GE MULTILIN ™

EnerVista PMCS 6.15
Quick Start Guide

Table Of Contents

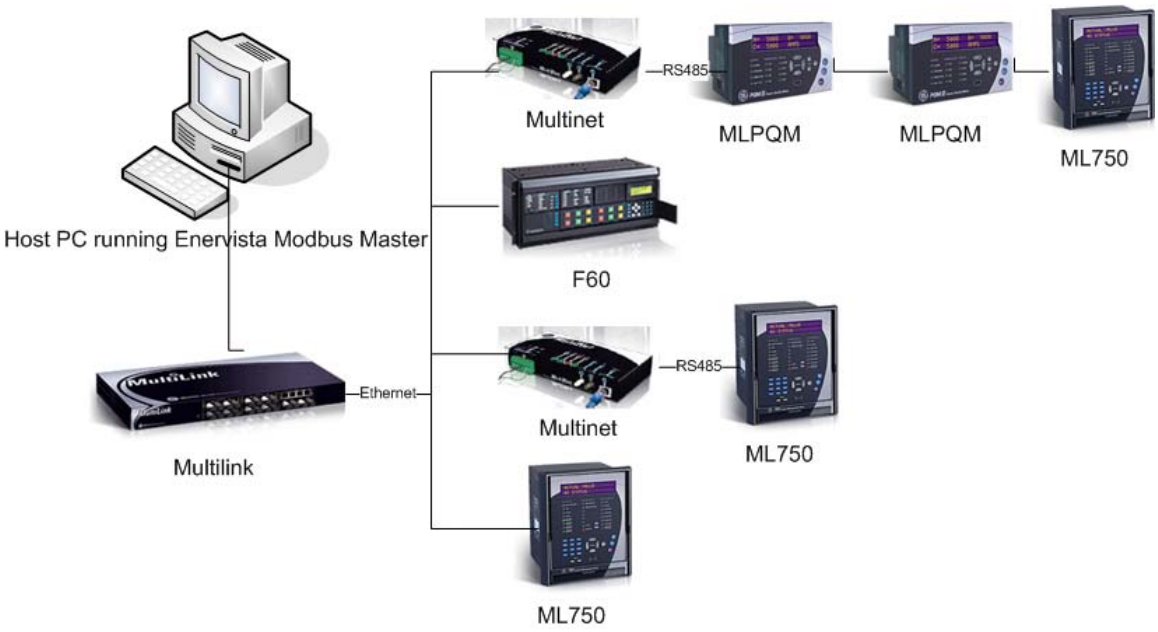
Introduction

Welcome to the Enervista PMCS 6.15 Quick start Guide. The intent of this guide is to quickly walk a user through the basic configuration and operation steps in order to view data from a GE Multilin device. It is not intended to illustrate or explain advanced features and configurations such as adding generic Modbus devices or creating/using custom one-line screens/wizards. Questions of this nature should be addressed in the Enervista PMCS 6.15 documentation found in the CD/Tech_Doc/GE_CD.

By the end of the guide a user should be able to do the following

- Add a GE Multilin Modbus RTU/TCP/IP device to PMCS 6.15.
- View device generated Events/Alarms from the device
- View waveforms from the device (if supported by device)

Below is a simple system combining all the base elements of a PMCS 6.15 system comprised of the following elements: Modbus RTU and Modbus TCP/IP devices, Multinet Gateway, Multilink Ethernet switch and the host PC running Enervista PMCS.



System Requirements

Minimum system requirements:

1 GHz processor
512 MB RAM
Video capable of displaying 16-bit, 1024x768 resolution
700 Mb of available space required on system drive, 40GB of available space required on installation drive
Internet Explorer 5.5 with Service Pack 1 or greater
Windows 2000 SP4, Windows XP SP2 or Windows 2003 Server R2
IIS Installed (on PMCS Server for Remote Viewing capabilities)

Recommended system requirements:

2+ GHz Processor
1GB RAM
Video capable of displaying 16-bit, 1280x1024 resolution

IEDs Supported - Summary

The following devices are supported in the Modbus Servers in this release.

Multilin Industrial IEDs	Multilin UR Family IEDs	GE IEDs
239	B30	Spectra MicroVersa Trip
269+	C30	Enhanced MicroVersa Trip C
369	C60	Enhanced MicroVersa Trip D
469	D30	EPM1000
489	D60	EPM2000
735/737	F35	EPM4000
745	F60	EPM5000P
750/760	G30	EPM5200P
F650	G60	EPM5300P
MMII	L60	EPM6000
MMIII	L90	EPM5350P
MIF2 *	M60	EPM9450Q
PQM	N60	EPM9650Q
PQMII	T35	PLEPM
RRTD	T60	MX150
SPM		MX250
		SMR2
		MPRO
		FIRETRACER
		UPSLP
		UPSSG
		VERSAMAX

Supported Firmware Versions

IED Product	Versions Supported
239	2.3x to 2.7x
269+	6.0x
369	1.6x to 2.3x
469	2.5x to 5.0x
489	1.3x to 4.0x
735/737	1.5x
745	2.4x to 5.0x
750/760	3.6x to 7.0x
F650	1.6x to 2.20
MMII	4.0x to 5.1x
MMIII	1.0x to 1.2x
MIF2	1.0
PQM	3.3x to 3.6x
PQMII	1.0x to 2.2x
RRTD	1.4x, 1.5x
SPM	2.0x, 2.1x
B30	2.6x to 5.0x
C30	2.6x to 5.0x
C60	2.6x to 5.0x
D30	3.0x to 5.0x
D60	2.6x to 5.0x
F35	2.6x to 5.0x
F60	2.6x to 5.0x
G30	4.4x to 5.0x
G60	2.6x to 5.0x
L60	2.6x to 5.0x
L90	2.6x to 5.0x
M60	2.6x to 5.0x
N60	3.4x to 5.0x
T35	2.6x to 5.0x
T60	2.6x to 5.0x

IED Product	Versions Supported
Spectra MicroVersa Trip	5.1x
Enhanced MicroVersa Trip C	4.1x
Enhanced MicroVersa Trip D	4.1x
EPM1000	3.8x
EPM2000	1.0
EPM4000	3.8x
EPM5000P	2.4x
EPM5200P	2.4x
EPM5300P	2.4x
EPM5350P	2.4x
EPM6000	1.0
EPM9450Q	2.1x
EPM9650Q	2.1x
PLEPM	1.0x
MX150	5.4x, 6.0x
MX250	5.4x, 6.0x
SMR2	1.0
MPRO	1.0
FIRETRACER	1.0
UPSLP	1.0
UPSSG	1.0
VERSAMAX	1.0

Note:
UR has been categorized into three different Device Types.

UR - supports 2.6x to 4.6x
UR480 - supports 4.8x
UR490 - supports 4.9x and 5.0x

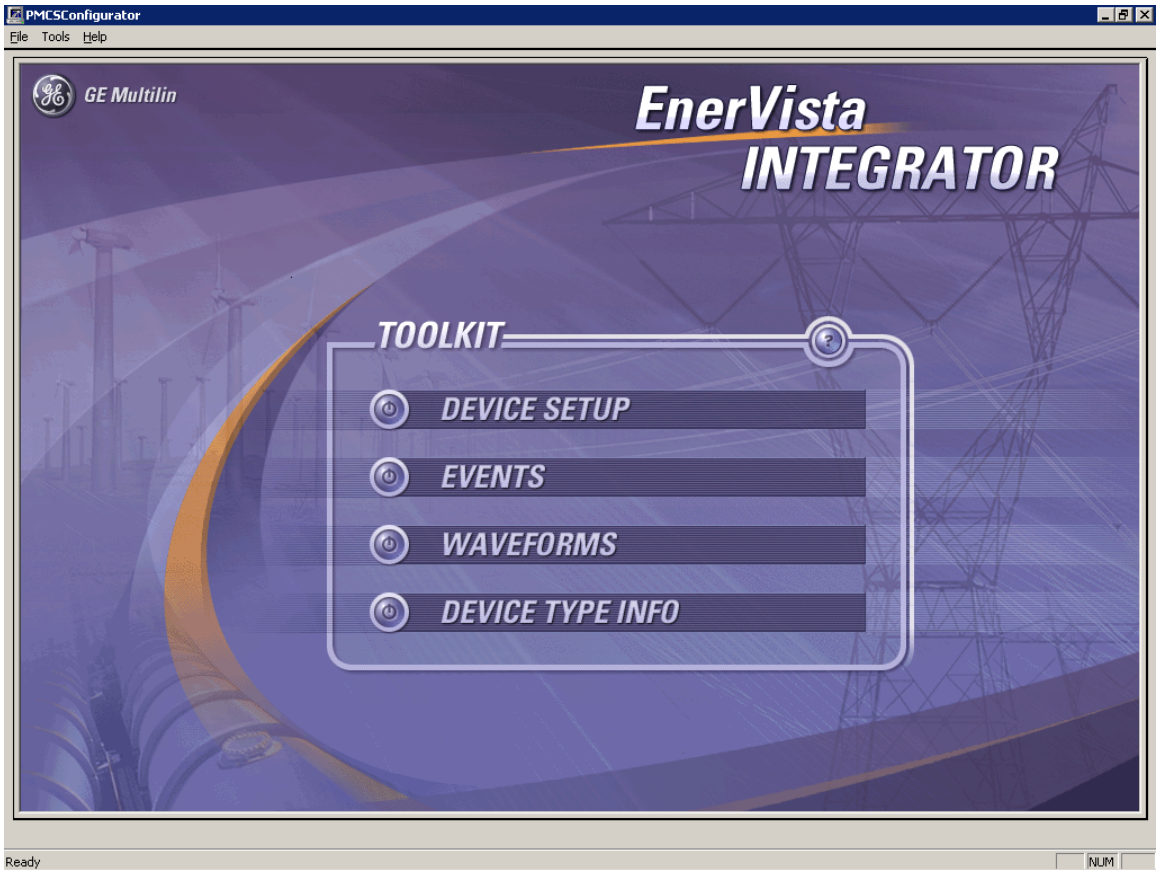
Basic steps to configure a system

Below is quick run thru of the basic steps to configure a system:

1. Install all the required hardware. This included any GE devices such meters, relays, breakers, etc.
2. Install and verify all required communication hardware and wiring. All serial, Ethernet, fiber (if used), etc. should be 'rung' out to verify their correct installation.
3. Configure all device communication settings. Modbus address, baud rate, IP address, etc. need to entered and recorded.
4. Add devices to Enervista PMCS using the **Device Setup** procedure below.
5. View any device-generated events using the **Event Viewer.**
6. View any automatically device-generated waveforms using the **Comtrade Viewer.**

Introduction to EnerVista PMCS 6.15

Welcome to EnerVista PMCS 6.15, a PC-based program for on-demand monitoring and control. After successful installation of PMCS 6.15, the first item to launch is the EnerVista PMCS 6.15 Configurator via the screen shortcut or the Start > Programs menu. You should now see the following main menu:



Here's a breakdown of the main tools of the EnerVista PMCS 6.15 Configurator:

Toolkit Group

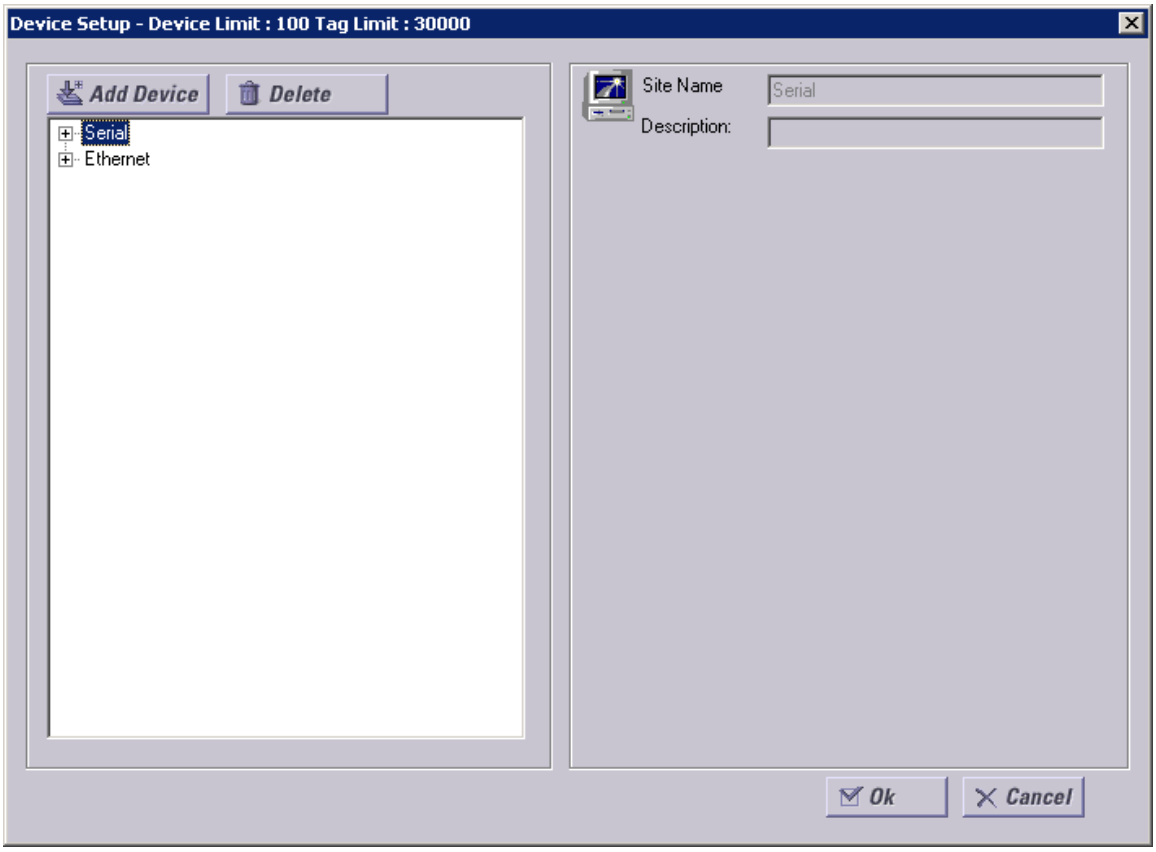
Device Setup - Setup device names and communication to Modbus devices and their components.
Waveforms – Client for viewing CSV formatted waveform files and Viewer for COMTRADE formatted waveform files.
Events- Can view device Events from devices configured in 'Device Setup'.
Device Type Info – Advanced options for adding Generic Devices and modifying the Mnemonics and Register groups for supported devices.

DEVICE SETUP

Toolkit group - Device Setup

Introduction to Device Setup

The Device Setup tool is used to add/edit/modify GE IEDs. The maximum number of devices that can be configured is 100. Upon accessing the Device Setup screen from the EnerVista PMCS Configurator main menu, the following screen will pop up, showing sites (Ethernet and Serial) or devices that you have previously set up:



To save your changes before exiting back to the main menu, click on the **[OK]** button. To exit back to the main menu without saving your changes, click on the **[CANCEL]** button.

Add Device – For adding devices to the system
Site Name – 2 fixed names and is either "Ethernet" or "Serial"

> "Ethernet" – reserved for devices using Modbus TCP/IP over Ethernet, fiber, etc. Examples are devices with a Modbus TCP/IP option such as the UR, EPM 9650, or serial devices using a Modbus TCP/RTU bridge, such as the Multilin Multinet.

"Serial" – reserved for devices using Modbus RTU accessed via PC COM ports connected to a RS232/485 line. Examples are the PQM, SR750, MicroVersaTrip, etc. Devices accessed via Ethernet Port Servers such as the GE Power Leader Ethernet Gateway are considered serial, as the port server serial ports are mapped as COM ports in the PC.

Adding a Device

To add a new device, perform the following steps:

A device can only be created within an existing Site (Ethernet or Serial). To begin, click on the Site type that you wish to add the device to and then click on the "Add Device" button. Next fill in all the parameters below for that device.

Common Device Parameters

Device Name – Username associated with the device being configured. The name must start with a letter and contain no special characters or spaces. If a space is required, the use of "_" is supported. Names should be short, but descriptive.

Interface – Ethernet or Serial selection. Select the interface the device is on.

Waveform and **Events** – Enables auto retrieval of waveforms and events from those devices supporting these functions. Enabled by default for those devices supporting these features. When these functions are enabled, the device will be continuously polled for new device events and waveforms. If a new event or waveform is found in the device, it will automatically be downloaded by the system. New events will be displayed in the Event Viewer. New waveforms can be viewed with the Waveform Viewer or Waveform Client .

Slave address – Modbus Slave address for device. Multiple devices on the same COM port or IP address (when using a Multinet) must have unique slave addresses.

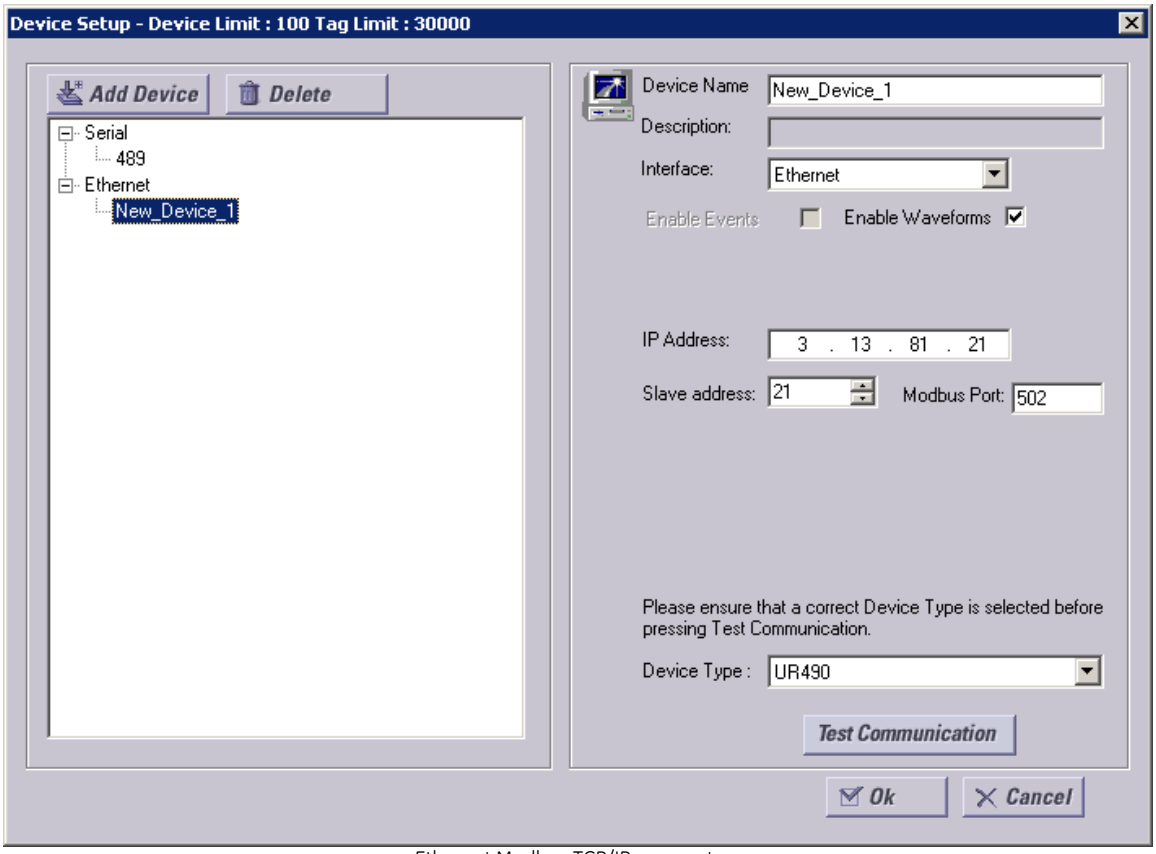
Device Type – Pre-defined Devices can be seen in the drop down menu. Device Types are the same in both Ethernet and Serial Interfaces.

Ethernet Site Only parameters:

IP Address – TCP/IP address of device or Multilin Multinet Modbus Bridge. Each IP address is processed with its own Modbus Master process. When using a Multinet gateway, be aware that the number of devices on the RS485 segment dictates the update performance. The more devices, the slower the data updates.

Modbus Port - TCP/IP port used by device for communication. The default Modbus TCP/IP port is 502. Different ports may be used if network conditions dictate it. Both the device port and the configured port above must match.

Device Modbus slave addressing considerations when using a Multinet or Modbus Bridge: *Modbus RTU devices configured with a Multilin Multinet are configured in the Ethernet Site. Each device will have the same IP Address (that of the Multinet) and Modbus Port,* ***but each device on the same Multinet must have a unique Modbus address.***



Ethernet Modbus TCP/IP parameters

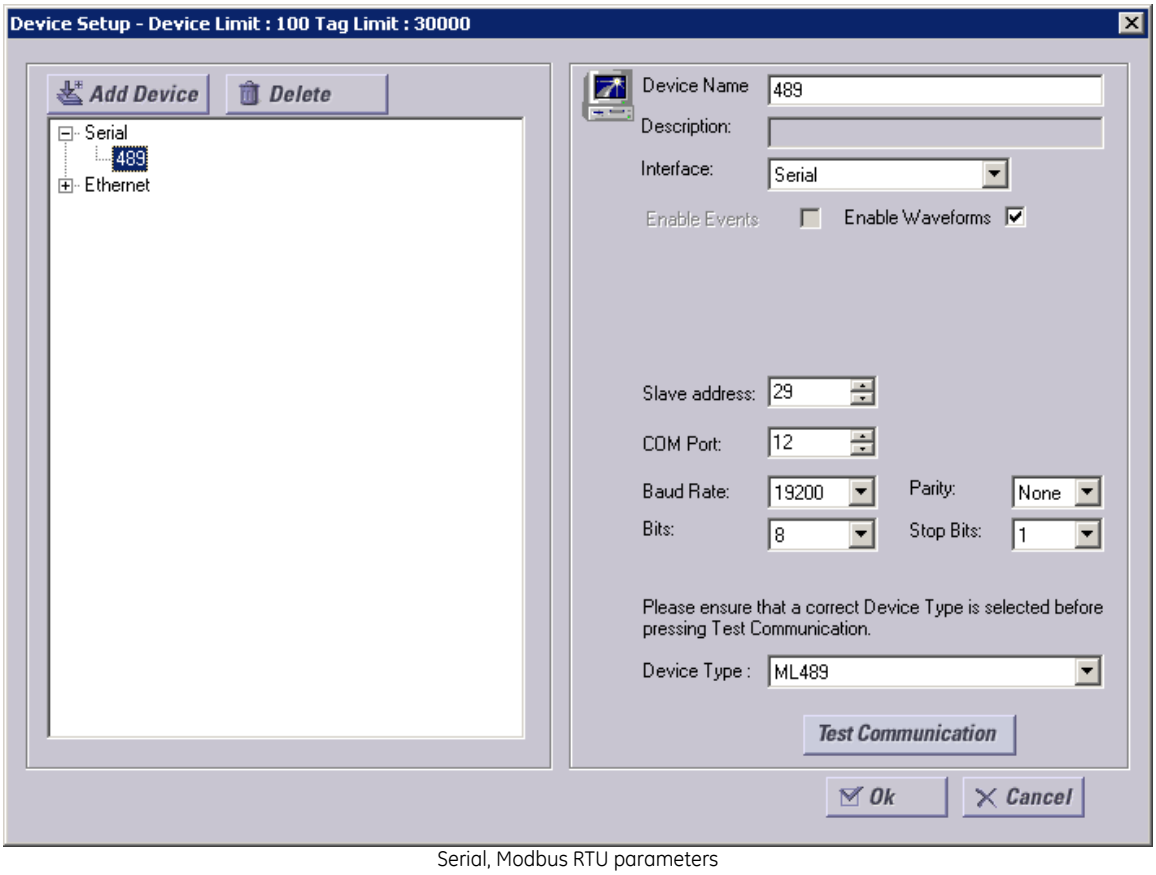
Serial Site Only parameters:

Serial Port – Enter PC COM port used to communicate to device. Be aware that the number of devices on the RS485 segment dictates the update performance. The more devices, the slower the data updates.

Baud Rate – Enter Baud rate for the COM port selected. Baud rates for the COM port and all devices on the COM port must match.

Parity, Bits, Stop Bits – Must match device and COM port settings for the COM port and all devices on the COM port.

Note: Native modem support is not available with the current version of EnerVista PMCS 6.15.



Serial, Modbus RTU parameters

Returning to the Main Menu:

Should you want to cancel your changes made in this session, you can click on the **[CANCEL]** button to exit back to the EnerVista PMCS Configurator menu without saving. If you wish to confirm the changes you have made in this section, click on the **[OK]** button.

EnerVista PMCS Configurator would automatically launch all the Server applications viz. GE32MTCP, GE32MODB, WFServer and EventServer based on the configurations. By default these are hidden.

Other features of Configurator are :
Import / Export Device Configuration : User can export the existing Modbus Serial and Ethernet Device configurations to a CSV file, which can be edited to add/modify/delete any device configuration through Microsoft Excel. This CSV can be imported to any EnerVista PMCS 6.15 System.
Note : Importing will overwrite the exisiting confuguration in the system.

Device Type Info : This is an advanced feature for adding Generic Devices to EnerVista PMCS 6.15 system. All the pre-configured Modbus Register addresses for supported devices can be found here and any Device Modbus address can be added/deleted/modified in the form of Mnemoc-Register Address pairs. This is feature is described in detail in *DDE_OPC Server (GEH-6510).pdf* document

EVENTS

The Enervista Event Logging system is made up of two components: the EventServer, which collects events and stores them in an on-line database, and the EventViewer, which provides an intuitive graphical interface for viewing, querying, and PMCS the database contents. The Eventserver is only configured and activated for a device when its 'Events' item is selected during device setup. By default it is not activated.

The EventServer is the workhorse, running invisibly in the background it is constantly polling, if the feature is activated, the devices for device events. When an event is reported from a device the event is retrieved and stored in a SQL database.

The EventViewer is easy to use. It provides a friendly Windows interface, with toolbars and pull-down menus for quick and easy selection of many pre-configured views of your data. There are also provisions for creating custom views of your data as well.

Please see the *PMCS Event Logger (GEH-6512).pdf* manual for more details.

Alarms and Events

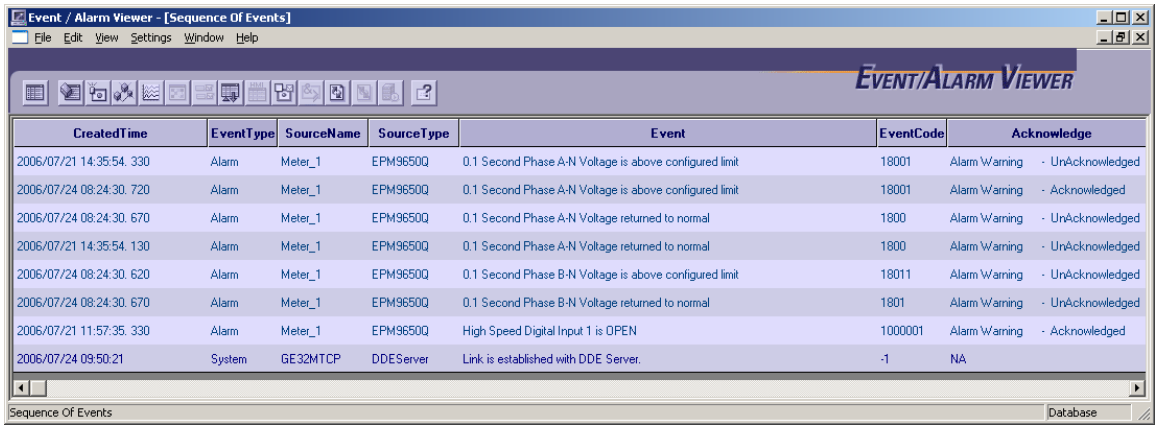
If a received device event is off sufficient high priority it is defined as an Alarm. An Alarm is the highest priority message and indicates a condition in the network, such as a circuit breaker tripped, a circuit breaker in pickup, or a set-point threshold crossed.

Events are lower-priority information that usually indicates a status change, such as a notification that the requested waveform capture has occurred, or an internal communication error with a device has been detected.

All device generated event definitions can be reassigned from their default definitions as Alarms or Events. Please see the *PMCS Event Logger (GEH-6512).pdf* manual for more details.

What's on the EventViewer Screen?

To launch the EventViewer select the "EVENTS" category in TOOLKIT area of the main window. After you click 'EVENTS', the main window appears, as shown below:



You can open additional child windows, but there must always be at least one window open for correct data transfer to the database.

Event viewer can display the Events with different search criteria pre-built as commands, viz.

- Device Events – a specific device related events
- Waveform Events – all waveform server related events
- Alarms - all alarms
- Unacknowledged Alarms – only unacknowledged alarms
- System Events – System (PMCS related) log events
- Annunicator events – all events configured in Event Server for Annunciator Panel

User can create his own Search Criteria to retrieve and display the Events from Database.

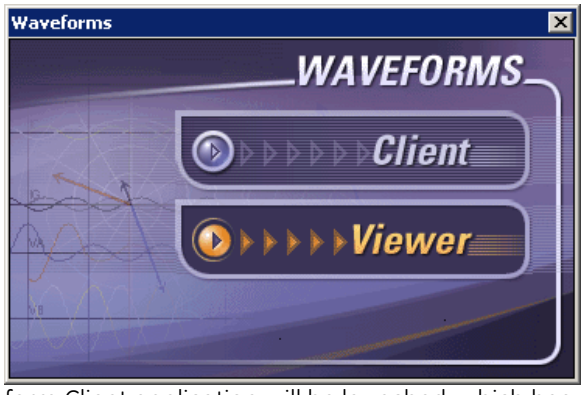
WAVEFORMS

Toolkit group – Waveforms

Waveform Availability

Waveforms are only available from devices that support and are configured for automatic waveform retrieval. Manual triggering of waveforms is supported through Waveform Client application on some of the Supported devices. By default automatic waveform retrieval is not activated. To activate it see the section on device setup. Once activated PMCS will poll the device continuously for any new waveforms. Once a new waveform is detected it will be retrieved. An event will also be generated and displayed in the Event/Alarm Sequence Of Events viewer. To check for new waveforms the user can either look at the Event viewer or check via the waveform viewer.

On clicking **[Waveforms]** on the toolkit, a dialog box appears to selecting the waveform-viewing applications.



On clicking **[Client]**, Waveform Client application will be launched, which has waveform display, harmonic display, Trigger, Upload features. CSV files retrieved by Waveform Server can be viewed in the Waveform Client.

On clicking **[Viewer]**, Waveform Viewer application will be launched, which has waveform display, harmonic display, phasors display, and playback features. Waveform files retrieved by Waveform Server in COMTREADE format can be viewed in the Waveform Client.

Introduction to COMTRADE

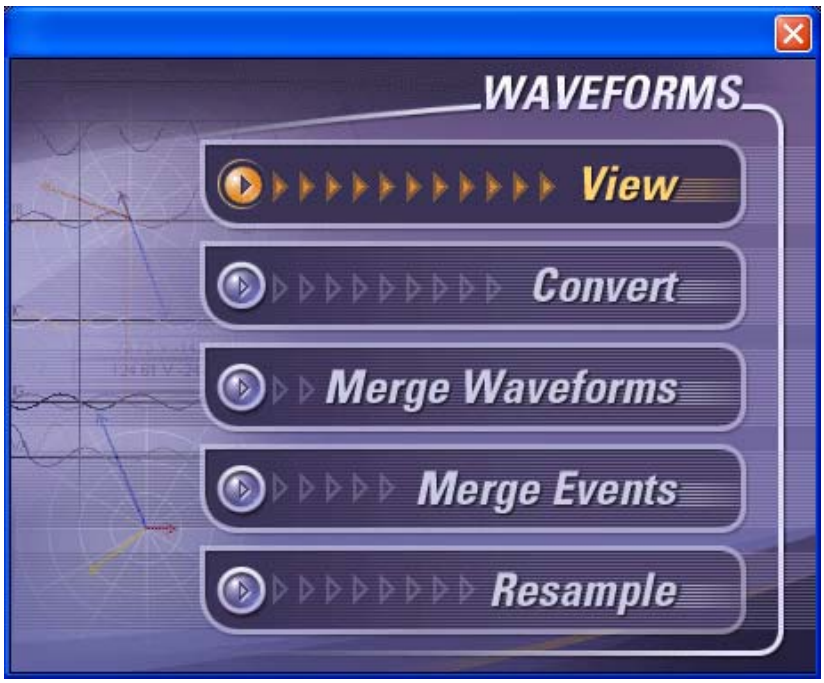
EnerVista PMCS supports the C37.111-1991 and C37.111-1999 COMTRADE Standards. The Comtrade Viewer feature within EnerVista PMCS provides a visual display of power systems data and relay operation data captured during a specific triggered event. The triggering of waveforms is determined by the settings made in the device. There are no default device settings for capturing waveforms, each device MUST be configured with the triggers. Triggers and settings are determined by the device owner.

This introduction will focus only on viewing retrieved waveforms. If there are no waveforms retrieved then the device must be configured to generate one.

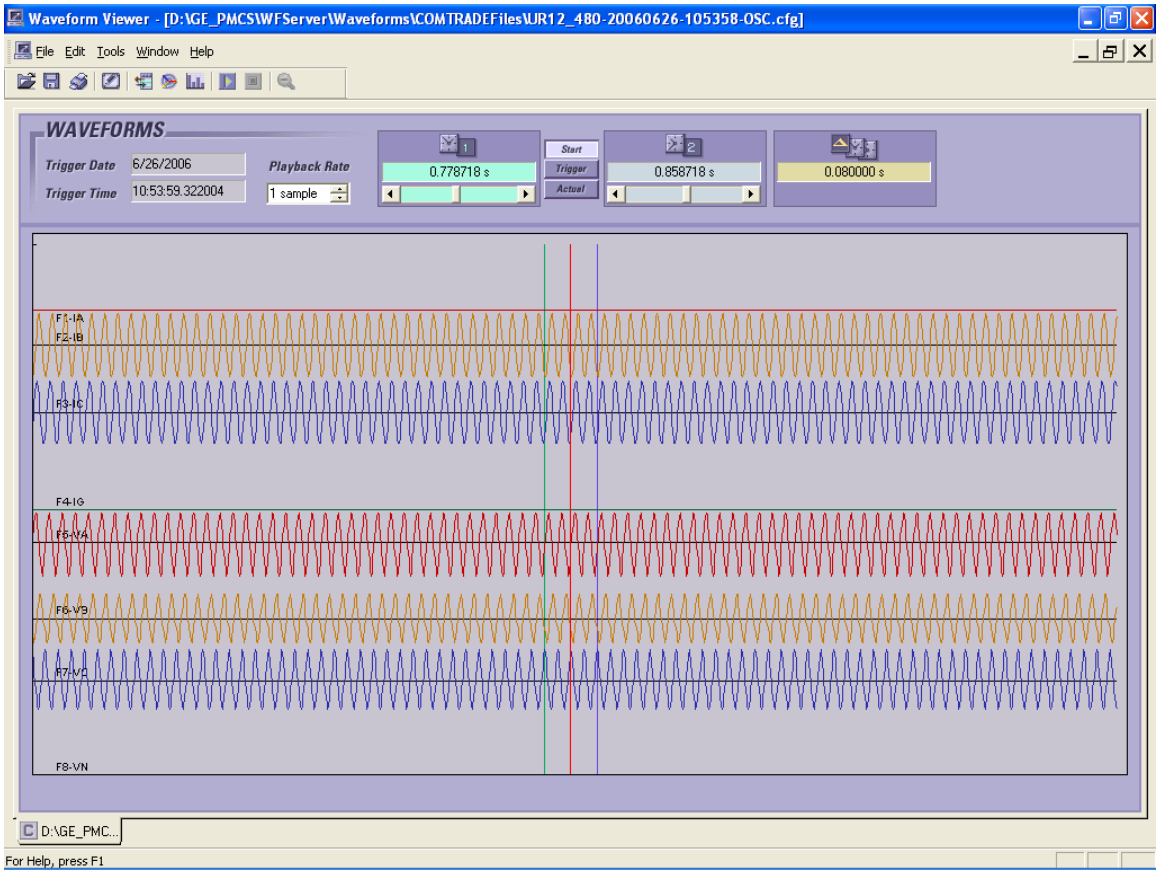
Viewing waveforms of COMTRADE Format

The Waveform Viewer feature within EnerVista PMCS provides a visual display of power systems data and relay operation data captured during a specific triggered event. The viewer application can display oscillography, phasors, harmonics and actual values retrieved from a COMTRADE file.

The following screen shows the options available for viewing and manipulating COMTRADE files.

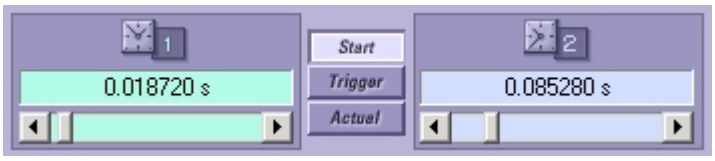


- Under 'AUTOMATED RETRIEVAL' select "View".
- A file selection box will appear. Select the waveform to be viewed. All retrieved waveforms from all devices will be displayed. The file name is comprised of both the device name and the timestamp of when the file was generated in the device.
- Once a waveform is selected the Comtrade Viewer will launch

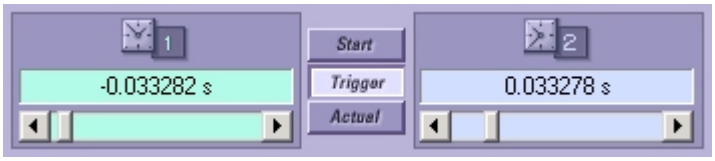


Displayed on the top of the window are the following parameter fields:

- The Blue field (labeled ) displays the parameter value at the blue cursor 2 position.
- The Green field (labeled ) displays the parameter value at the green cursor 1 position.
- The Yellow field (labeled ) indicates the relative parameter difference between the two cursor amplitude values.



- Select the Start button to have the cursor times displayed as an offset from the 'Start Time' of the COMTRADE file. These times will always be a positive value.



- Select the Trigger button to have the cursor times displayed as an offset from the 'Trigger Time' of the COMTRADE file. A negative time indicates a time that occurs before the trigger and a positive time indicates a time that occurs after the trigger.

- Select the Actual button to have the cursor times displayed as an actual time stamp. The format for the time stamp is "mm/dd/yy HH:mm:ss.ssssss" and the hours are based on a 24 hour system.

- The vertical Red line indicates the trigger position (%) of the data capture.

- The Blue and Green vertical cursors indicate initial and final time.

- The top left of the screen shows the trigger time/date as well as the playback rate. The playback rate is related to the number of samples shown every second and can be increased/decreased in order to improve the resolution of the playback.

- **Cursors** can be moved either by clicking and dragging them left and right across the duration period of the signal, or by using the Cursor 1, Cursor 2 scrolling bars at the top of the window. The Red line is the Trigger position and can not be moved across the captured waveforms. There are three time values indicated in the top right boxes listing the selected time for Cursor 1, Cursor 2 and the Delta value. The delta value is the calculated time difference between Cursor 1 and Cursor 2.

- The **Play** button  may be used to play Cursor 1 throughout the entire period or sections of the waveform(s). The speed of play is controlled by the **Playback Rate** which is by default **1 Sample**. The larger the value the faster the cursor will travel across the signal. (**Note**: Playback rate can be changed on the spot during playback) During playback, the **Phasors** screen and **Harmonic** screen will also be updated accordingly in real time based on the position of cursor 1.

- Press the **Stop** button to halt play. The playback automatically rewinds and repeats when the end of oscillography record is reached.

- A selected section of waveform(s) may be **Zoomed** in on by right mouse clicking and dragging the outline box over the desired segment. Multiple zooms are also possible. When utilizing this function, the Zoom Out button  becomes active for zooming back out.

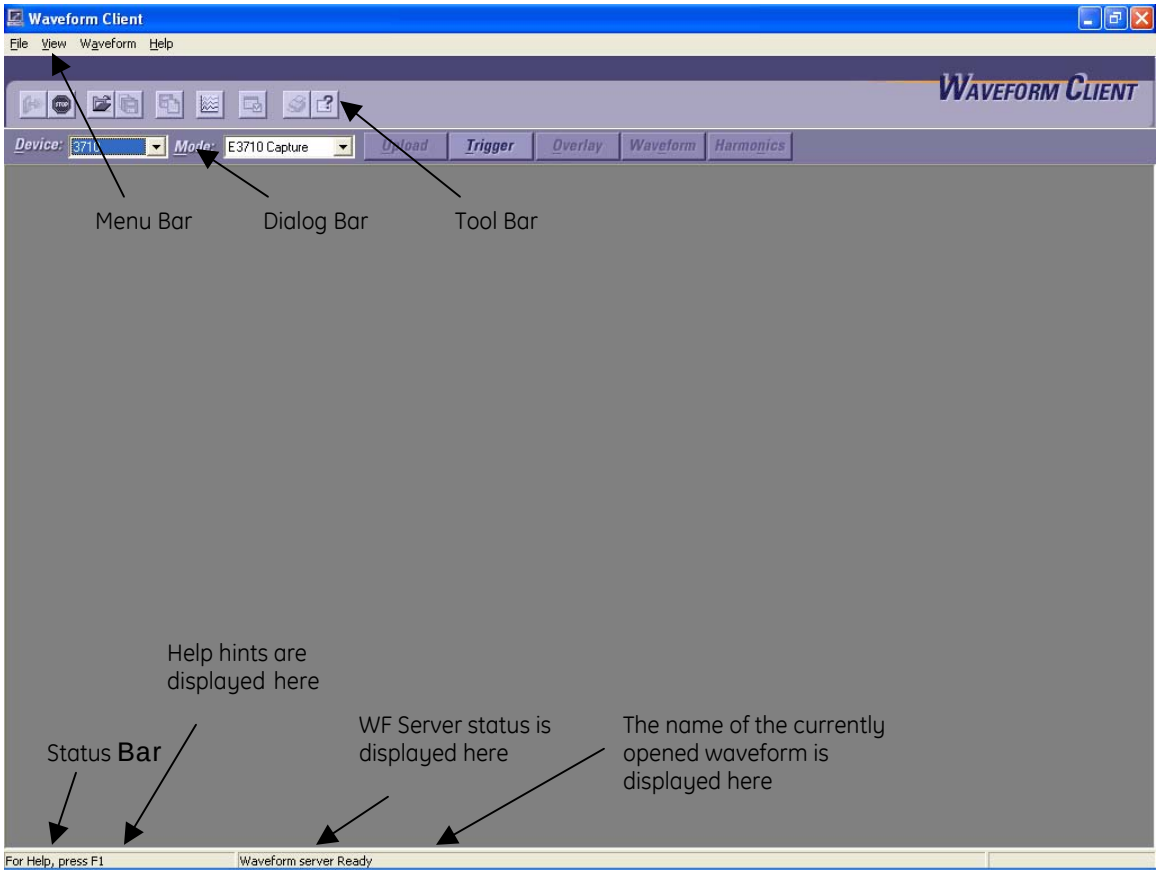
- The **Delta** box on the top right side of the screen indicates the difference in magnitude between the two cursors. The box at the top shows the difference in time (milliseconds).

- Waveforms may be super-imposed on top of each other by selecting and dragging them with the left mouse button.

Please refer to the *Waveform Viewer Help* from the application's main menu for usage details.

Viewing waveforms of CSV Format

The Waveform Client feature within EnerVista PMCS provides a visual display of power systems data and relay operation data captured during a specific triggered event. This is on-line client application for WaveformServer. This application can display oscillography, harmonics from a CSV file.



This application has additional features for Triggering and Uploading waveforms from on-line devices and displays the retrieved waveforms. Also user can select a Oscillography buffer mode for a device while triggering.

User can configure the Waveform Server information to which it has to connect. It can communicate to the configured Waveform Server through DDE/COM interface.

EnerVista System Tray Tool

This is a simple tool for accessing the Server applications, in case user wants. This tool is automatically launched by the system while start-up and this is available always as Tray Icon on the Task bar. By double clicking on this Icon, the tool gets visible.

EnerVista PMCS Configurator would automatically launch all the Server applications viz. GE32MTCP, GE32MODB, WFServer and EventServer based on the configurations. By default these are hidden.

User can make these servers visible by pressing **[SHOW]** buttons against the corresponding application name on the Tray Tool. It's possible to **[Hide]** the applications also.

Tray tool displays the applications' ***Status*** with Color Indicators as well as with text.

RED indicates the application is not running and the ***State*** is displayed as **Closed.**
YELLOW indicates the application is launched and is in **Stopped *State***. Stop condition means the corresponding server application is not on-line with devices.
GREEN indicates the application is started and is on-line with devices.

The **[Show]** and **[Start]** buttons depict the applications state, inherently. When a button is displaying **[Show],** that application can be made visible by clicking on it and the button text will change to **[Hide]** automatically. If clicked on the button again, the application will get hidden. Similarly, the **[Start]** button can be used to **[Start]** and **[Close]** the applications.

On clicking on **[Minimize],** the tray tool gets hidden and the tray icon gets active for the same.

Understanding Enervista PMCS Architecture

PMCS 6.15 is designed to communicate with GE Modbus devices. It is comprised of the following components.

Hardware

Modbus devices – Typically GE Multilin devices supported Modbus RTU and/or Modbus TCP/IP (natively or with a Multinet gateway)

Communication network – Either a twisted pair RS-485, RS232, TCP/IP network (fiber or copper) or a combination or all.

PC – Windows based x86 PC

Software

Operating system – Windows2000 SP4, XP SP2 or Windows 2003 Server R2

Enervista PMCS 6.15 software – From the users prespective PMCS appears to be a single software unit but it is actually a suite of software running together.

Two Modbus Server exe's – Polls for Modbus data from appropriate device. The servers will not poll for data on their own. A client (PMCS, wfserver, eventserver, etc.) must request data.

GE32MODB.exe – The Modbus RTU server. This server is used whenever the PC is connected directly to a Modbus RTU communication network via the PC's COM ports. Most typically via a RS232/485 converter

GE32MTCP.exe – The Modbus TCP/IP server. This server is used when communicating to devices supporting a direct Ethernet interface (UR's, etc.) or Modbus RTU devices using a GE Multinet converter.

EventServer.exe – Polls and retrieves the device generated eventlog. The event is placed in a Sql database For a device to be polled it must support an eventlog function and also have the poll enabled in PMCS (see 'device setup' section)

Eventlog.exe – Viewer for events collected by the Eventserver.exe

Wfserver- Polls and retrieves waveforms generated by devices. For a device to be polled it must support a waveform function and also have the poll enabled in PMCS (see 'device setup' section)

WFC.exe – Waveform Client application to view the waveforms retrieved in CSV format by WFServer.exe. It has features to Trigger and Upload the waveforms from supported devices.

WaveformViewer.exe – Waveform Viewer application to view the waveforms retrieved in COMTRADE format by WFServer.exe. It has features like viewing Phasors and Harmonics and Play-back.

Understanding Modbus device update times and system performance

Subject: GE Multilin Modbus TCP/IP/RTU device update times

Applies to: All GE Multilin Modbus devices and most generic (third party) modbus devices

Perquisites: A basic understanding of Modbus devices, polling, Multilin devices and the Enervista product line (and its associated Modbus masters). Since Modbus TCP/IP and Modbus RTU are, from a polling standpoint, identical the term "MODBUS" in this appnote will be used to describe both types. Exceptions to this will be noted.

Objective: There are many misunderstandings about Modbus, both how it operates and how to calculate update times. This appnote will help both the integrator and end user understand the Modbus protocol and help them to calculate update times.

Basic Modbus Operation: The Modbus protocol is a master/slave transaction where a master device generates a query and a slave device generates a response. A slave should never give a response unless queried by the master. The time it takes for the device to answer is the "device response time". The "device update time" is how quickly a specific queried data set is updated in the master for that specific device.

Note: Network baud rates (both serial and Ethernet) account for only a small percentage of the device response time. Internal processing of the Modbus command by the device takes up the majority of the time.

The nature of Modbus is that only one device can be queried at a time on the same network. For Modbus RTU the 'same network' is defined as a specific serial line (RS485/232) and all the devices on that line. For Modbus TCP/IP the 'same network' is defined as a specific IP address (UR, Multinet, etc.) and the IPport associated with that IP address. If the 'same network' has only one device on it than the "device response time" and the "device update time" should be roughly equal. What if there are multiple devices on the 'same network'? If the master needs to query all the devices it must query them one at a time. Let's suppose a network has 3 devices on it (device A, B and C) and the master needs data from all three. To get this data the master MUST do the following:

1. Query device A for data,
2. Listen for the response
3. receives the response from device A
4. Query device B for data,
5. Listen for the response
6. Receives the response from device B
7. Query device C for data,
8. Listen for the response
9. Receives the response from device C

The whole process is then repeated as needed. From the steps outlined above it can be seen that the "device update time" is dependent on how fast the other devices on the same network can respond. Let's suppose that each device has a device update time of 1 second. This means that the master will see fresh data from each device every three seconds (1+1+1=3). The more devices added to the network the longer each device's update time! Ten devices on the same network means an update time of 10 seconds (assuming each device takes 1 second to respond). The complete update cycle on the same network is also called the 'round robin' time. The more devices on a serial line the longer the overall device update time on that line (assuming each device is polled for data).

Modbus TCP/IP vs. Modbus RTU:
Many people assume that Modbus TCP/IP is 'faster' than Modbus RTU due to the fact that it runs on Ethernet or a TCP/IP network. However since Modbus TCP/IP follows the same rules as Modbus RTU the device update performance is almost identical. The major advantage for using Modbus TCP/IP is related to 'ease of transport'. Use of Ethernet is now almost ubiquitous and allows much easier support vs. a traditional RS-485 network. Both hardware and installation of an Ethernet network is much cheaper and can be maintained by a facilities networking group. It also allows flexibility in data access via intra/inter/extranets. Another advantage is that a device with embedded Modbus TCP/IP is configured in the modbus master as a single device on the same network. As explained in the basic operation section above a single device on a single network (serial or IP) has no 'round robin' time associated with having multiple devices on the same network. **Thus a dedicated Modbus TCP/IP device gives the appearance of a 'performance gain' however this is derived solely from the virtue of its 'single device on a network' status. A serial Modbus RTU device would have similiar update performance if alone on a serial line.**

GEMultinet or Modbus TCP/IP/RTU bridge:
With industry moving toward Modbus TCP/IP devices and Ethernet in general there is a need for legacy support of Modbus RTU and RS-485 networks. To support these networks and devices a Modbus TCP/IP/RTU bridge such as the GE Multilin Multinet is required. The Multinet converts the Modbus RTU protocol to the Modbus TCP/IP protocol. Once converted to Modbus TCP/IP the messages are then passed on via the available TCP/IP network (Ethernet, fiber, etc.).

The biggest difference between a dedicated Modbus TCP/IP device (such as the UR) and a Multinet is the fact that the Multinet can have multiple devices daisy chained on its RS-485 port. These multi-dropped devices follow the same rule as other multiple devices on a 'same network'. If each device is polled for data a 'round robin' occurs. The update time is then controlled by the number of devices and how many modbus blocks are needed from each one.

The graphic below illustrates 1 meter and 3 relays. If each device was being polled for registers requiring four separate modbus blocks than each device would take 1000ms to poll (4 x 250ms/block). Since there are 4 devices the total update time would be 4000ms (device1+device2_device3+device4 = 4000ms).

Device update times using Multilin Enervista software:
Enervista can communicate to a wide range of GE device ranging from relays, trip/breaker units, motor controllers and meters. Each device has a unique register map and number of registers. Whenever a client requests data from the Enervista Modbus server the server processes the request

and creates a Modbus request(s) for the required registers. If all the registers required can fit into one Modbus return message (124 registers in sequential order) than only one request for one block of data is made. With only one request needed the GE device update time will be roughly 250ms.

However if the amount of registers are too many for one return message or the registers needed are more than 125 registers apart than the server breaks the request into two or more Modbus requests requiring two or more blocks of data. Devices with small register maps, such as an MVT trip unit , generally need only one Modbus request with a device response time of about 500 Milliseconds (the MVT has a longer device response time than most devices thus it takes 500ms to process and return a request). . But larger devices (MLPQM, UR, ML750, etc.) may require 4, 5, 6 or more Modbus requests with a device response time that could be as much as three seconds. To poll the entire register map of a UR can take over 6 seconds requiring over 24 modbus requests. The device update cycle time is, to a high degree, independent of the physical network. A single device on a 19.2K baud serial line will have roughly the same update time as a single device on Ethernet.

The Enervista Modbus Master supports each network independent of other networks. So what happens on network A does not affect what happens on network B. A network is defined as a specific serial line or IP (and its IPport) address. If network A has only 1 device on it and only one register block is needed than data updates will be 250ms. But if network B has 4 devices on it requiring a single block from each than the 'round robin' time will be 1000ms.

Enervista PMCS, Integrator and PMCS also support polling of the devices Alarm/Event Registers and the Oscillography registers. These registers are constantly polled along with any other device registers. If a new device alarm/event or waveform is detected in the device (this condition is monitored via dedicated registers) the Enervista software then initiates an auto download of the required data. These data is interleaved with the real time data request so the impact on performance is minimized. New alarm/events can be downloaded very quickly but waveforms can take up to 10 minutes to download (it depends on the device and waveform settings).

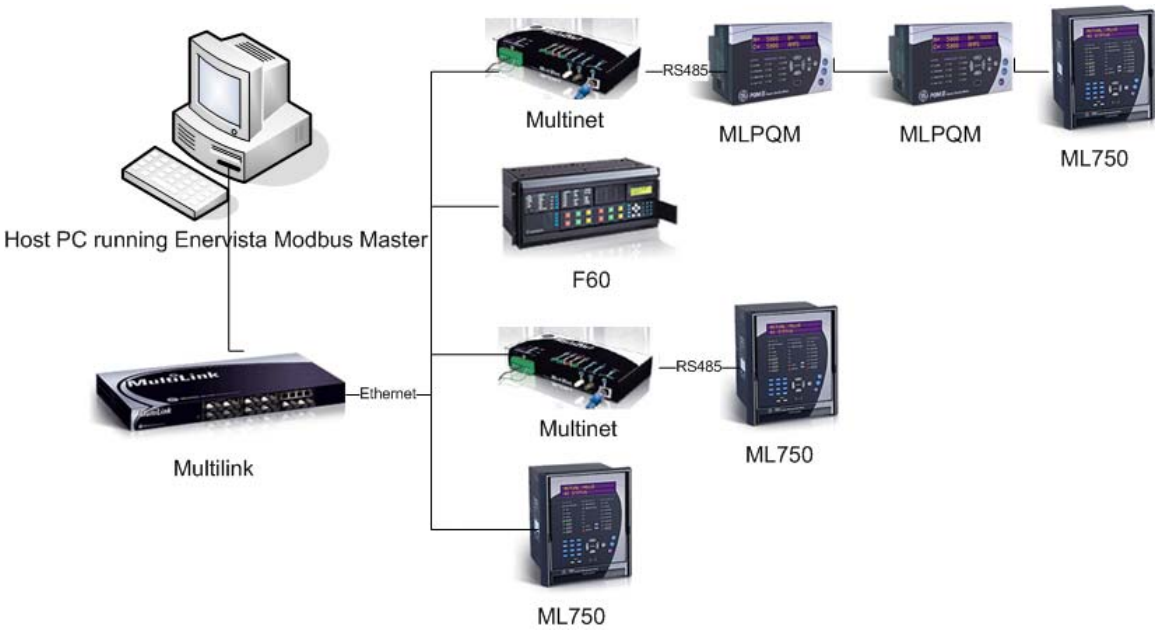
Test Case:

Generic data retrieval:

Configuration:

Network	Device 1	Device 2	Device 3
Multinet 1.1.1.1	MLPQM	MLPQM	ML750
UR 1.1.1.2	F60		
Multinet 1.1.1.3	ML750		
*ML750 1.1.1.4	ML750		

*ML750 with built in Modbus TCP/IP



Data needed from each MLPQM. For this example each PQM will require 5 modbus requests for the required data. Below is a breakdown of the example requests.

PQM Mnemonic	Reg Address (Hex)	Reg Address (Decimal)	
AMPS_AVG	R3X0243	579	Block 1(both within 124 registers)
VOLTS_AVG_LINE	R3X028EL	654	
KW	R3X02F0L	752	← Block 2
KW_DMND	R3X0404L	1028	Block 3 (both within 124 registers)
FREQUENCY	R3X0441	1089	
New Waveform	R3X0B83	2947	← Block 4
New Alarm/Event	R3X0AD0	**2768**	← Block 5

Data needed for the ML750: Assume the ML750 requires 6 Modbus requests for the required data
Data needed for the F60: Assume the F60 requires 8 Modbus requests for the required data
Assumptions: Each block takes 250ms to retrieve

Performance:
PQM – 5 blocks x 250ms = 1250 ms
ML750 – 6 blocks x 250ms = 1500 ms
F60 – 8 blocks x 250ms = 2000 ms

Network	Device 1	Device 2	Device 3	Device Update Time
Multinet 1.1.1.1	1250ms	1250ms	1500ms	4000ms

UR 1.1.1.2	2000ms			2000ms
Multinet 1.1.1.3	1500ms			1500ms
*ML750 1.1.1.4	1500ms			1500ms

Enervista PMCS

Enervista PMCS incorporate device 'wizards' or pre-made device graphics containing data point displays of typically required in a device. These displays can be very data intensive requiring many register calls to a device. This can cause a temporary or 'exception' condition where a device that is normally responding in under a second suddenly begins to update at a much slow rate. This is due to additional modbus blocks needed by the graphics. The graphics work in an 'on-demand' mode meaning the only time data is requested is when the actual display is open. Once the display screen is closed the device update rate returns to normal. For example when a UR tabular screen is opened in a PMCS system the device update time for that UR can suddenly jump from 1 second to 6 seconds as all the required registers begin to get polled. Once the screen is closed the update time then goes back to 1 second.

Please note that all the screens in a Enervista PMCS system typically require multiple modbus requests to populate the screens. Add in the fact that many other devices are typically multi-dropped on the line and PMCS of waveforms and alarm/events yields an update time of 2~5 seconds on a typically device screen. **UPDATES BELOW 1 SECOND CAN ONLY RELIABLY BE ACHIEVED FOR SYSTEMS RUNNING A DEDICATED MASTER REQUESTING A CONSTANT 1 OR 2 MODBUS BLOCKS FROM A SINGLE DEVICE ON A NETWORK. THIS IS A HIGHLY CUSTOMIZED FUNCTION REQUIRING CAREFUL PLANNING AND DESIGN.**

Testing:

For systems testing and evaluation it is frequently impractical or impossible to have a test bed of all the required devices. To this end the Enervista Device Emulator has been designed. This is a Modbus TCP/IP/RTU software slave designed to emulate all the GE Multilin device types, their register maps and real device values both dynamic and static. It will accurately emulate all required network architectures in both size and timing characteristics. With it an integrator or end user can accurately bench test a project for both operation and update times. Virtual network configurations can be created and adjusted/changed before any physical hardware is installed. This software can be purchased thru your local GE Multilin rep.

Conclusion:

Before any network is designed or constructed a careful examination of needed data update performance must be done. Each site and customer is unique. Some customers may require data refresh times in the ten's of seconds while other require less than a second. Any update requirement within the physical confines of Modbus can be done but require correct network layout and a thorough knowledge of what specific data is needed from each specific device type. This appnote has also examined the differences between Modbus TCP/IP and Modbus RTU. Integrators and users must be especially cogent of the fact that use of Modbus TCP/IP and Ethernet **DOES NOT ITSELF INCREASE PERFORMANCE TO ANY NOTICABLE EXTENT**. Much more critical is **HOW MANY DEVICES ARE ON THE SAME NETWORK AND HOW MUCH IS POLLED FROM EACH ONE.**