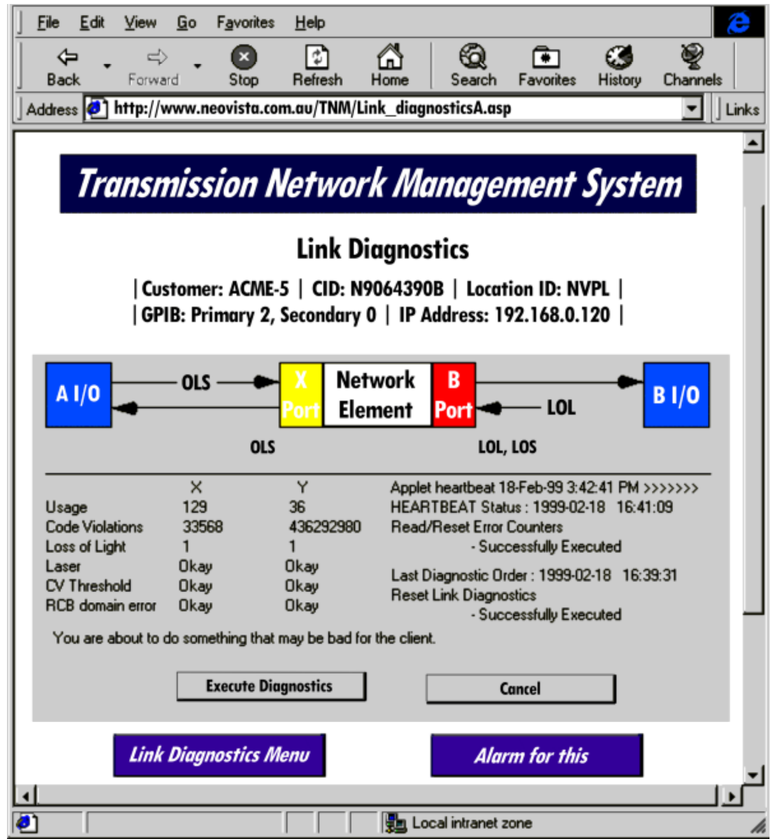

Figure 4 Diagnostic Web Page