Web-based control software enables users to control experiments remotely, view data from multiple locations, and geographically distribute processing and testing tasks.
n a very real sense, the Internet has changed the way we think about information and exchange of resources. Now engineers are using the Internet and software applications to remotely monitor and perform distributed execution of test and control applications. Such an approach reduces the time and cost involved in tests by sharing optoelectronics instrumentation and by distributing tasks to optimal locations.

A typical automated test and control system uses a computer to control positioning equipment and instrumentation. We’ll use the term “remote control” to refer to the technique of enabling an outside computer to connect to an experiment and control that experiment from a distance. Such an approach benefits engineers who need to monitor applications running in harsh environments that offer them limited access, or for tests whose long durations are impractical for continuous human monitoring.

In addition, remote control offers engineers the ability to change test parameters at certain intervals without traveling to the site or even running from their office into another area of the building. This convenience allows a test operator to view results and make test modifications from home on the weekend, for example. The user simply logs on to the network from home, connects to the application, and makes those changes just as though he or she were on site.

control via Internet
To effectively control applications via the Internet, companies are developing software programs that champion remote execution. For instance, LabVIEW (National Instruments; Austin, TX) allows users to configure many software applications for remote control through a common Web browser simply by pointing the browser to a Web page associated with the application. Without any additional programming, the remote user can access fully the user interface that appears in the browser. The acquisition still occurs on the host computer, but the remote user has complete control of the process and can view acquired data in real time. Other users also can point their browser to the same URL to view the test.

Windows XP makes it easier to control applications via the Internet. With this Microsoft OS, users now get Remote Desktop and Remote Assistance, which offer tools for debugging deployed systems. After a system is deployed in the field, it is often cost-prohibitive for the support staff to visit every site. With Remote Desktop, a support operator can log in to a remote Windows XP machine and work as if he or she were sitting at the desk where that machine is located. With Remote Assistance, the onsite operator can remain in control of the desktop but the support operator can view the desktop on his or her remote machine. At any time, the onsite operator can give up control of the desktop to the support operator and still monitor which troubleshooting techniques are in use. Industry-standard software development tools take advantage of these new features.

At times, it may be desirable to use the Web browser to initiate a measurement or automation application but not actually control the experiment. In this case, the remote operator can log in, set certain parameters, and run the application over a common gateway interface (CGI). With CGI, the user communicates with a server-side program or script run by an HTTP server in response to an HTTP request from a Web browser. This program normally builds HTML dynamically by accessing other data sources such as a database. As part of the HTTP request, the browser can send to the server the parameters to use in running the application.

distributed execution
In classical remote control, one person or machine at a time is charged with controlling the experiment. In distributed execution, however, a user can truly take advantage of the benefits of networking, extending control to an entire remote system connected on the same network. In this way, individual machines focus on specific functions, and each system is optimized to perform its chosen task. Because data can be

![Figure 1](image-url)
shared among the distributed components and each component accomplishes a unique task, this network functions as a complete system. For instance, it is possible to dedicate certain machines for acquisition and control while relegating analysis and presentation to other systems. Technology makes it possible to remotely monitor, control, and run diagnostics while the system itself is dedicated to running acquisition and control, introducing the ability to multitask.

Certain test and control applications require an embedded, reliable solution. For these applications, the user can download the software to a headless, embedded controller to connect and control remotely. The controller can be a single unit or a series of form factors (such as the FieldPoint module that is able to perform monitoring and control tasks in harsh environments). In either case, software runs on a real-time operating system, but it can be accessed from a host computer using an Ethernet connection.

For example, consider a structural test system measuring the vibration and harmonics of a bridge design. It is possible to set up one node with a camera to monitor the testing of the bridge, then set up another node to measure parameters such as temperature, humidity, and wind direction and speed. Finally, one can set up a node to measure the load, strain, and displacement on certain areas of the bridge. The system can send all the data back to a main computer that correlates the data, analyzes it, and displays the results of the test on a Web page.

Each of these nodes would need to be running autonomously, acquiring data and sending it onto other computers to correlate the data and create reports. With the right software and hardware, each measurement node becomes an embedded, reliable, and durable solution. The user could easily control any of the measurement nodes to modify parameters of the test. In some systems, the origin of the test and the code is completed using a Windows operating system and then downloaded to the measurement node. This enables the user to make major modifications to the test and download them to the embedded target without visiting the site.

Next, one of the live data-sharing techniques could be used to transfer the data to another cluster of computers that would correlate and analyze the data. Finally, an Internet server could allow project members to share the Web reports and analysis in geographically separated locations.

data sharing

A key to accomplishing remote control and distributed execution is the data-sharing ability inherent to the Web. With new software programs, live data sharing can be as easy as simply right-clicking the item and placing a checkmark in a checkbox, which saves time for users and allows them to take advantage of the Web economies of scale such as efficient data transfer from one computer to another and the ability to access data in real time. Applications must also afford users real-time access to acquired data to control or monitor a process or perform a test across a network.

Sharing data leads to convenience—users can be remote while control applications are running, and contact methods can extend to mobile phones or pagers. For example, certain software programs allow users to send e-mail alerts. Electronic notifications can be created that allow operators to receive alerts from the production area via mobile phones or pagers when certain process values exceed established limits; at that point, the operator can log on to control the application. Such updates generated automatically during the testing process free up operator time to be spent on more productive tasks. As an example, this technique would be useful for a small company running burn-in tests, which can take six to 10 hours. With the type of system described above, the engineer could go back to his or her desk and receive an alert if test results don’t fall within set test parameters.

With distributed execution tasks, the network enables users to access various measurement nodes. It is possible to develop software that uses each computer to complete a portion of the application; a test could have several acquisition nodes, each sharing data with the main computer or cluster of computers that perform the analysis, generate reports, and send them to the Web.

XML and other strategies

For data sharing, extensible markup language (XML, which enables definition, transmission, validation, and interpretation of data between applications and organizations), is quickly becoming a standard way to transfer data in a text-readable fashion that can easily be displayed on the Web. Because of the universal XML standard, one can generate a Web report featuring a defined data set and easily import it into other applications. Because the data is readily accessible, applications can download any XML document, parse the data, and perform custom analyses. Some software applications now include built-in functions for creating or reading XML documents.

Manufacturers have realized the enormous cost benefits of using common off-the-shelf, Internet-related hardware and software components to communicate process data. The same technology used for Internet applications can also be used to connect the enterprise. On the plant floor, data acquisition and automation systems serve as information-access points to the larger corporate IT systems. Data can be transported using existing, widely accepted protocols to guarantee not only interconnectivity but also interoperability. The workforce is already trained to fetch and use data supplied through a browser.

Figure 2 This application sends an e-mail alert when specified limits are exceeded.
NI’s DataSocket provides another method of sharing data directly with other parts of an organization. DataSocket implementation requires no extra development time—it streams the data in a graph or other user interface item over the network. Because DataSocket also is implemented as an ActiveX control, a Java Bean, and a component of Measurement Studio for C/C++ and Visual Basic development, users can incorporate the technology in many other applications. Project members who want to subscribe to the DataSocket Server item that contains the data use a URL to begin receiving data and any updates sent. With DataSocket, engineers can generate Web pages to display quality information from a manufacturing floor, changing properties of materials during an ongoing test, or even updates of the weather.

**the drawbacks**

Although remotely controlling applications and distributing control via the Web has countless benefits related to operator convenience, as well as company time and cost savings, operators should also be cognizant of possible drawbacks. High amounts of traffic on the network could lead to slow updates or data transfer. The method of communication (Ethernet) is not a deterministic bus and offers no guarantee that data or execution will occur in a reliable amount of time.

Security is often a concern of Internet-related activities. If the remote system is on the same network as hundreds or millions of other users, the potential exists for possible system interference. Test and control applications should be implemented so that the network is protected by existing IT security systems. Best practices call for users to work with IT professionals to determine the best way to implement Web-based control applications without interfering with the particular IT system security.

In addition, many people could be trying to access the same application simultaneously. This requires companies to choose applications capable of handling multiple users accessing at the same time. If multiple access to an application is not possible, the users ultimately accomplish no more than they would through a single transaction.

The benefits of Web-based control far outweigh the disadvantages. Although certain hindrances may occur as a result of doing business on a network shared by millions, the advantages of convenience, cost, and time prompt software developers to investigate new ways to deal with the potential problems. For example, to avoid user confusion, software constraints can limit access so that only one client can control the application at a time, but that control can pass easily among the various clients at run-time. In addition, the host computer can take control of the application away from any of the remote clients at any time. The technique can also minimize cost by allowing service personnel to control and test remotely, for example.

The Internet is changing the way we control our applications by providing new ways to take measurements and distribute results. Many different options exist for remotely controlling applications and distributing execution. The best software programs allow users to take advantage of the power of the Web without having to become experts in any of its technologies, helping them incorporate the Internet into many different aspects of their application. This allows companies to integrate their applications easily into the existing corporate networking infrastructure so they can increase the productivity of those performing control.

“Remote control offers engineers the ability to change test parameters at certain intervals without traveling to the site or even running from their office into another area of the building.”

Kris Fuller is a product manager at National Instruments, Austin, TX. Phone: 512-683-5032; fax: 512-683-5569; e-mail: kris.fuller@ni.com.