



## Using the DNP3.0 Protocol via Digi Device Servers and Terminal Servers

---

For years, electric power utilities have relied on Digi internal serial cards (i.e., *DigiBoard® solutions*) to connect UNIX, Linux® and Windows® based Supervisory Control and Data Acquisition (SCADA) applications to remote power control and monitoring devices. These devices include, but are not limited to, Remote Terminal Units (RTUs), Intelligent Electronic Devices (IEDs), meters, relays and breakers. While protocols such as Modbus and others are used in SCADA systems, this document specifically discusses Distributed Network Protocol (DNP).

Serial ports on the SCADA system in the central office typically connect to modem banks. These modems connect to leased line or dial-up modems, then to serial ports on remote devices. Large substations can have 128 or more serial devices. Smaller locations can range from one to four devices. A typical installation is 16-32 ports.

Today, more and more utilities are connecting remote substations and even pole-mounted devices via Ethernet (wireless and fiber) TCP/IP connections. This application note describes two proven methods for DNP3 communications using Digi One® and PortServer® device servers and terminal servers. These Digi servers can be located at remote locations, eliminating costly modem connections while providing real-time access to most any serial device via TCP/IP communications.

Most SCADA and other control applications provide two methods of communicating to serial ports:

1. TTY or COM ports: Multiport internal serial adapters provide additional TTY or COM ports to an application. Drivers are loaded on the SCADA host computer to create and recognize the additional ports.
2. TCP Sockets: Many newer SCADA applications support this method of direct TCP/IP communications. DNP data is sent to serial ports of a remote terminal server via a TCP Socket connection using the terminal server's IP address; a TCP port number associated with the individual serial port is used to identify the data path (e.g. 192.168.10.5:2101). DNP data is simply encapsulated into TCP by the sender (i.e., SCADA master application) and then removed by the receiver (the Digi server). [Note: The Digi ports are set to TCP *server* mode. See the Digi server manuals for details.]

Digi device and terminal servers support both methods, providing seamless transition from internal serial ports to network connected serial ports. Following is a description of how it works.

### **TTY/COM Port Redirection Via RealPort®**

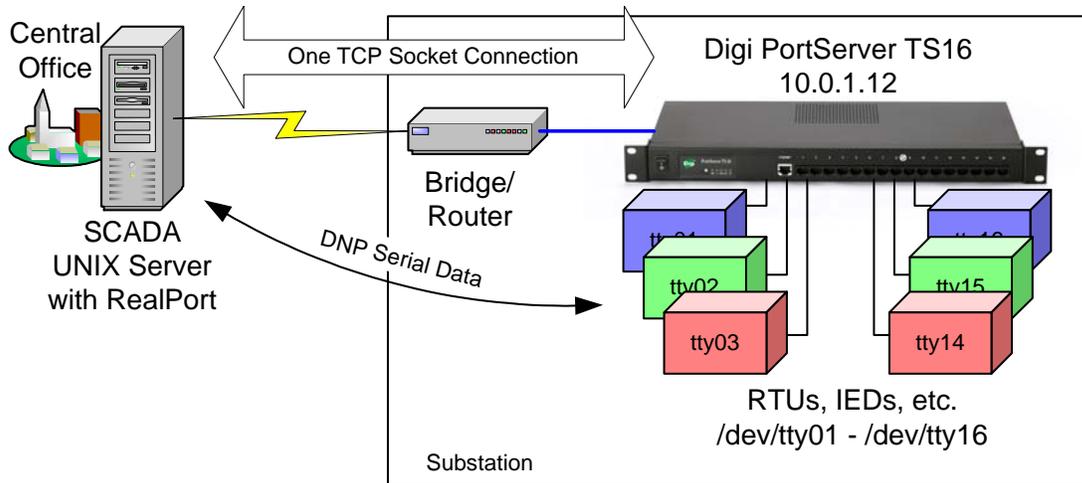
Digi's patented serial port redirection protocol and driver, RealPort, creates real TTY or COM port devices on a local SCADA system; the physical ports can be located anywhere an Ethernet (hardwired or wireless) TCP/IP connection is available. The user installs the RealPort driver on the SCADA system, which creates additional COM or TTY ports. These serial ports are mapped to the IP addresses of the remote Digi servers.

RealPort can handle virtually any serial protocol (including DNP, Modbus and others), just as if it is on a computer's local serial port. It is ideal for legacy applications that support only local TTY or COM serial ports. RealPort supports many different operating systems. [See [www.digi.com/support](http://www.digi.com/support) for a current list of supported operating systems.]

Also note that some versions of RealPort and Digi PortServer TS device and terminal servers support *encryption* to provide secure data communications between the SCADA system and the Digi server.

## Digi Application Note - Using DNP3 Protocol via Device and Terminal Servers

Example: A PortServer TS 16 terminal server is installed at a remote substation. The substation is connected to a UNIX based SCADA server by means of Ethernet radio routing TCP/IP. TTY ports are mapped to the IP address of the PortServer TS 16. For example, /dev/tty1 thru /dev/tty16 are located on a PortServer TS 16 at 10.0.1.12.



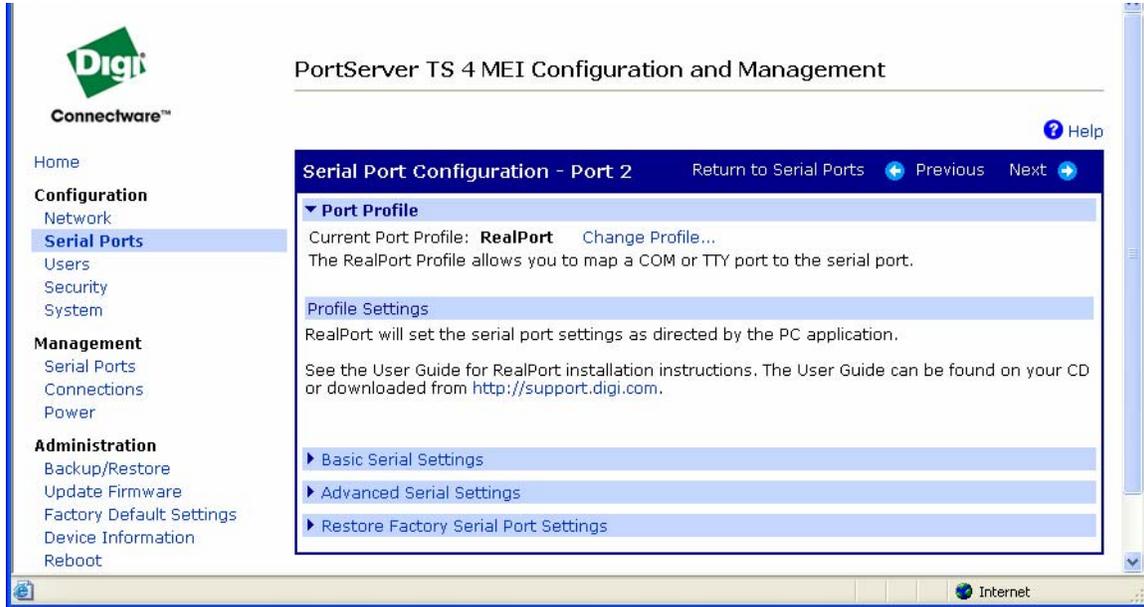
**Figure 1 - RealPort Connections**

RealPort works with all Digi device and terminal servers. RealPort has five distinct advantages over other port redirectors:

1. Only one TCP socket connection is used per device or terminal server. As shown above, all 16 ports communicate over *one* socket. Other redirectors use one socket connection per port.
2. RealPort has virtually no limit on the number of TTY or COM ports supported. The limiting factor is usually the number of ports supported by the application.
3. Port communications parameters are controlled by the application. That is, once the Digi device has been configured for RealPort no further port configuration is required.
4. RealPort is WAN friendly, meaning it communicates well over slower WAN speeds and has robust reconnection capabilities.
5. RealPort is optimized for low latency.
6. RealPort now supports encryption on certain PortServer TS models.

Setting up a PortServer or Digi One for use with RealPort is simple. Via the WebUI select Serial Port > Port 1 (or whichever port is the first port to be used with RealPort) and select the RealPort profile. Use the Copy function to copy the RealPort profile to other ports. If all ports are not used for RealPort, it is recommended to leave one port set to Terminal for local console port access to

the TS.



**Figure 2 - Configure a Port for RealPort**

No other changes are needed. The application controls the serial port parameters such as bit rate, parity and flow control.

## TCP Socket Connections

Most new and some older applications support direct port communications via TCP sockets. Instead of using a COM or TTY port, the SCADA application communicates DNP data directly to the terminal server serial port via:

- The IP address of the terminal server
- A TCP port number associated with the physical port on the terminal server

For example, Port 1 on a PortServer TS would be accessed via TCP Port 2101; Port 2 uses port 2102; port 3 uses 2103 and so on. These TCP port numbers are derived from the base socket of the Digi device, which by default is 2000.

[Note the default TCP port for DNP is 20000. Most DNP SCADA applications allow this to be changed when creating TCP socket connections. If not, you can still use a *single*-port Digi device server like the PortServer TS 1 by setting the base socket to 19899, or use COM/TTY port redirection via RealPort.]

Serial DNP polls are encapsulated into TCP by the application, sent to the Digi device or terminal server, and un-encapsulated at the serial port. DNP responses are handled in reverse order.

Here is an example diagram using TCP sockets:

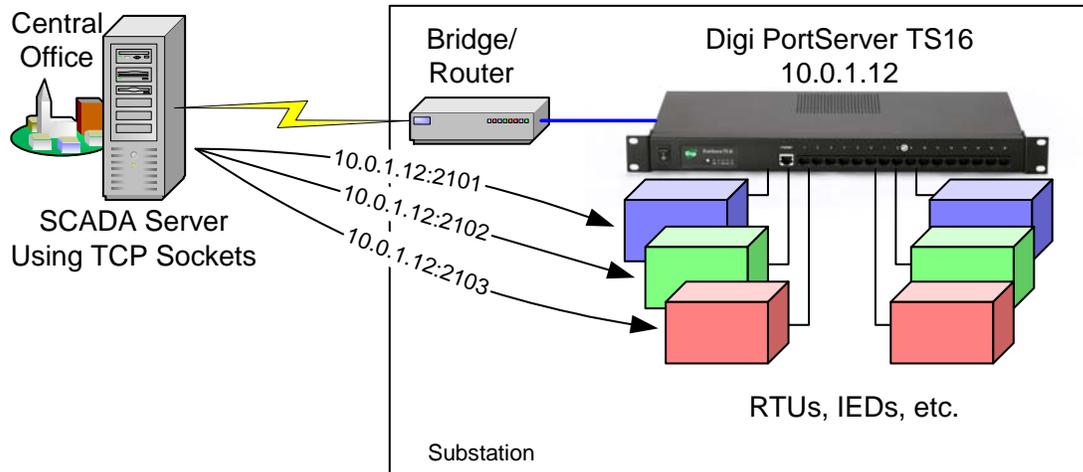


Figure 3 - TCP Socket Connection

Applications supporting socket connections do not require RealPort. However each port does require one TCP socket which may not be as efficient as using RealPort.

These steps are used to configure a Digi port for socket connection via the WebUI:

1. Login into the Digi's WebUI using a standard web browser that supports frames
2. Select "Serial Port"
3. Select the serial port to configure
4. Select "TCP Socket" Profile and click Apply
5. Leave the default setting of TCP Server

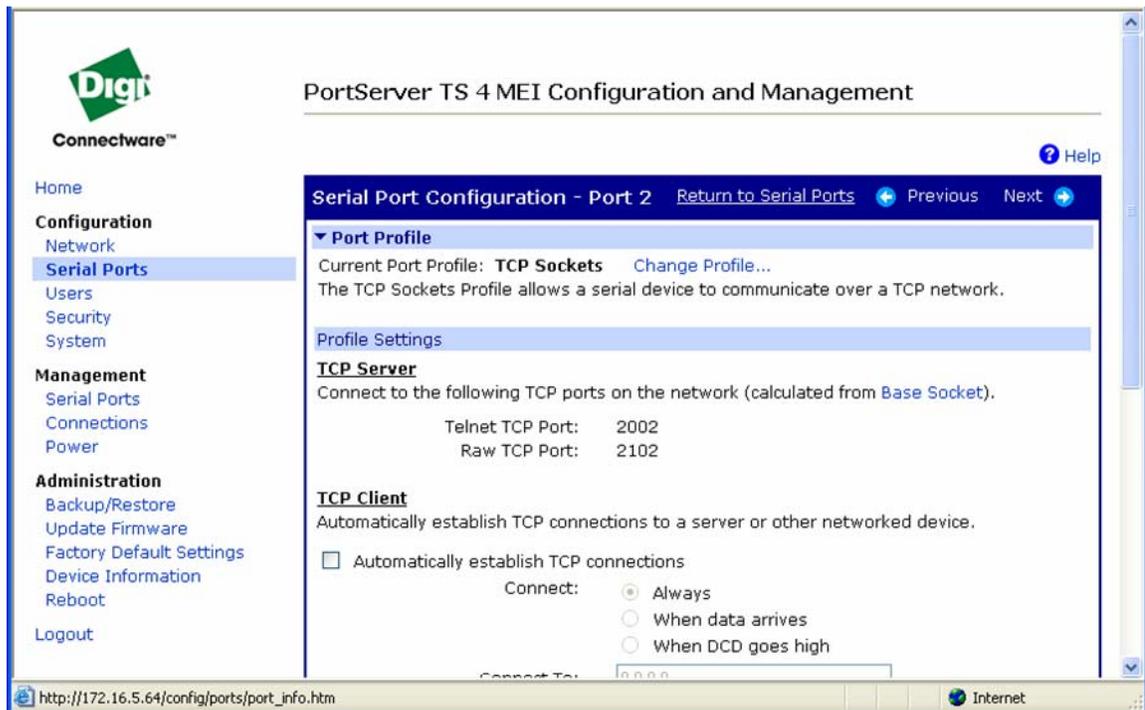


Figure 4 - TCP Server Setting

6. Scroll down and select "Basic Serial Settings". Enter the proper serial port parameters and click Apply.

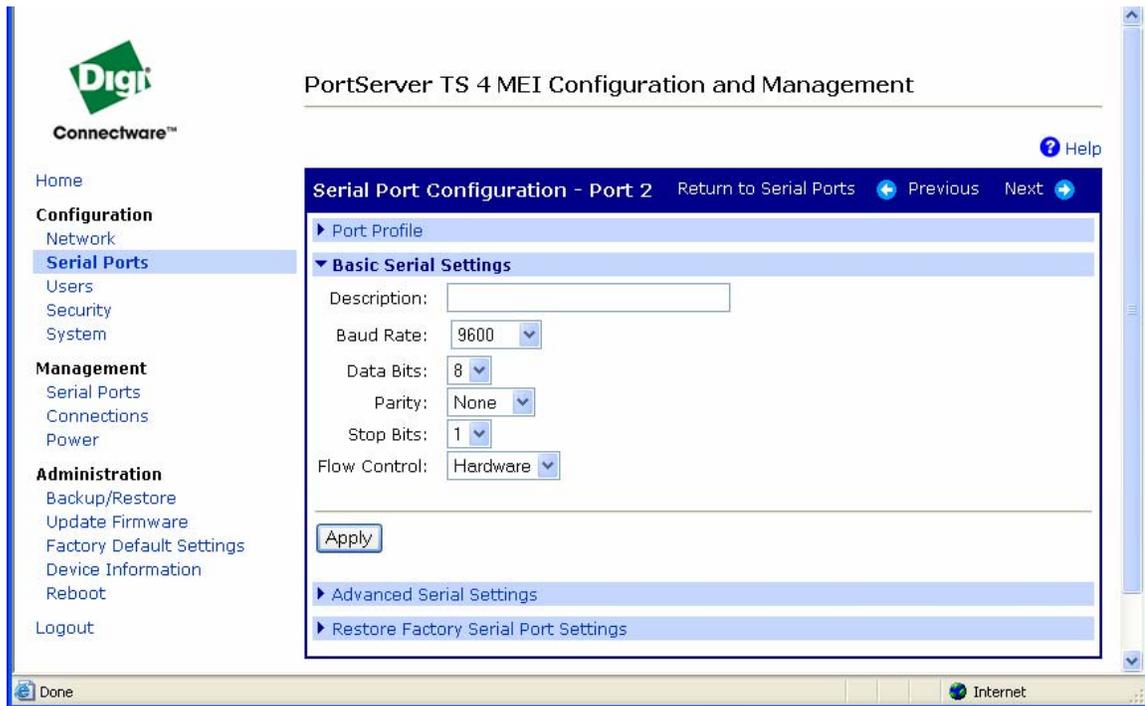


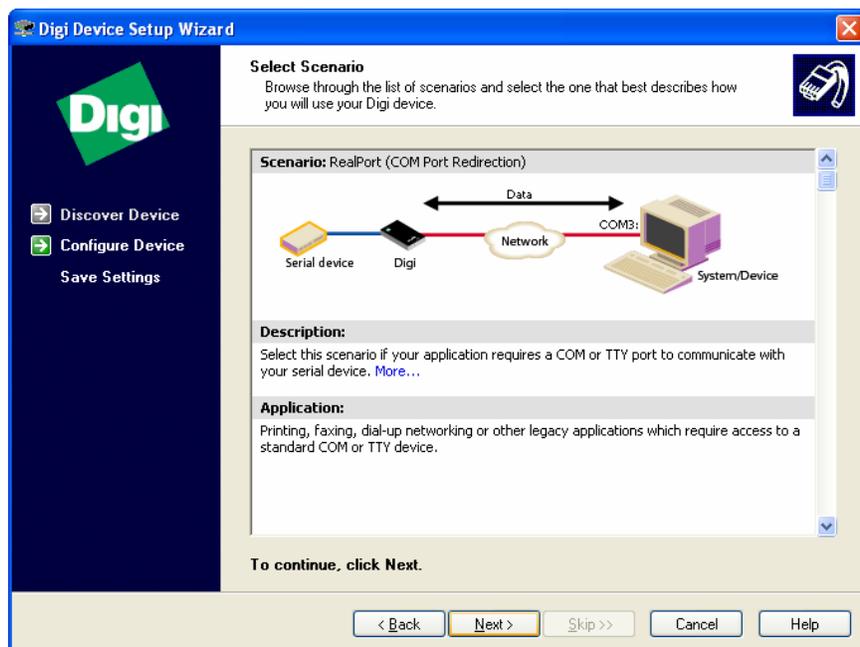
Figure 5 - Serial Port Settings

7. Click "Return to Serial Ports" and copy this configuration to other ports as needed, or select other ports as above.

As noted above, it is recommended to leave one port set to Terminal for local console port access to the PortServer TS.

### Configuring Digi One and PortServer TS Device & Terminal Servers

Digi makes it easy to find and configure units on a network. The screenshots above show the WebUI. Using Digi's Setup Wizard to initially configure the terminal server is even easier. The Setup Wizard will find all Digi servers on the local network, allow you to choose the one to configure, optionally assign it an IP address, and determine a profile for one, all or select ports. It will then send the configuration to the unit and reboot it. The Digi server is then ready to use. A sample screen



shot is shown for a RealPort configuration.

Configuration changes or setting advanced options are done via the WebUI. UNIX and Linux users may prefer to use the full command line interface available via telnet, SSH or direct serial port connection.

Digi One and PortServer device and terminal servers range from one- to 64-ports. 802.11b wireless and hardened (i.e., extended temperature range) units are available. RealPort support is available for a plethora of different operating systems.

Many electric utilities and cooperatives have proven Digi terminal and device servers work reliably with DNP. Please contact your local Digi reseller or Digi directly at 877-912-DIGI or [www.digi.com](http://www.digi.com) for more details.

### FAQ

Q: Does Digi support DNP3 over Ethernet / TCP/IP?

A: Yes. See the explanations above for details. Native DNP over TCP support is not required to use Digi device and terminal server serial ports.

Q: Are other protocols such as Modbus supported?

A: Yes. Like DNP, most serial protocols work fine using either RealPort or sockets or a serial tunnel (where two Digi servers are used to connect serial ports via TCP/IP). For Modbus, some models of Digi device servers provide additional functionality such as Modbus/TCP to Modbus RTU/ASCII serial bridging and multi-master support (i.e., multiple masters can access one Modbus slave).

Q: Do Digi device and terminal servers support UDP?

A: Yes.

Q: What about security? Are RealPort connections secure? How about TCP socket connections?

A: Many utility networks are private, some are secured by VPN. Digi is has added security in these ways:

RealPort: Some versions of RealPort and device/terminal servers support encryption. More work is being done on this and support is expanding. Check with your Digi rep on latest developments.

SSH: Many Digi device/terminal servers support Secure Shell (SSH) Server. If an application supports SSH client, then remote serial ports can be accessed securely via SSH.