Wheatstone Corporation
Technical Documentation

WheatNet-IP
Scheduler
Technical Manual
Preliminary V 2.0.0

Table of Contents

1. Introduction To WheatNet-IP Scheduler

The WheatNet-IP Scheduler program is designed to fire Events in a WheatNet-IP system at pre-arranged times.

In order to integrate the software into a WheatNet-IP system, you will need to run it on a computer having a unique IP address on the same subnet as the WheatNet-IP system (typically 192.168.87.xxx). You will also need to know the IP address of at least one Blade in the system, since the software will need to directly address a Blade.

What are events? Simply put, they are actions that will occur within a WheatNet-IP system. The software supports Connect, Disconnect, Salvo, and ACI (Automation Control Interface) events. Events are configured within the software, and are also **fired** by the software. Events act on system **signals** (Connect and Disconnect events) and **salvos** (Salvo events). ACI events direct automation-specific **commands** to various devices in the WheatNet-IP system. Events may be saved to a file, and the saved event files may be loaded into the program. This allows you to create a schedule on one computer, then load and run the schedule on a different computer.

Since a scheduler by its very nature performs tasks at specific times, it is important to note that the WheatNet-IP Scheduler depends on the clock on the PC that the program is running on to know precisely when to fire an event. Thus the PC clock needs to be set to the correct time and date to match its location. Otherwise, events will not fire as expected.

1.1 Events And Signals

As stated above, the software utilizes various event types, signals, salvo definitions, and commands to compose the events necessary to effect the desired actions within the system at the appropriate times. These items will be discussed briefly here.

As previously stated, an event is an occurrence within the system that can be caused to take place at a specified time according to a schedule. Four types of events are supported in the software.

1.1.1 Connect Event

A **Connect** event is one that causes a particular source (input) in the system to feed a particular destination (output) in the system at the scheduled time. Both sources and destinations can be specified by signal Name and Location (which are assigned by the system administrator via the WheatNet-IP Navigator program) or signal ID (which is automatically assigned within the WheatNet-IP system

based on the rules the system imposes to define signals).

It should be noted here that only one source can be connected to a given destination at a time. For this reason, an event that causes a source to be connected to a destination which already has a different source connected will cause the original source to be disconnected prior to making the new connection.

Also worthy of note is that connections in the system may be locked, and that the Scheduler software will not override signal locks.

1.1.2 Disconnect Event

The next type of event is a **Disconnect** event. In this case no source need be specified, only a destination. If a source is currently connected to the specified destination, the disconnect event will break that connection. However, once again, the Scheduler software will not override signal locks.

1.1.3 Salvo Event

The third type of event is the **Salvo** event. Salvos are configured within the WheatNet-IP Navigator program, and consist of sets of connections and/or disconnections that will happen rapidly in sequence when the salvo is fired. Salvos are identified both by Name and ID when constructing a salvo event.

1.1.4 ACI Event

The fourth type of event is the **ACI** event. This type of event is used to send ACI commands to various pieces of equipment within the WheatNet-IP system. The program currently supports sending ACI commands to Blades, Surfaces, and Processors. It is beyond the scope of this manual to list all of the ACI commands available for your use. Examples will be shown, but otherwise please contact Wheatstone Technical Support at 252-638-7000, or email us at techsupport@wheatstone.com, to discuss your specific requirements.

2. Installation

The following instructions guide you through the software installation process.

2.1 Begin Installation

The software is delivered as an installation file named **wheatnetipsched_install_x.x.x.exe**, where **x.x.x** is replaced with numbers indicating the version being installed. Double-click the installer file icon to start the process.

2.2 Accept The License

The first screen that you see in the installer displays the EULA (End User License Agreement). You must accept the license to install the program.
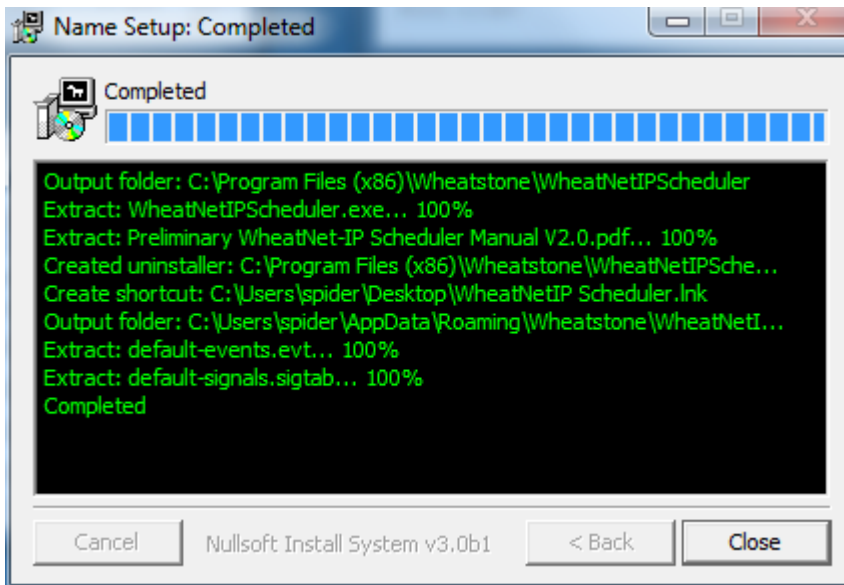
Click the *I Accept* radio button and click *Install* to continue.

2.3 Finish The Installation

You will see a progress screen that will look something like this when the installation is complete:

Click *Close* to finish the installation.

The installer will also add a shortcut to your Desktop for a convenient way to start the WheatNet-IP Scheduler after installation.

2.4 Uninstalling The Program

Should it become necessary or desirable to uninstall the WheatNet-IP Scheduler, navigate to the program's working directory, which will be in the folder **Wheatstone\WheatNetIPScheduler** under **C:\Program Files (x86)** (Windows 7) or **C:\Program Files** (Windows XP). Find the **uninstaller.exe** program and double-click its icon to remove the Scheduler. Or you can use the Windows Add or Remove Programs Control Panel.

3. First Time Operation

First time operation of any software is a learning experience. With the WheatNet-IP Scheduler program, the steps involved in first time operation will differ slightly depending on whether or not you are actually connected on the same network as a WheatNet-IP system.

3.1 If You Are Not Connected To A System

This section will deal with what happens when you start the program and it is not connected to a WheatNet-IP System.

3.1.1 Default Signal Warning

Whenever the program starts it will check to see if it can find a saved signal set. The program installer provides a sample signal set in the directory the program searches by default, and whenever you save a signal set the path to that file is remembered, so you may never see this warning.
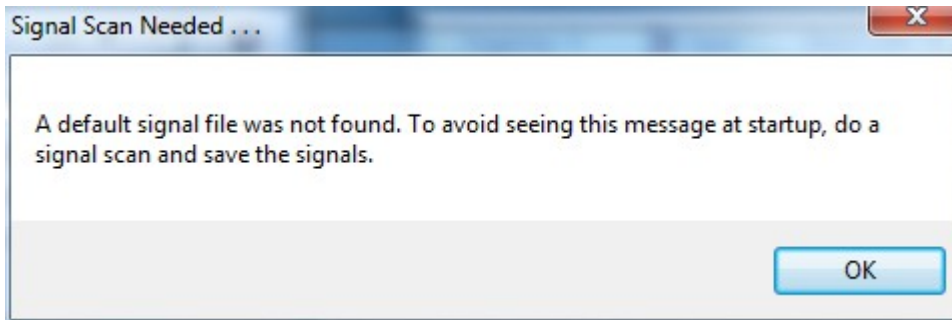
If you do see this warning, it simply means that your program will not be aware of any system signals needed to create events. Later on we will see what steps you can take to rectify this situation, but for the present, if you see this warning, just click OK.

3.1.2 License Warning

The first time you start up the program you will be asked for a License Key. This will also happen every time you open the program during the software's seven day trial period if you haven't yet entered a license key. Here is what the warning looks like on the second day of the trial period:


If you wish to run the software beyond the seven day Trial Period you will need to purchase a site license for the program from your Wheatstone Corporation sales representative.

To continue without entering a License Key click *Later*. Each time you restart the program without having entered a valid License Key you will again be prompted to enter one. After the seven day Trial Period is over the nag screen will no longer give you the option of continuing unless you enter a valid License Key.

3.1.3 License Key Entry

If you have a License Key to enter, click *Enter License Key...* to bring up the following screen:
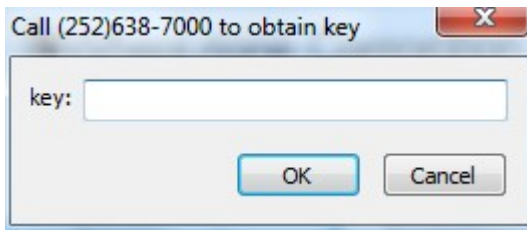
Enter the License Key that you obtain from Wheatstone Technical Support. You can either type the license key in, or right-click in the *key:* field and select *Paste* from the popup menu. Once the key is entered, click *OK*. The following screen appears when a valid License Key has been entered:
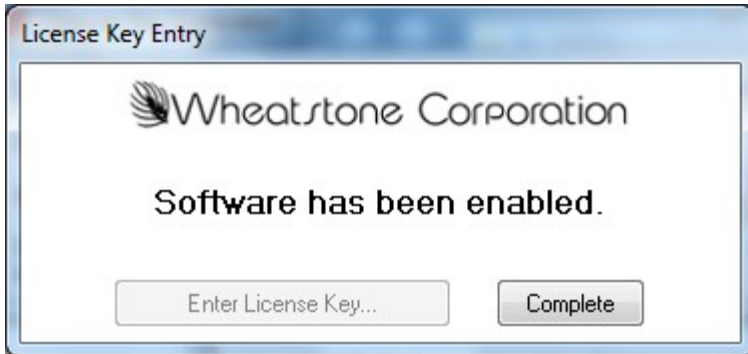
Click *Complete* to continue.

3.1.4 Main Screen

Once you have either entered a key, or indicated you will enter a key later if you are still in the trial period, the following main screen will appear:
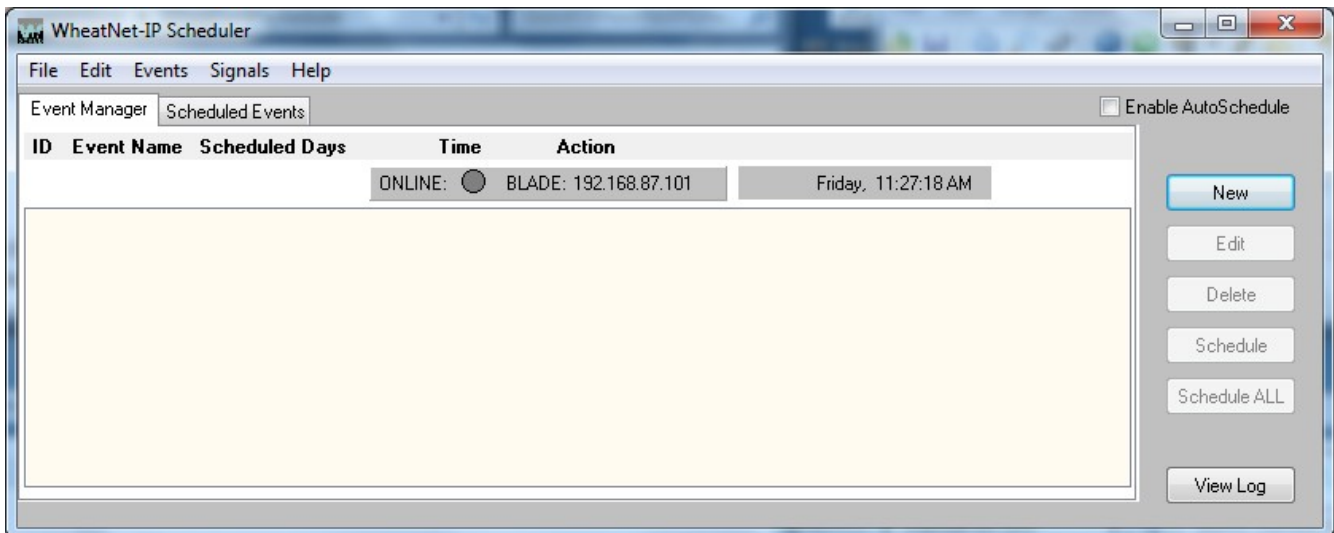
Proceed to section 4 on day to day operation.

3.2 If You *Are* Connected To A System

This section will deal with what happens when you start the program and it *is* connected to a

WheatNet-IP System.

The first few steps will be the same as when not connected to a system. You may encounter the default signal warning, and the first time you run the program you will certainly encounter the license warning, unless you had an older copy of the program already installed and licensed.
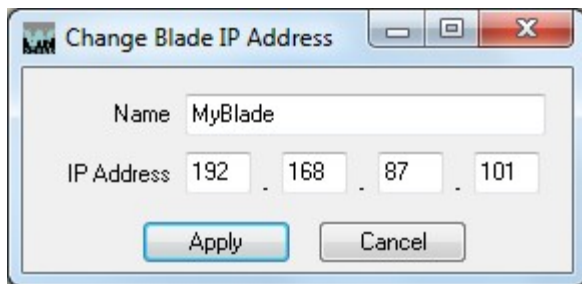
If you are connected to a system which contains a Blade with the ID of 1 and the default IP address of 192.168.87.101, your main screen will look like the one shown earlier, with the exception that the ONLINE: indicator will be green rather than gray.

For the benefit of those who may have a system without a Blade of ID 1, or have their system set up on a different network subnet, the next step will be to change the IP address that the program looks for its host Blade at.
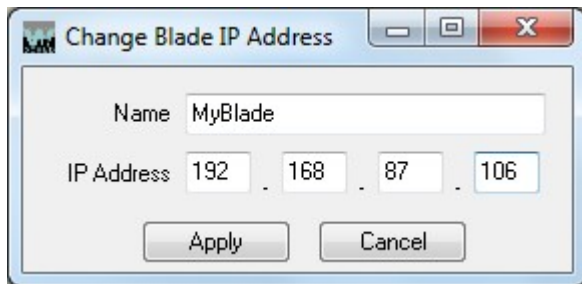
3.2.1 Set IP Address

There are two different ways to change the Blade that the program will connect to.

From the menu, select *File>Set IP Address...* to bring up the following form:


Note that the form indicates the current setting. Change the address information to that of the Blade you will use to host the Scheduler:

For convenience you can type a name in the *Name* field. The name is not actually used.

Once the IP address is entered, click *Apply*. You will be prompted to restart the WheatNet-IP Scheduler Program before your new IP address will take effect.
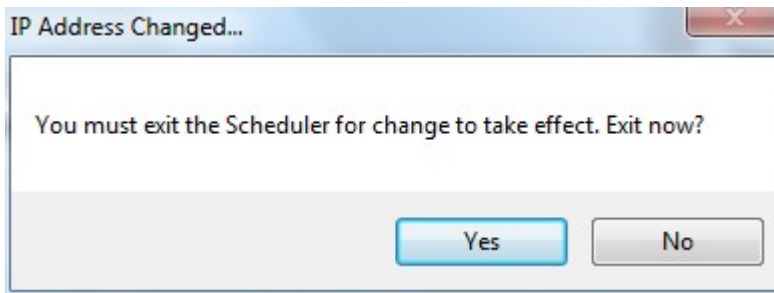
You can click *Yes* to exit immediately, or click *No* if you are not ready to exit. Until you do exit and restart it, the program will still be attempting to communicate with the previously specified Blade.

The other way to change the IP address is to click on the word *BLADE* next to the *ONLINE* indicator.

This will bring up the change dialog, and the rest of the procedure is the same as when started from the menu.

Once you are connected to the host Blade the main screen ONLINE indicator will be green.
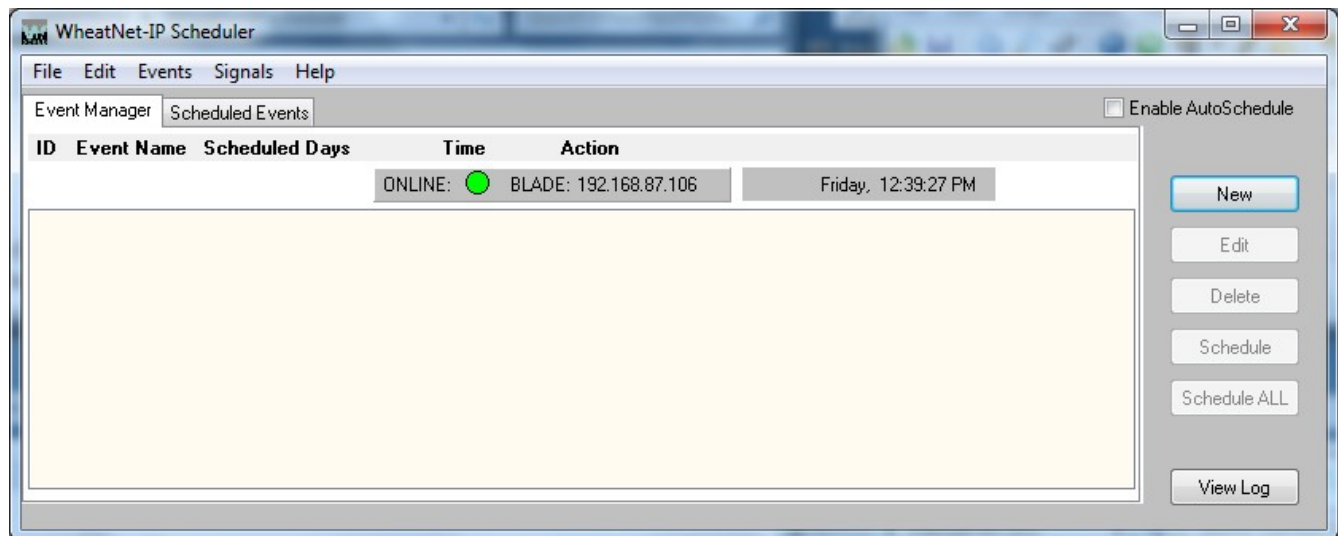
4. Day To Day Operation

As you become more familiar with the program operation you will begin discovering the tasks you want to perform with the software. This section of the manual details the various tasks, what they are for, and how to perform them. For the sake of discussion, tasks are organized by category: System, Signals, Events, Schedule Management, and Logs.

4.1 System

The following tasks relate to the interaction between the WheatNet-IP Scheduler and the WheatNet-IP

system.

4.1.1 Set IP Address

In order to perform any useful function, the program ultimately needs to interface with a WheatNet-IP Blade. Unless you will only be interfacing with a Blade at the default Blade 1 IP address of 192.168.87.101, you will need to know how to specify the Blade that the program communicates with. This procedure has already been discussed in section 3.2.1 above.

Please note that in a system with multiple Blades, it generally doesn't matter which Blade you connect to when running a schedule.

4.1.2 Define Devices

If you plan on using any ACI command events you will need to define the device or devices you want to control. From the menu, select *File>Define Devices* to bring up the following form:

This is quite a busy form, so let's take it on in sections.

The *Current Settings* area indicates any ACI devices that have already been set up. Until you have set up one or more devices this area will indicate that all eight devices are of type *blade*, and all of the IP Address entries will show   . . .   to indicate that they have not been defined.

The *New Settings* area on the right also shows any settings that have already been made, and allows

you to easily add devices or edit existing devices.

The normal procedure for adding a device is to first use the Radio Buttons to specify if the device type is *Blade* (default), *Surface*, or *Processor*. Next, enter the IP Address of the device in the spaces provided. ***It is important to correctly match the device type and the IP Address since different device types use different ports for their ACI communications.***

Editing is just as simple; change the device type, if needed, and enter the new IP Address.

If you want to undefine a device, simply click the *Delete # X* button that corresponds to the device you no longer need.

You can commit your changes at any time by clicking *Apply*; the form will remain open. If you have made some changes but want to cancel your changes, you can click *Cancel* to revert any uncommitted changes back to what they were. *Cancel* will **not** affect changes that have been committed.

When you are done defining devices click the *Close* button to commit your changes and close the form. You will be prompted to restart the program before the new device data will actually be used. Click *Yes* or *No* as desired.
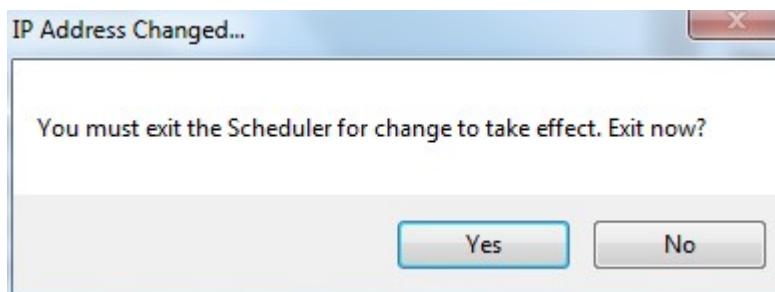
4.2 Signals

Signals are an integral part of Connect and Disconnect events. Although you can schedule one of these events without specifying signals, the event will do nothing when fired.

Signals are operated upon within the WheatNet-IP system according to their Signal ID, and you can, in fact, specify signals by referring to the ID. However, the signal IDs are somewhat cryptic, and signals can be given names that we humans will find much more meaningful.

Further, it makes sense from a human standpoint to have items with similar functions named with similar names. For an example relative to the project at hand, we may find it convenient to have a signal named "Fred" feeding an input on a Blade located in Studio 1, and a signal of the same name feeding a Blade in Studio 2, because Fred sometimes works from one studio and sometimes from the other. And so we allow the name to be extended by appending a Location to the name. In this example we would have Fred – Studio 1 and Fred – Studio 2. In this way we can easily distinguish the two signals even when they share a common name.

Signal names and locations are generally configured with the WheatNet-IP Navigator GUI. We just use the same names here, getting them from the fetched signal set.

Although earlier versions of the Scheduler always attempted to fetch signals from the attached system at program start, that has changed with version 2.0. Now, when the program starts, it first looks for a signal file residing on the computer hard drive. If at some time since installing the program you have

saved a signal set, the program will attempt to load the last saved signal file. Prior to the first time you save a signal set the program will attempt to load a default sample set that was installed for you along with the program.
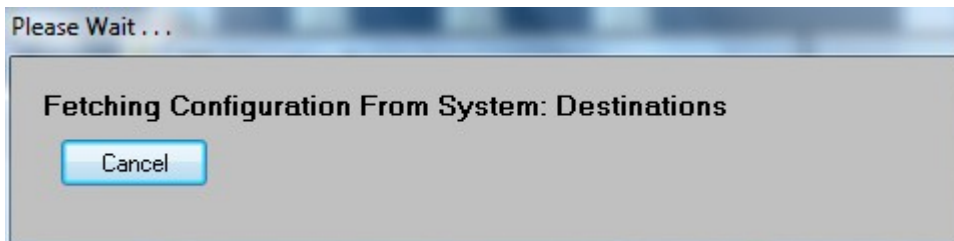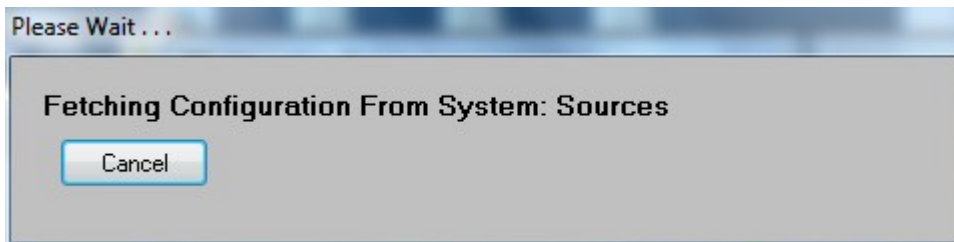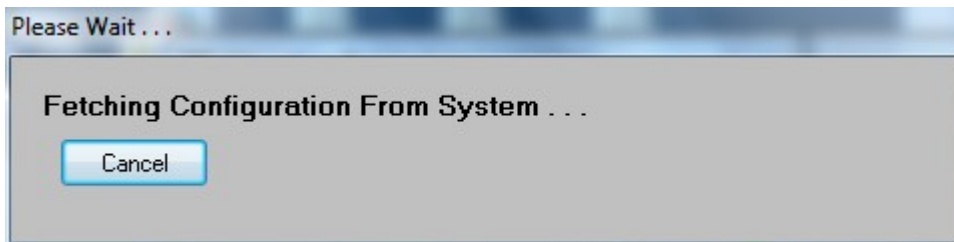
If Scheduler cannot find a signal file at startup it will display a warning, as described above in section 3.1.1. This will not be the case, however, if the AutoSchedule feature has been enabled (see section 4.4.4 below).

It is also possible at any time while the Scheduler is running to request signals from the connected system. This is covered in the next section, 4.2.1.

4.2.1 Request Signals From Connected Blade

To request signals from the connected Blade while at the main screen, select *File>Request Config* from the menu. Or, if you are on the *Event Properties* screen (section 4.3.3) you can click the *Request Config* button. Either method will start the signal fetch process described below:

The fetch process shows the following sequence of screens when connected to a Blade:

Please Wait . . .

Fetching Configuration From System: Salvos

Cancel

Please Wait . . .

Fetching Configuration From System: Done!
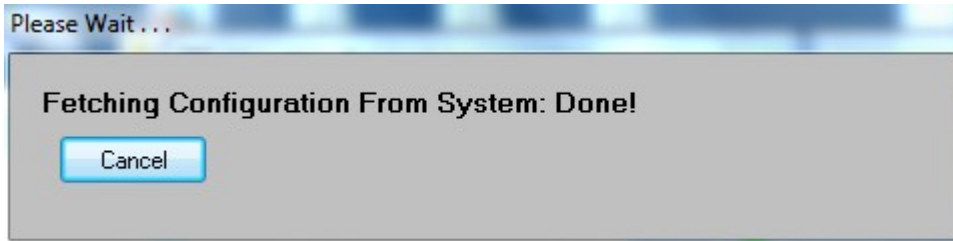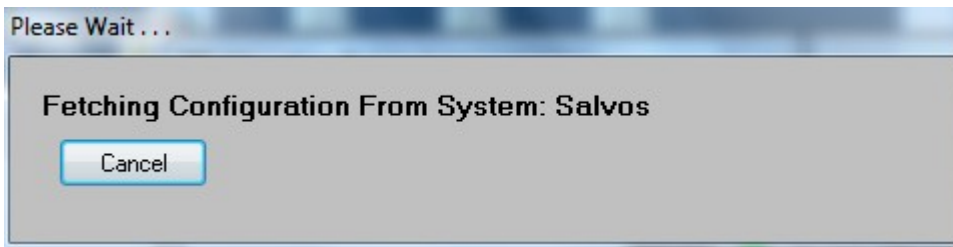
Cancel

You may stop the fetch process at any time by clicking *Cancel*.

When the process completes, a window appears that advises you of some of the actions you can perform with the Scheduler. Close it by clicking *OK*. If you expect to be fetching signals on a regular basis (something you might want to do i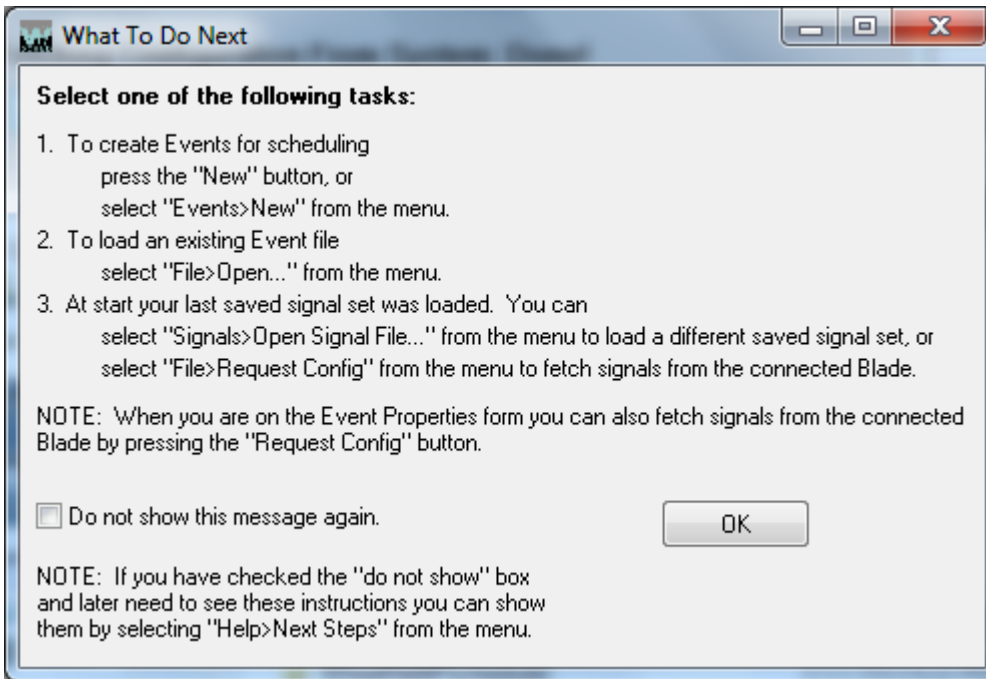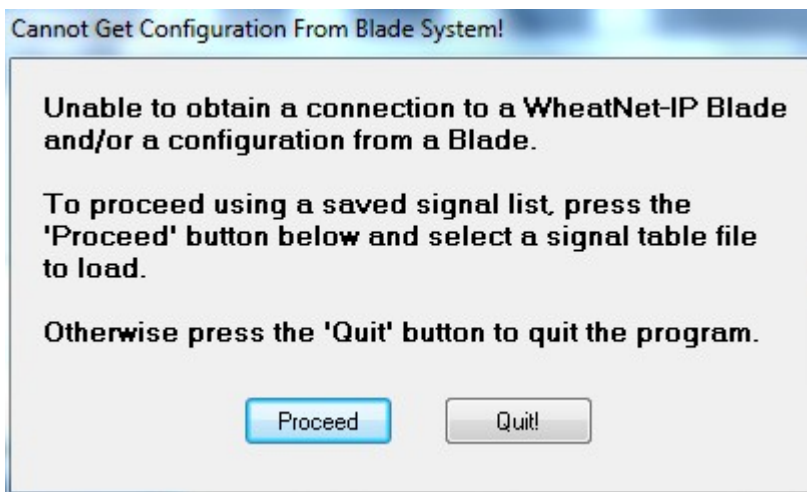f you are slowly building up your system, or if you are making changes to signal names) but don't want to see this screen every time, check the *Do not show this message again* box before clicking *OK*.

What To Do Next

Select one of the following tasks:

1. To create Events for scheduling
   press the "New" button, or
   select "Events>New" from the menu.
2. To load an existing Event file
   select "File>Open..." from the menu.
3. At start your last saved signal set was loaded. You can
   select "Signals>Open Signal File..." from the menu to load a different saved signal set, or
   select "File>Request Config" from the menu to fetch signals from the connected Blade.

NOTE: When you are on the Event Properties form you can also fetch signals from the connected Blade by pressing the "Request Config" button.

☐ Do not show this message again.   OK

NOTE: If you have checked the "do not show" box and later need to see these instructions you can show them by selecting "Help>Next Steps" from the menu.

Should you be unable to fetch signals because the program can't connect to the system, you'll be presented with the above screen. After dismissing that one you'll be presented with the following screen:

Cannot Get Configuration From Blade System!

Unable to obtain a connection to a WheatNet-IP Blade and/or a configuration from a Blade.

To proceed using a saved signal list, press the 'Proceed' button below and select a signal table file to load.

Otherwise press the 'Quit' button to quit the program.

Proceed   Quit!

Clicking *Quit!* will exit from the Scheduler. Clicking *Proceed* will open a standard Windows file open dialog via which you can load a previously saved signal set file.

4.2.2 Load Signals From File

If you have previously save a signal set, or if you want to view the sample signal set that is provided for you during product installation, select *Signals>Open Signal File...* from the main menu. This will open a Windows file open dialog, which will point to the default location for event and signal files the first time the program is run, or the location loaded from or saved to the last time the program was run. Navigate to the desired location, select the file, and click *OK*.

4.2.3 Save Signals To File

If you have successfully fetched signals from a system and want to have a copy of the signal set available so you can create and edit events without needing to be connected to a Blade, you will want to save your set of found signals. Select *Signals>Save Signal File...* from the menu to open a standard Windows file save dialog. Navigate to the desired folder if necessary, choose a memorable name for the file, and click *OK*.

4.3 Events

The unit building block used by the schedule as it operates is termed an Event. As previously discussed, there are four types of events: the Connect Event, the Disconnect Event, the Salvo Event, and the ACI Event. Events are created and edited from the *Event Properties* form, as discussed below in section 4.3.3. Sets of events can also be saved in a file for future scheduling.

4.3.1 Load Events From File

If you have previously saved an event set, you can load that set by selecting *File>Open...* from the main screen. A standard Windows file open dialog allows you to select a file to open. Once the file has been loaded you can begin working with it.

4.3.2 Save Events To File

You may find it desirable to save a set of events you have created. Select *File>Save...* from the main menu. This opens a standard Windows file save dialog.

4.3.3 Create Events

If you are starting from scratch to create events, select *Events>New* from the main menu. This opens the *Event Properties* screen, which looks like this:

Let's take a closer look at the various parts of this form, starting with the right side of the form. At the top we see an *ID* box that we can use to scroll through the events. This is followed by a *Name* field, where the event is given a name. To the right of that is an *Event Type* drop down box, where the type of event will be selected. Note that, in the picture, the *Event Type* is blank, and the tabbed area below displays a tab titled *Undefined*. The tabbed area title and contents will change depending on the type of event selected. When the *Event Properties* form is first viewed by clicking the *New* button on the main screen, or selecting *Events>New* from the menu, this is how the form will look. We will take a closer look at the various event type tabs later.

Worthy of note is the text displayed on the *Undefined* tab. This defines what is required before an event being created is accepted by the software. If you try to proceed from the currently viewed event to another by clicking *Apply* or *OK*, you will not be able to proceed unless certain information is furnished. The event must have an *Event Type* specified, must have at least one *Day* box checked, and must have a *Name* assigned.

Note that you can get away with not assigning signals to a Connect or Disconnect Event; however, such events will do nothing when fired.

The left side of the dialog, at the top, gives you the tools for specifying when an event will fire. The *Start Time* specifies the first time on a checked *Day* that the event will fire. Enter the time in 12 hour mode, and specify *AM* or *PM*.

The *Repeat Interval* determines how often an event will fire on any checked *Day*. The *Repeat Interval* is specified in hours and minutes. As an example, if the *Repeat Interval* is specified to be 4 hours and 30 minutes, and the *Start Time* is 2:15:00 PM, the event can be expected to fire on a checked day at 2:15, 6:45, and 11:15 (all these are PM times).

There are separate check boxes for each day of the week, *Sunday* through *Saturday*. Events will not be fired on days that are not checked. For convenience, the *Check All Days* and *Uncheck All Days* buttons are provided, with rather obvious consequences. The *Invert Selection* button checks all unchecked *Day* boxes and unchecks all checked *Day* boxes.

The *1-Time-Only* check box is a little less obvious. When this box is checked, once an event fires it is deleted from the Scheduled Events list. However, if the box is not checked, the fired event is added at the bottom of the schedule, one week later in time than when it just fired.

Below the schedule settings are two buttons used to enter the event in the *Event Manager* list. *OK* enters the current event in the list and closes the *Event Properties* dialog, while *Apply* adds the current event to the list but leaves the *Event Properties* dialog in place. A third button, *Cancel*, er – ah – cancels the event, and it is not added to the list.

Advancing the previously described *ID* spin box without first clicking *Apply* on the current event results in a prompt that asks you to decide if you want to apply changes before viewing the next event.

The previously discussed *Request Config* button on the left side of the dialog at the bottom is used to fetch signals from a connected Blade.

4.3.4 View Events

The *Event Properties* dialog shows you the properties of the event currently being viewed. As mentioned above, the tabbed area will change depending on the *Event Type*. Let's look at this a little more closely.

If *Connect* is selected as the *Event Type*, the tabbed area looks like this:

Notice that a Connect Event is used to connect a given source signal to a given destination signal. The signals are specified on this tab. You can select a signal by its *Signal ID* if you know it, but you do not need to know the *Signal ID* to specify it. The signal can also be selected by *Name and Location*. If you are not sure of the *Name and Location* associated with the signal you want to use, and do not know the *Signal ID* either, you can still have the event added to the *Event Manager* list. However, the event will do nothing when fired unless valid signals are specified.

Note that when you scroll to an already defined Connect Event the programmed signals will be shown.

If *Disconnect* is selected as the *Event Type* the tabbed area looks like this instead:
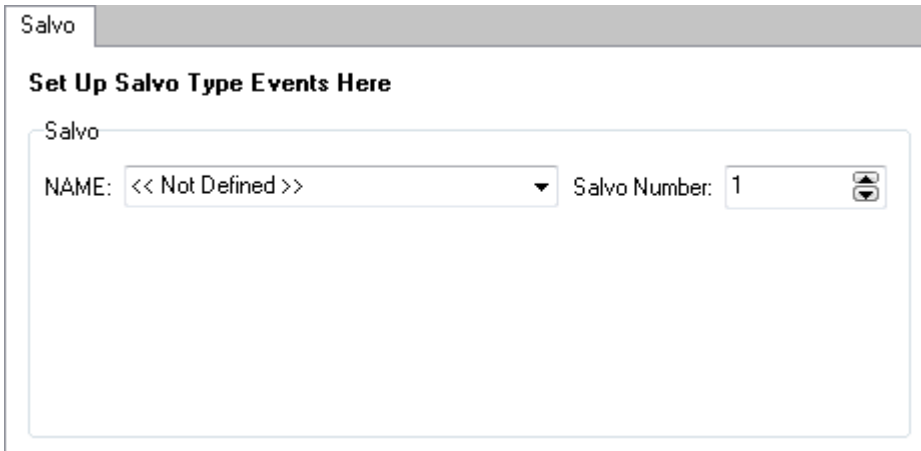

A Disconnect Event is used to disconnect the current source signal (which does not need to be specified) from a given destination signal. The destination is specified on this tab. Once again you can

select a signal either by its *Signal ID* or by its *Name and Location*.

Note that when you scroll to an already defined Disconnect Event the programmed signal will be shown.

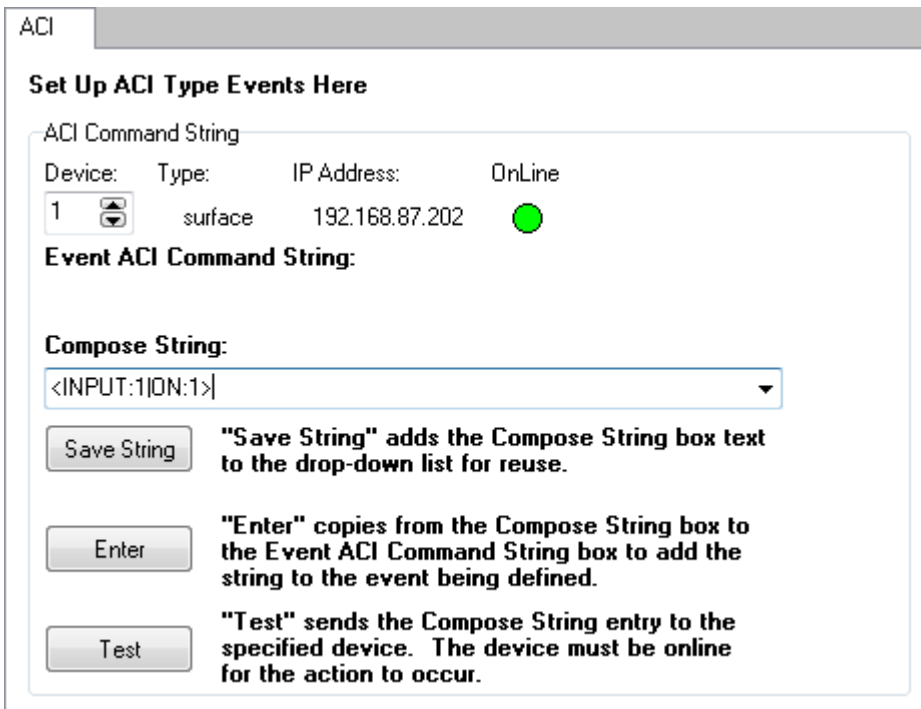If *Salvo* is selected as the *Event Type* the tabbed area changes once again:

A Salvo Event is used to select a particular salvo to fire (please read the section of the WheatNet-IP manual that deals with the Navigator software to find out more about salvos). You can select the desired salvo either by its *Signal Number* or by its *Name*.

Note that when you scroll to an already defined Salvo Event the programmed salvo information will be shown.

Selecting ACI as the Event Type results in this in the tabbed area:

Use the *Device* spin dial to select the desired device for this **ACI** event. As you cycle through the defined devices the device *Type* and *IP Address* fields will reflect the properties of the device. In this

instance we have selected device 1, which is a control surface located at 192.168.87.202. The *OnLine* indicator will be green if the program is able to connect to the selected device, or red if not.

Once the device is selected you must compose a command string appropriate to the device. In our example we want fader 1 on the surface to turn on when the command is issued. The basic format for a fader on/off command to a surface is this: `<INPUT:x|ON:y>`   where `x` indicates the fader number and `y` indicates the desired fader state, with `1` indicating on and `0` indicating off. This command is entered in the *Compose String* field. Note that letters are automatically converted to upper case when entered, so there is no need to use Shift or Caps Lock.

Once you have composed the desired command string you may want to test it. Press the *Test* button and the string in the *Compose String* field will be sent to the selected device.

When you are sure the command string is doing what you want, press *Enter*. This copies the command string from the *Compose String* field to the *Event ACI Command String* field.

If you have composed a command string you think you may need again press the *Save String* button. The program can remember up to 50 command strings this way.

Note that when you scroll to an already defined ACI Event the programmed device information will be shown, and command string will appear in the *Event ACI Command String* field.

As previously mentioned, it is beyond the scope of this manual to list all of the ACI commands available for your use. Please contact Wheatstone Technical Support at 252-638-7000, or email us at techsupport@wheatstone.com, to discuss your specific requirements.

4.3.5 Edit Events

If you have an existing event that you want to modify, you once again need to be on the *Event Properties* dialog. If you are already on this form you can use the *ID* spin dial to scroll to the event you want to change. Or if you are at the main window and viewing the *Event Manager* tab you can highlight the event and click *Edit* or select *Events>Edit* from the menu. The process of editing an event is to change the *Event Properties* from the current settings to the desired settings. The pertinent information has already been discussed.

4.3.6 Schedule Events

With a few events in the Event Manager list, your main screen, *Event Manager* tab, could look like this:
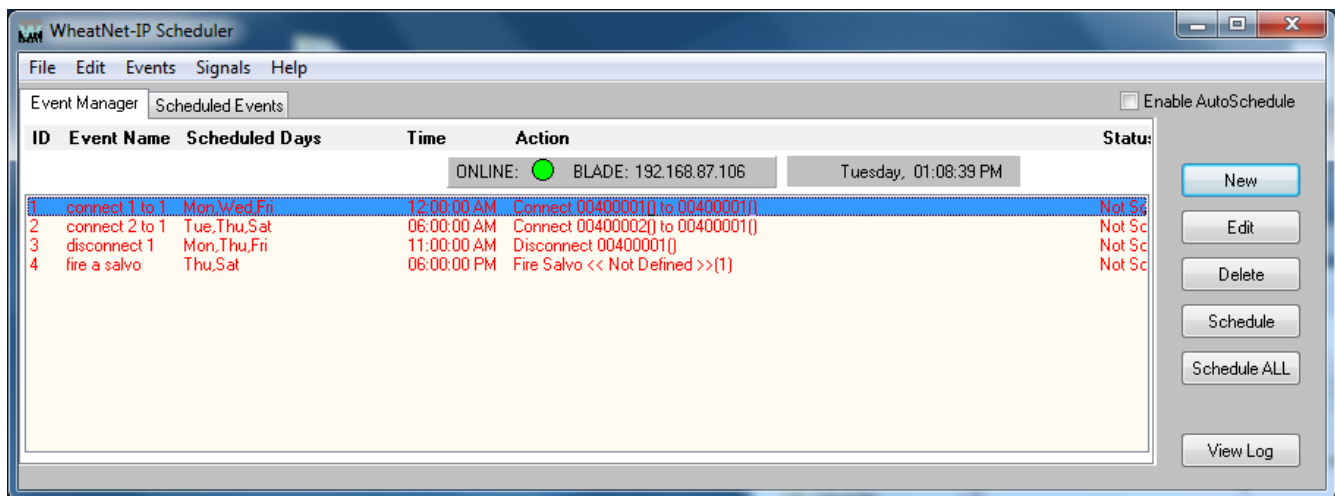
With a single event highlighted, as shown, you can click *Schedule*, or select *Events>Schedule*, to schedule only that event. You could also highlight, using either the Shift key or the Ctrl key, a small number of events, say two or three, and use *Schedule* to schedule only those events. Or you can use the *Schedule All* button or select *Events>Schedule All* to add all events in the list to the schedule – in this case you only need to highlight one event.

4.3.7 Delete Events

From the *Event Manager* list, you can highlight an event and click the *Delete* button or select *Events>Delete* from the menu to delete that event. This will delete the event from the *Event Manager* list and also delete any as yet unfired instances of the event from the schedule. If the event (or events) has not yet been scheduled, the delete process will happen automatically. But when removing scheduled events you will be asked to verify your intent:
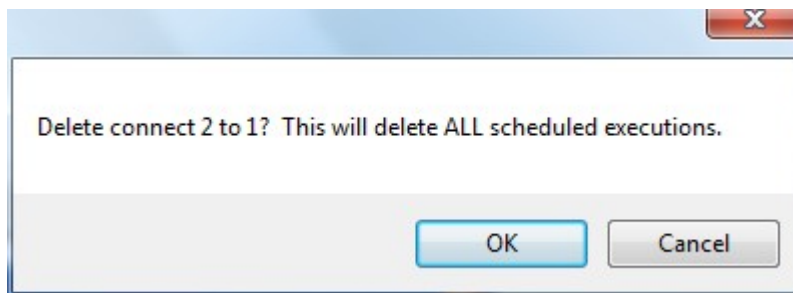
Click *OK* to proceed, or *Cancel* if you've changed your mind.

4.4 Schedule Management

Once you have created some events and added them to the schedule, you can view the schedule list from the main form by selecting the *Scheduled Events* tab:
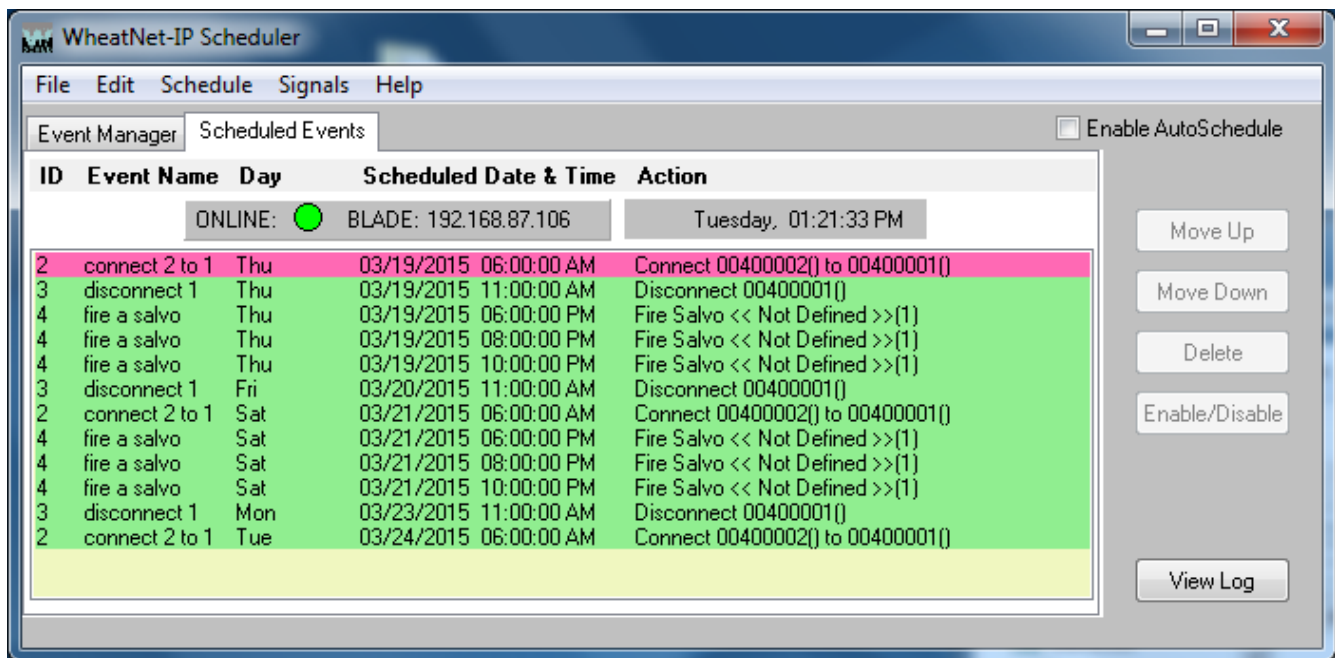
As the event at the top of the list fires, it is removed from the list. If the event is not specified to fire one time only, the same event will then appear at the bottom of the schedule list, for a time one week in the future from the time the event just fired.

The event at the top of the list is the next one due to fire. It is highlighted with a light red background, whereas the other events in the list are normally highlighted with a light green background.

If you click on an event in the list to select it for an action, such as to move, delete, or disable it, the event text changes from black to white. Multiple events can be selected using the Shift or Ctrl keys, as in normal Windows operation.

If an event is disabled it will show in the list as red text on a white background. An example of this is shown later.

4.4.1 Moving Items

Events in the schedule must fire in sequence. This means that two events scheduled to fire at the same time will still fire one after another, in the sequence they are listed on the *Scheduled Events* tab. If you have two (or more) such events and need to fire them in a specific sequence, you can adjust their relative positions in the list. If you highlight one of two (or more) events scheduled for the same time, one (or possibly both) of the *Move* buttons (*Move Up* and *Move Down*) will become active. Clicking *Move Up* moves the position of the event one place higher up the screen, whereas *Move Down* does the opposite. The buttons will be disabled (grayed out) if the event highlighted is scheduled for a different time than the one in the direction that the button would move it if enabled.

4.4.2 Deleting Items

Just as you can delete events from the *Event Manager* tab, you can also delete events from the *Scheduled Events* tab. But the choices are a little more complicated. Highlight an event on the *Scheduled Events* tab, click *Delete* (or select *Schedule>Del From Schedule* from the menu) and the following dialog pops up:
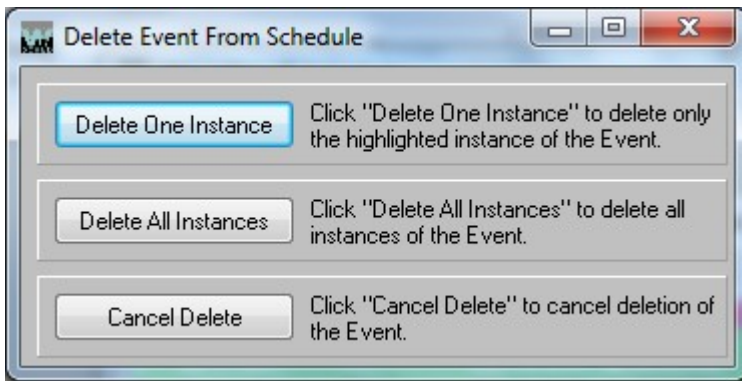
The choices are clearly explained on the dialog box. Click the appropriate button.

4.4.3 Enabling And Disabling Items

You can disable all instances of an event from the *Scheduled Events* tab. This will prevent the event from firing but will not delete it from the schedule. The event can later be enabled. Select an event and click the *Enable/Disable* button or select *Schedule>Enable / Disable* from the menu. A dialog box asks you if you want to disable all instances of the event:
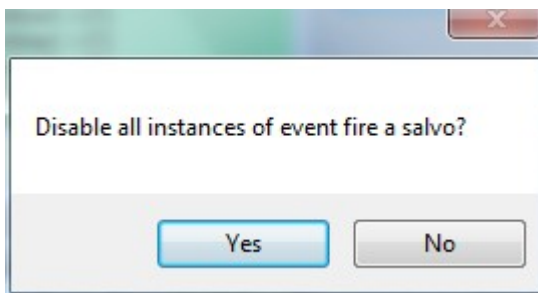
Click *Yes* to disable all instances of the event, or *No* to cancel the delete.

You can select multiple events from the list using the normal Windows techniques with the Shift and Ctrl keys. You will be prompted individually for each event you have selected.

Here is a section of the *Scheduled Events* tab showing a few disabled events:

To re-enable an event (or events) the process is similar. The prompt dialog asks if you want to enable all instances of the event.

4.4.4 The AutoSchedule Feature

The Scheduler can be set up to load and run your schedule automatically at program start. The check box for enabling and disabling the **AutoSchedule** feature is located near the upper right corner of the Scheduler main window, and appears no matter which tab, *Event Manager* or *Scheduled Events*, is being displayed.

When **AutoSchedule** has not been enabled the check box looks like this:
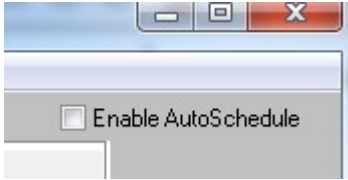
When **AutoSchedule** is enabled the check box looks like this:
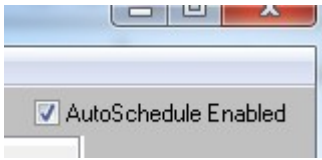
When **AutoSchedule** is enabled and the Scheduler started, it will attempt to load the last saved signal set. Whether this operation succeeds or not it will also attempt to load the last saved event file. If it succeeds at finding the event file it will schedule all events in the file and begin firing scheduled events at the required times.

The Scheduler, when *not* in **AutoSchedule** mode, will pop up a message indicating failure to find a signal set if it can't find the default set, which will then require an operator to click *OK* on the message box to continue. But in **AutoSchedule** mode, if the signals cannot be found, no message about that appears. This is to keep the lack of a found signal set from preventing the running of the schedule automatically.

When the Scheduler is in **AutoSchedule** mode, it will stop and present a message requiring user action if it cannot find a schedule to run.

The main purpose of providing the **AutoSchedule** feature is so that a computer running a schedule on the Scheduler program can recover from a power outage without requiring user intervention to get the schedule running. In order for this to succeed you will need to be running the Scheduler as a user on the computer who is automatically enabled at computer startup without the need to have the user log in. Otherwise the program will not be able to start without some user input.

4.4.5 Exiting With A Schedule Running

Obviously, the software can only fire events from the schedule when the software is running. If a schedule is running and you close the program the following prompt will appear:
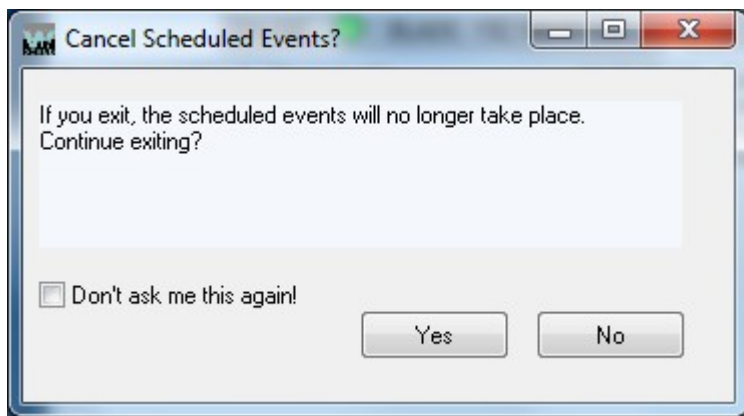
Clicking *Yes* will close the program, with the result that any as yet unfired events in the schedule will not be fired. Clicking *No* will dismiss the prompt and the program (and schedule) will continue to run. If you are tired of seeing this screen at program exit, check the *Don't ask me this again!* check box. The software will then no longer warn you at program close.

4.4.6 More On Exiting

Upon exiting the program you will see the following prompt:
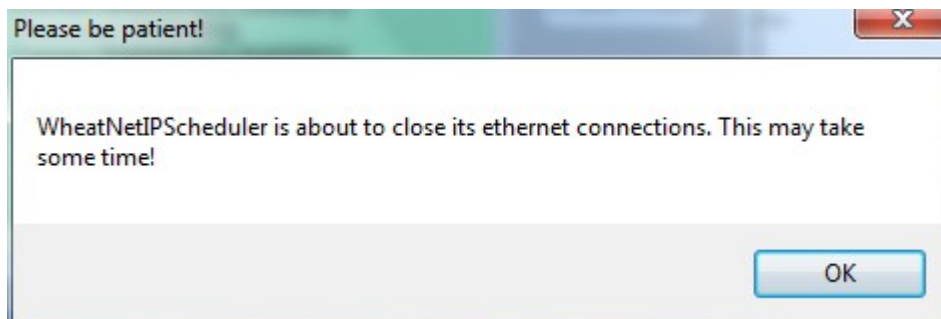
The program may continue running for a short time after all its windows disappear as it shuts down device connections.

4.5 Logs

The WheatNet-IP Scheduler program automatically keeps logs to indicate various types of activity. The files are named `EventLog_weekX.log`, where the X can be a number from 1 to 4. Upon program exit, log data from the current session is appended to the end of the currently active log file. Older logs are overwritten, so you need to look at the Date/Time stamp of the log files to determine which one is most current. Log files live in the same default location as the signal and event files previously discussed, typically `C:\Documents and Settings\<username>\Application Data\Wheatstone\` in Windows XP or `C:\Users\<username>\AppData\ Roaming\Wheatstone\` in Windows 7.

Should you need to contact Wheatstone Technical Support regarding an issue with the software you may be asked to provide the logs.

4.5.1 Viewing Logs

The current log may be viewed by selecting *View Log* from either the *Events* or the *Schedule* menu item, or by clicking the *View Log* button on the main window. A typical log when viewed shortly after the program starts might look like this:
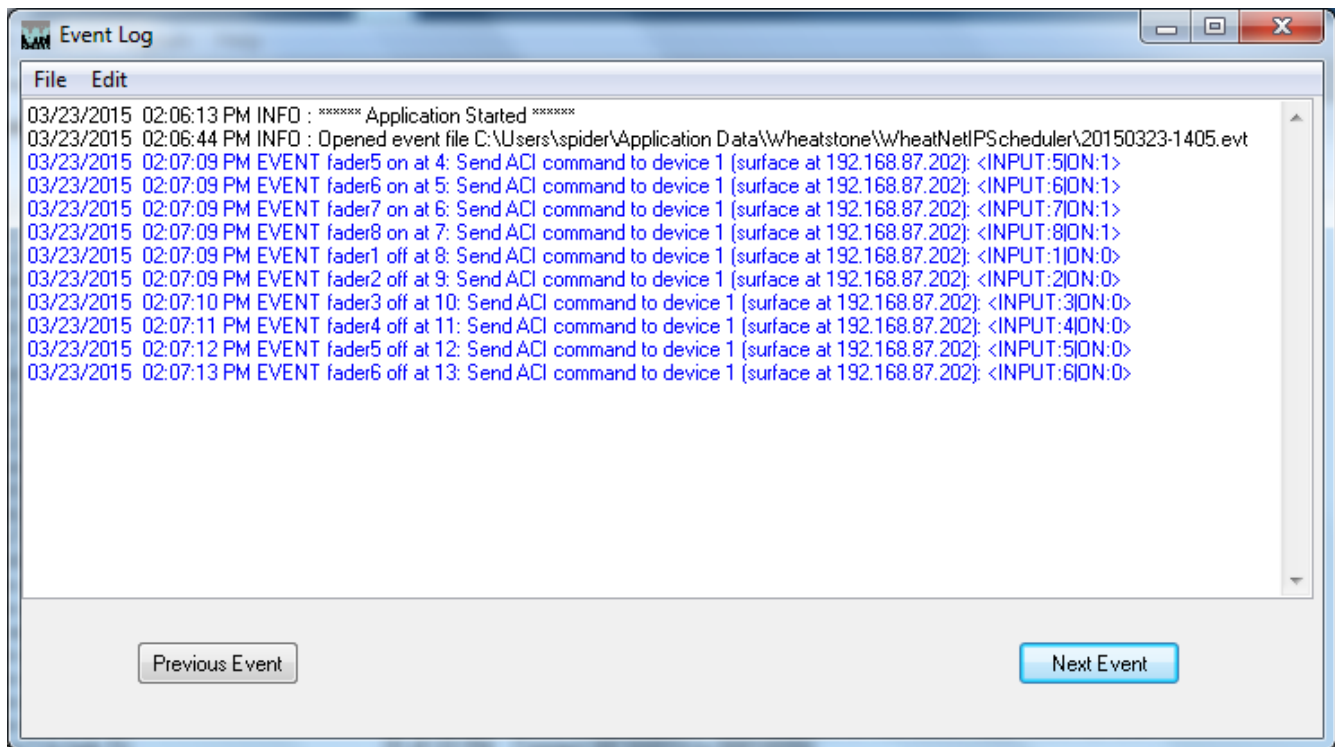
Also added to the log file are indications of events that have fired, requests for fetching signals, opening and saving event and signal files, and other useful information.

If any events have fired, they will be listed in the log. Any such listing contains the text "EVENT". This keyword is used when clicking the *Previous Event* and *Next Event* buttons to find all such listings in the log.

Since the log files are plain text files, they can also be viewed in any simple text editor.

5. Menus

The WheatNet-IP Scheduler software is operated using buttons and menus. This section serves to bring together the descriptions of the various menu items that have been described throughout the manual. Some items in the main menu depend on which tab is being viewed and what items on that tab are highlighted. The small menu on the log form is described at the very end of this section.

5.1 File

The *File* menu contains a few items that are necessary for basic program configuration and operation. The available items are:

Open...   This item allows you to open a previously saved set of events for editing and/or scheduling.

Save...   This item allows you to save the current set of events for later use.

Set IP Address...   This item allows you to specify the Blade that you expect to connect to for obtaining signals and as the avenue by which your fired Connect, Disconnect, and Salvo events will control the system. You will need to restart the program after changing this setting.

Define Devices   This item allows you to specify the device or devices that you expect to connect to for firing ACI events. You will need to restart the program after changing any device settings.

Request Config   This item allows you to fetch the signals known to the Blade to which you are connected. The program's existing signal set is deleted first, so you may want to save a signal set before fetching a new one.

Print...   When selected while viewing the *Event Manager* tab, this item allows you to print the contents of the event list. When selected while viewing the *Scheduled Events* tab, this item allows you to print a snapshot of the current schedule list. Events that have already fired will not appear on the printout.

Exit   This item allows you to close the Scheduler software.

5.2 Edit

The *Edit* menu contains a couple of items that can be useful in creating new events that are similar to existing events. This menu is only usable on the *Event Manager* tab. The available items are:

Copy   This item will copy the *Start Time* and *Repeat Interval* of the currently highlighted event to the clipboard, with a *Name* like "Copy of This Event", assuming the *Name* of the event copied is "This Event".

Paste   This item will paste the contents of the clipboard into the next available line on the event list.

5.3 Events

The *Events* menu is only visible when viewing the *Event Manager* tab. It contains items useful in managing the creation and editing of events. The available items are:

New   This item calls up the *Event Properties* dialog with the first available undefined event showing.

Edit   This item calls up the *Event Properties* dialog with the currently selected event showing. If multiple events are highlighted, the last one selected is the one that will be showing.

Delete   This item begins the delete process for the event(s) highlighted. When an event is deleted from the *Event Manager* tab it is removed from the event list, and all scheduled instances are removed from the schedule list.

Schedule   This item adds the highlighted event(s) to the schedule, at all scheduled times indicated by the *Event Properties*.

Schedule All   Regardless of event(s) highlighted, this item adds **all** events to the schedule at all scheduled times indicated by the *Event Properties*.

View Log   This item calls up the *Event Log* window.

5.4 Schedule

The *Schedule* menu is only visible when viewing the *Scheduled Events* tab. It contains items useful in managing the scheduling of events. The available items are:

Move Up   If the currently highlighted event is scheduled at the same time as the event immediately above it in the schedule list, this item moves the highlighted event up so it will fire first.

Move Down   Similarly, if the currently highlighted event is scheduled at the same time as the event immediately below it in the schedule list, this item moves the highlighted event down so it will fire second.

Del From Schedule   This item begins the delete process for the event(s) highlighted. When an event is deleted from the *Scheduled Events* tab it is **not** removed from the event list. You can also specify whether all scheduled instances are removed from the schedule list, or only the highlighted instance.

Enable / Disable   This item toggles the "firing state" of the highlighted event(s) in the schedule, at all scheduled times indicated by the *Event Properties*.

View Log   This item calls up the *Event Log* window.

5.5 Signals

The *Signals* menu contains a couple of items that can be useful in managing signal lists. The available items are:

Open Signal File...   This item opens a previously saved signal set file. This is useful if you want to do offline editing of events without being connected to a Blade since it allows you to use the same signal set the software will be working with when connected to the target system.

Save Signal File...   This item allows you to save the signal set in current use to a file. The file can be loaded to do offline editing. You can also print the file (after opening it in a standard text editor) for printed documentation of your system's signals.

5.6 Help

The *Help* menu contains a couple of useful items:

Next Steps   This item displays a message box with some quick hints on using the software.

About   This item reveals the Version number of the software.

5.7 Event Log Menu

The *Event Log* window has a small menu to help perform some useful tasks.

5.7.1 Event Log Menu – File

The *File* menu on the *Event Log* window contains a few items that are helpful in working with logs. The available items are:

Save as...   This item allows you to save the current contents of the *Event Log* window to a file without removing the text from the window.

Purge to File...   This item allows you to clear the *Event Log* window. The current contents of the window are appended to the end of the currently active `EventLog_weekX.log` file. A brief line is added to the now clear *Event Log* window indicating the date and time the window was purged and what file the data was appended to.

Print...   This item allows you to print the contents of the *Event Log* window.

Exit   This item closes the *Event Log* window. It does not close the Scheduler software.

5.7.2 Event Log Menu – Edit

The *Edit* menu on the *Event Log* window contains one additional item useful in working with logs:

Copy   This item copies any highlighted text in the *Event Log* window to the Windows clipboard. The text can then be pasted into another application, such as a text editor, from the clipboard. This is useful when you want to copy some data from the *Event Log* window, but don't want the entire contents.