Technical Document

Niagara$^{AX-3.4}$ McQuay Driver Guide

June 10, 2010

Niagara^{AX} McQuay Driver Guide

Confidentiality Notice

While every effort has been made to assure the accuracy of this document, Tridium is not responsible for damages of any kind, including without limitation consequential damages, arising from the application of the information contained herein. Information and specifications published here are current as of the date of this publication and are subject to change without notice. The latest product specifications can be found by contacting our corporate headquarters, Richmond, Virginia.

Trademark Notice

Copyright and Patent Notice

CONTENTS

Preface

This documents usage of the McQuay® driver for the NiagaraAX framework.

The following main sections are in this document:

- "Compatibility and Installation" on page 1-1
  Explains McQuay devices/protocols supported, as well as the NiagaraAX platform, software, and licensing requirements.
- "McQuay Driver Quick Start" on page 2-1
  Provides several quick procedures for online station configuration to add a McQuayNetwork, McQuayDevices, and McQuay proxy points.
- "NiagaraAX McQuay Concepts" on page 3-5
  Provides concepts behind the McQuay driver, including all major components and views, including various screen captures.
- "McQuay Plugin Guides" on page 4-13
  Provides brief summaries of the different McQuay manager views, each with links back to the more detailed concepts section. Entries are used in NiagaraAX context-sensitive help "On View".
- "McQuay Component Guides" on page 5-15
  Provides brief summaries of the different McQuay components, most with links back to the more detailed concepts section. Entries are used in NiagaraAX context-sensitive help "Guide On Target".

Document Change Log

Updates (changes/additions) to this *NiagaraAX McQuay Driver Guide* document are listed below.

- Published: June 10, 2010
  Initial document.

CHAPTER 1

Compatibility and Installation

Currently, this section has the following main subsections:

- Compatibility
- License requirements
- Software installation
- McQuay OPM to JACE connection

Compatibility

The NiagaraAX McQuay driver has the following compatibility criteria:

- NiagaraAX platform compatibility
- McQuay compatibility

NiagaraAX platform compatibility

The McQuay driver will function on all NiagaraAX platforms that support serial communications. NiagaraAX-3.4 or later is required.

McQuay compatibility

The McQuay driver is compliant with a McQuay MicroTech® Open Protocol Master (OPM) device, including any MicroTech controllers that are RS-485 networked to that OPM device.

License requirements

To use the NiagaraAX McQuay driver, you must have a target NiagaraAX host (JACE) that is licensed with the "`mcquay`" feature, as well as the "`serial`" feature. In addition, the "`mcquay`" feature may have other device limits or proxy point limits.

Software installation

From your PC, use the Niagara Workbench 3.*4.nn* or higher installed with the "installation tool" option (checkbox "This instance of Workbench will be used as an installation tool"). This option installs the needed distribution files (*.dist* files) for commissioning various models of remote JACE platforms. The dist files are located under your Niagara install directory under a "sw" subdirectory. For more details, see "About your software database" in the *Platform Guide*.

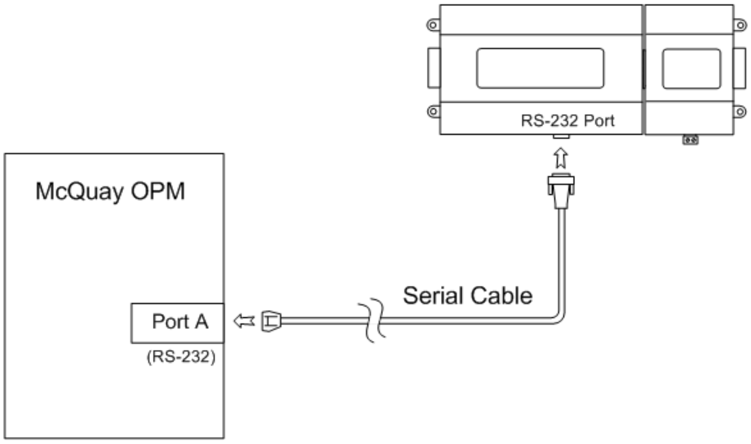
Apart from installing the 3.*4.nn* or higher version of the Niagara distribution in the JACE, make sure to also install the *mcquay* module too, plus any modules shown as dependencies. For more details, see "About the Commissioning Wizard" in the *JACE NiagaraAX Install and Startup Guide*.

See the next section "McQuay OPM to JACE connection" for cabling information.

McQuay OPM to JACE connection

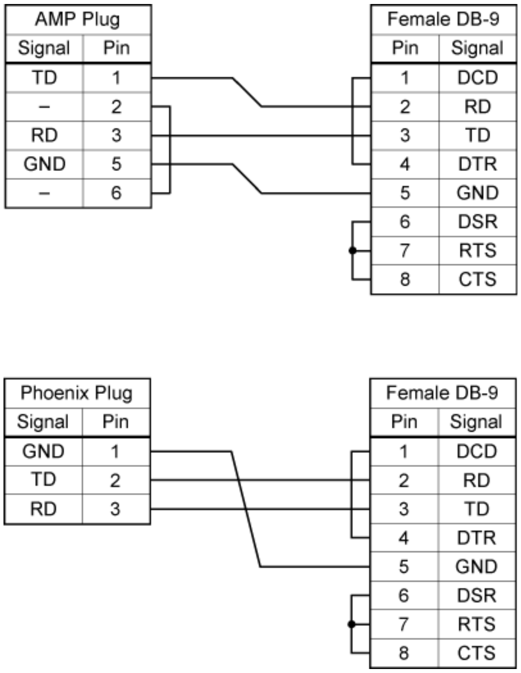
Driver communications is via RS-232 between a JACE serial port and Port A of the McQuay OPM device, using a cable as shown in Figure 1-1.

Figure 1-1 *McQuay OPM to JACE serial cable connection*



The serial cable must be a twisted, shielded pair type with drain wire, maximum length 50 feet. The OPM end must be properly terminated to an AMP or Phoenix plug, with the JACE end typically terminated to a DB-9F (socket) connector for its RS-232 port. Pinouts for the cable are shown in Figure 1-2.

Figure 1-2 *Pinouts for serial cable using 6-pin AMP connector (top) or 3-pin Phoenix connector (bottom)*



AMP Plug	
Signal	Pin
TD	1
–	2
RD	3
GND	5
–	6

Female DB-9	
Pin	Signal
1	DCD
2	RD
3	TD
4	DTR
5	GND
6	DSR
7	RTS
8	CTS

Phoenix Plug	
Signal	Pin
GND	1
TD	2
RD	3

Female DB-9	
Pin	Signal
1	DCD
2	RD
3	TD
4	DTR
5	GND
6	DSR
7	RTS
8	CTS

Note: *Check the OPM to see if it requires the Phoenix plug or the AMP plug (possibly using an AMP-to-Phoenix plug adapter cable).*

AMP part numbers for the AMP connector shown in Figure 1-2 are:1-480270-0 (plug) and 60617-1 (female pin terminals).

After you connect this serial cable, the JACE is now ready for McQuay integration, as described in the rest of this document. See the next section "McQuay Driver Quick Start" for a series of task-based procedures.

CHAPTER 2

McQuay Driver Quick Start

This section provides a collection of procedures to use the NiagaraAX McQuay driver to build a McQuayNetwork with proxy points. Like other NiagaraAX drivers, you can do most configuration from special “manager” views and property sheets using Workbench.

Note: *First see “Compatibility and Installation” on page 1-1 for licensing and software requirements.*

These are the main quick start subsections:

- Add and configure the McQuayNetwork
- Add McQuayDevices
- Create McQuay proxy points

Add and configure the McQuayNetwork

- Add the McQuayNetwork
- Configure McQuayNetwork communication parameters

Add the McQuayNetwork

The McQuayNetwork is the top-level McQuay component in the station.

To add a McQuayNetwork in the station

Step 1 Double-click the station’s **`Drivers`** container, to bring up the **`Driver Manager`**.

Step 2 Click the **`New`** button to bring up the New network dialog. For more details, see “Driver Manager New and Edit” in the *Drivers Guide*.

Step 3 Scroll to select “`McQuay Network`,” number to add: `1` and click **`OK`**.

This brings up a dialog to name the network.

Step 4 Click **`OK`** to add the McQuayNetwork to the station.

You should have a McQuayNetwork named “McQuayNetwork” (or whatever you named it), under your Drivers folder, showing a status of “`{fault}`” and enabled as “`true`.”

After you Configure McQuayNetwork communication parameters, status should change to “`{ok}`”.

Configure McQuayNetwork communication parameters

In the McQuayNetwork property sheet you must define several parameters for communications.

To set the McQuayNetwork communications parameters

To set the communications parameters for a McQuayNetwork:

Step 1 Right-click the McQuayNetwork and select **`Views > Property Sheet`**.

The **`Property Sheet`** appears.

Step 2 Expand the **`Serial Port Config`** slot.

Set the properties for the JACE serial port used, where defaults are typically correct for OPM access, as:

- Port Name: `none` — Enter the JACE RS-232 port being used, like COM2 or COM3.
- Baud Rate: `Baud9600` — Or choose different from selection list.
- Data Bits: `Data Bits7` — Or choose different from selection list.
- Stop Bits: `Stop Bit1` — Or choose different from selection list.
- Parity: `Even` — Or choose different from selection list.
- Flow Control Mode: `none` — Or choose different using checkbox.

Note: *If the OPM's serial port is configured differently from the defaults shown, adjust the baud rate, data bits, stop bits, parity, and flow control settings as necessary to match.*

Step 3 Enter a **`Site Password Hex`** property value that matches the McQuay Open Protocol Site License Password. This is an 8-character field, using hexadecimal notation.

Note: *If less than 8 characters are entered, this value is automatically set to all "F"s, that is:* `FFFFFFFF`

Step 4 Click the **`Save`** button.

For further details on the McQuayNetwork, see "About the McQuay Network" on page 3-6.

Add McQuayDevices

After adding a McQuayNetwork, you can use the network's default "McQuay Device Manager" view to add McQuayDevices.

Note: *You need the address information for each McQuay MicroTech controller under the OPM. Also have the McQuay device documentation available, in order to map data correctly in NiagaraAX proxy points using register addresses.*

To add a McQuayDevice in the network

Use the following procedure to add McQuay devices in the network.

Step 1 In the Nav tree or in the Driver Manager view, double-click the network, to bring up the device manager (McQuay Device Manager).

Note: *For general device manager information, see the "About the Device Manager" section in the Drivers Guide.*

Step 2 Click the **`New`** button to bring up the **`New`** device dialog.

`Type` will be preselected to McQuay Device.

Step 3 Select for number to add: `1` (or more, if multiple) and click **`OK`**.

This brings up a dialog to name the device(s), where for each you must enter its unique two-part network address as a combination of its:

- Master Node L2 Addr — 1 to 255 decimal (OPM itself is L2 address 0).
- Slave Node L3 Addr — 0 to 255 decimal.

MicroTech devices default to hexadecimal L2 = 00, L3 = FF, or in decimal L2=0, L3=255. For more details on this step, see "McQuay Device Manager" on page 3-8.

Step 4 Click **`OK`** to add the McQuayDevice(s) to the network.

You should see the device(s) listed in the device manager view, each showing a status of "`{ok}`" and enabled as "`true`."

If a device shows "`down`" check the configuration of the network and/or the device addresses. You can simply double-click a device in the device manager to review settings in an **`Edit`** dialog, identical to the **`New`** dialog when you added it.

After making any address changes, click **`Save`**, then right-click the device and select **`Actions > Ping`**.

Note: *The default "ping monitor" frequency for McQuayDevices is every 5 seconds—if necessary, you can adjust this from the property sheet of the McQuayNetwork (expand the "Monitor" slot) for this and other properties related to device status. See "About the McQuayDevice" on page 3-9.*

Create McQuay proxy points

As with device objects in other drivers, each McQuayDevice has a **`Points`** extension that serves as the container for proxy points. The default view for any Points extension is the Point Manager (in this case, the "**`McQuay Point Manager`**"). You use it to add McQuay proxy points under any McQuayDevice.

For general information, see the "About the Point Manager" section in the *Drivers Guide*.

Note: *Unlike the point managers in many other drivers, the* **`McQuay Point Manager`** *does not offer a "Learn mode" with a* **`Discover`** *button and pane. The simplicity of the McQuay OPM protocol excludes these functions. Instead, you simply use the* **`New`** *button to create proxy points, and refer to the appropriate McQuay documentation to specify the register addresses of data items in each device.*

To add McQuay proxy points

Once a McQuayDevice is added, you can add proxy points to read and write data. If programming online (and the device shows a status of "{ok}"), you can get statuses and values back immediately, to help determine if point configuration is correct. Use the following procedure:

To create McQuay proxy points in a device:

Step 1 In the **Device Manager**, in the **Exts** column, double-click the Points icon in the row representing the device you wish to create proxy points.

This brings up the **McQuay Point Manager**.

Step 2 (Optional) Click the **New Folder** button to create a new points folder to help organize points, and give it a short name, such as "Analog Inputs", or whatever name works for your application. You can repeat this to make multiple points folders, or simply skip this step to make all proxy points in the root of **Points**.

Note that each points folder has its own **McQuay Point Manager** view, just like **Points**. If making points folders, double-click one to move to its location (and see its point manager).

Step 3 At the location needed (**Points** root, or a points folder), click the **New** button.

The **New** points dialog appears, in which you select a point "Type" and "Number to Add".

For more details on adding proxy points, see "Adding McQuay proxy points" on page 3-10.

Step 4 Click **OK**.

This brings up another **New** dialog to name the point(s), enter register hex address, and specify data length in number of registers. Additional properties are also available for editing, if needed (sometimes Conversion type should be changed, for example).

Default point names are similar to: "*<PointType>*", for example: "McQuayAI" or "McQuayEnum".

- Refer to the appropriate McQuay document for valid (hex) register addresses for specific data items, along with data length—most items are one (1) but sometimes two (2) or more registers in length.
- Point type "McQuay Enum" provides read-only access to one of 6 different enumerated values in the device, chosen only by "EnumType". Do not enter values in "Register Hex" or "Data Len Registers".

Again, find more details on this step in "Adding McQuay proxy points" on page 3-10.

Step 5 Click **OK** to add the proxy point(s) to the Points extension (or to the current points folder), where each shows as a row in the point manager.

Each point should have a status of "{ok}" with a polled value displayed.

- If a point shows a "{fault}" status, check its ProxyExt "Fault Cause" property value.
  In such a case, re-check the address in the point against the documented address for the data item.
- Except for McQuay Enum points, values displayed reflect raw values, typically 0 - 255. Sometimes for AI and AO points you must apply scale and offset values.
  - Do this by editing a point's **Conversion** property, selecting **Linear**.
  - In the Scale and Offset fields, enter the appropriate values, per McQuay documentation.
- As needed, set the Facets property for each AI, AO, BI, and BO proxy point (Enum proxy points automatically use an appropriate frozen facet).

Step 6 Continue to add proxy points as needed under the **Points** extension of each McQuayDevice.

As needed, double-click one or more existing points for the **Edit** dialog, similar to the **New** dialog used to create the points. This is commonly done for re-editing items like register addresses, names, and so on.

CHAPTER 3

NiagaraAX McQuay Concepts

This section provides conceptual details on the NiagaraAX McQuay driver and its components, including views. These are the main subsections:

- About McQuay to NiagaraAX Architecture
- About the McQuay Network
- McQuay Device Manager
- About the McQuayDevice
- McQuay Point Manager
- About McQuay proxy points

About McQuay to NiagaraAX Architecture

McQuay's MicroTech network architecture permits a "multi-level" approach, where the *single* "level 1" master controller (the OPM) can have multiple "level 2" controllers on a comm trunk, each of which may have one or more "level 3" controllers on a different comm trunk. Level 2 controllers can thus be both a "slave" (always, to the level 1 OPM) as well as a "master" to level 3 controllers. Each MicroTech controller has an "L2" and "L3" (level 2 and 3) address that reflects its network configuration.

In a NiagaraAX integration, this network topology is essentially "flattened" under a station's McQuayNetwork, where the same type of McQuayDevice component represents each MicroTech device, regardless of its position in the McQuay network architecture. Additionally, a McQuayDevice component represents the OPM device itself. Distinction is simply the specified L2 and L3 address for each McQuayDevice component.

Additionally, the standard NiagaraAX network architecture applies, similar to other serial-based polling drivers. See "About Network architecture" in the *Drivers Guide* for more details. For example, real-time data is modeled using McQuay proxy points, which reside under McQuayDevices, which in turn reside under a McQuayNetwork container in the station's DriverContainer (Drivers).

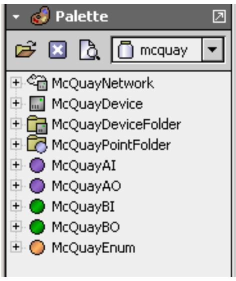
Hierarchically, the component architecture is: network, device, points extension, points. The points extension is the only "device extension" under a McQuayDevice—meaning there are no schedule or history device extensions. You can use the default "McQuay manager" views of the network (and the points extensions under each device) to *manually* add new McQuay device components and McQuay proxy points. Alternatively, you can "drag and drop" these components from the `mcquay` palette.

Note: *Due to the simplicity of the McQuay OPM protocol, there is no "Discover, Add, and Match" (Learn process) for either McQuay devices or the proxy point data in those devices. Instead, you use the* **`New`** *button in manager views, after studying job and McQuay documentation for those MicroTech devices.*

Palette for McQuay driver

The `mcquay` module has a palette, in which you can open to copy components into a station (Figure 3-1).

Figure 3-1 *Components in mcquay palette*



Palette

mcquay

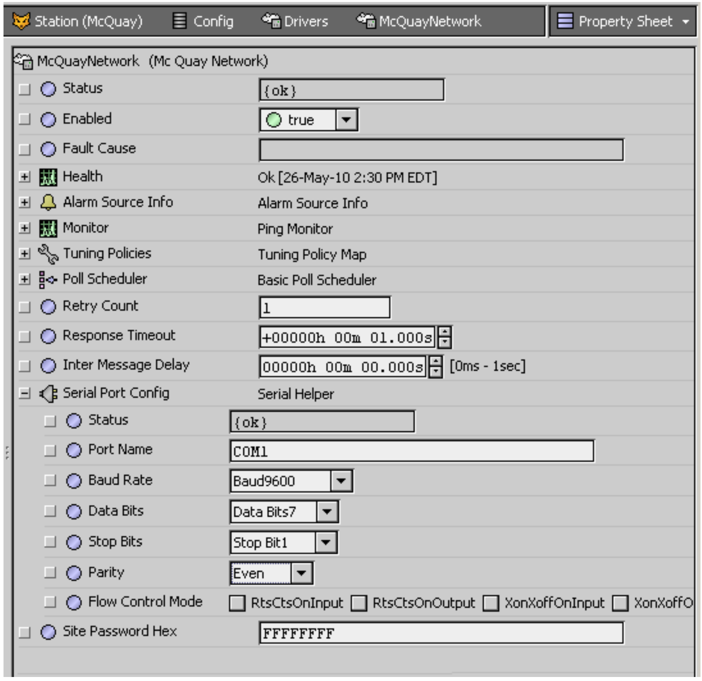
- McQuayNetwork
- McQuayDevice
- McQuayDeviceFolder
- McQuayPointFolder
- McQuayAI
- McQuayAO
- McQuayBI
- McQuayBO
- McQuayEnum

However (as with most NiagaraAX drivers), you *rarely* need to work from the palette. Instead, the various McQuay manager views simplify component creation, enforcing proper component hierarchy.

About the McQuay Network

The McQuayNetwork is the top-level component for the McQuay driver in a station. On its property sheet (Figure 3-2), you configure specific settings for accessing McQuay devices on the attached serial network, where the station operates as the “BAS” (Building Automation System) to that OPM controller.

Figure 3-2 *Property sheet of McQuayNetwork*



Station (McQuay) | Config | Drivers | McQuayNetwork | Property Sheet

McQuayNetwork (Mc Quay Network)

Status	{ok}
Enabled	true
Fault Cause	
Health	Ok [26-May-10 2:30 PM EDT]
Alarm Source Info	Alarm Source Info
Monitor	Ping Monitor
Tuning Policies	Tuning Policy Map
Poll Scheduler	Basic Poll Scheduler
Retry Count	1
Response Timeout	+00000h 00m 01.000s
Inter Message Delay	00000h 00m 00.000s [0ms - 1sec]
Serial Port Config	Serial Helper
Status	{ok}
Port Name	COM1
Baud Rate	Baud9600
Data Bits	Data Bits7
Stop Bits	Stop Bit1
Parity	Even
Flow Control Mode	□ RtsCtsOnInput □ RtsCtsOnOutput □ XonXoffOnInput □ XonXoffO
Site Password Hex	FFFFFFFF

The following sections provide more details on McQuayNetwork properties and slots:

- McQuay-specific network slot
- Common McQuayNetwork slots
- McQuayNetwork action

McQuay-specific network slot

In addition to Common McQuayNetwork slots, the McQuayNetwork contains a key configuration property that permits it to communicate, found near the *bottom* of its property sheet:

- **Site Password Hex**
  The McQuay Open Protocol Site License Password, an 8-character field, in hexadecimal notation.
  Note: *If less than 8 characters are entered, this value is automatically set to all “F”s, that is:* FFFFFFFF

Common McQuayNetwork slots

The McQuayNetwork component includes the typical collection of slots and properties as most other network components. For general information, See “Common network components” in the *Drivers Guide*. The following sections provide additional details:

- McQuayNetwork serial port config properties
- McQuayNetwork status notes
- McQuayNetwork monitor notes
- McQuayNetwork tuning policy notes
- McQuayNetwork poll scheduler notes
- McQuayNetwork message handling properties

McQuayNetwork serial port config properties

The McQuayNetwork has a Serial Port Config container found near the bottom of the property sheet, containing the following properties:

- Status — Either `{ok}` or `{fault}`.
- Port Name — Enter the string for the `COM`*n* serial port (RS-232) used on the host platform. For example, enter `COM2` or `COM3`.
- Baud Rate — Selected from a drop-down list. Typically, the default 9600 baud is used.
- Data Bits — Selectable as 5,6, 7, or 8 bits. Typically, the default 7 data bits is used.
- Stop Bits— Selectable as 1 or 2 bits. Use the default 1 stop bit.
- Parity — Selectable as None, Odd, Even, Mark, or Space. Typically, the default Even is used.
- Flow Control Mode — Do not select any flow control for OPM protocol.

McQuayNetwork status notes

As with most “fieldbus” drivers, the status of a McQuayNetwork is either the normal “ok”, or less typical “down” or “fault” (fault might result from licensing error, or if a non-existent COM port is assigned to Serial Port Config). The Health slot contains historical timestamp properties that record the last network status transitions from ok to any other status. The “Fault Cause” property further explains any fault status.

Note: *As in other driver networks, the McQuayNetwork has an available “Alarm Source Info” container slot you can use to differentiate McQuayNetwork alarms from other component alarms in the station. See “About network Alarm Source Info” in the Drivers Guide for more details.*

McQuayNetwork monitor notes

The McQuayNetwork’s monitor routine verifies child McQuayDevice component(s)—the “pingable” device in the McQuay driver. The default ping frequency is every 5 minutes, and is adjustable. For general information, see “About Monitor” in the *Drivers Guide*.

McQuayNetwork tuning policy notes

The McQuayNetwork has the typical network-level Tuning Policy Map slot with a single default Tuning Policy, as described in “About Tuning Policies” in the *Drivers Guide*. By default, only a single “Default Policy” exists, however, you can add new tuning policies (duplicate and modify) as needed.

If you have multiple tuning policies, you can assign McQuay proxy points as needed to different ones. Under any proxy point’s **`Proxy Ext`** properties, in its **`Tuning Policy Name`** property, select the desired tuning policy.

McQuayNetwork poll scheduler notes

The McQuayNetwork has the typical Poll Scheduler slot, as described in “About poll components” in the *Drivers Guide*. It enables/disables polling, determines fast/normal/slow poll rates, and maintains statistics about proxy extension polls.

By default, a newly created McQuay proxy point uses the “Normal” poll rate. If needed, you can assign proxy points to different poll rates. Under any proxy point’s **`Proxy Ext`** properties, select the rate in its **`Poll Frequency`** property.

McQuayNetwork message handling properties

The McQuayNetwork has several “message handling” network-level properties common among serial drivers, described separately as follows:

- Retry Count — Determines how many retries the communications handler will try to send a message if the initial attempt is unsuccessful. For the OPM protocol, this should normally be set to 1.
- Response Timeout — Specifies the maximum time to wait for a response to a OPM message once sent. If a response is not received before this timeout, the OPM message is resent up to “Retry Count”

times, each of which waits for this timeout period.

- Inter Message Delay — The minimum amount of time to wait between receiving a message on the McQuay bus, and sending the next request. This gives time for some McQuay devices to prepare for receiving messages again. Note that although setting this to a non-zero value has a negative impact on overall throughput, it may be necessary if a "slow-to-turn-around" McQuay device is on the network.
- Unsolicited Receive Handler — Contains a child "Unsolicited Message Count" property that tallies all unsolicited messages sent by the OPM. For informational use only.

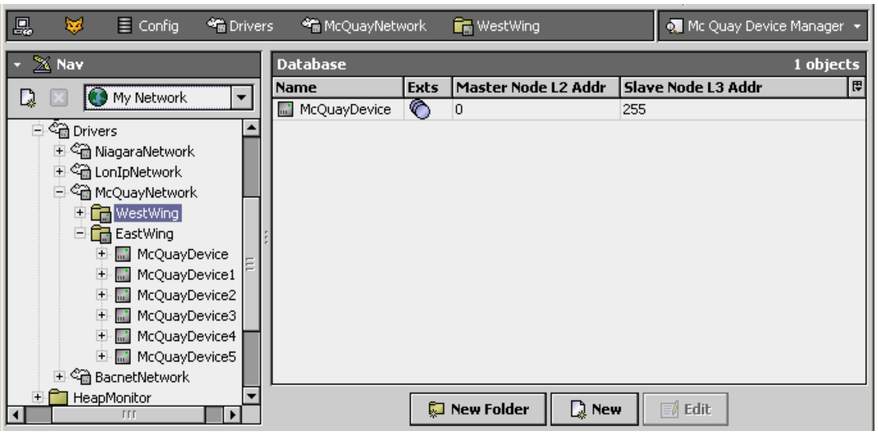
McQuayNetwork action

The McQuayNetwork has a single action: Ping. It reconfirms communications to the OPM and McQuay network.

McQuay Device Manager

The McQuay Device Manager (Figure 3-3) is the default view for the McQuayNetwork, as well as any child McQuayDeviceFolder, and works similar to other device managers that *do not* support online device discovery. See "About the Device Manager" in the *Drivers Guide* for general information.

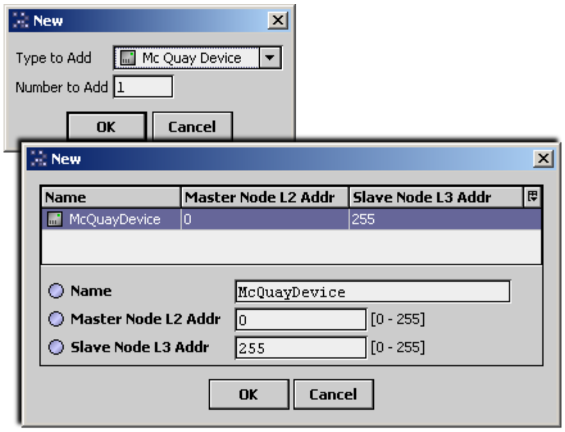
Figure 3-3 *McQuay Device Manager is default view for McQuayNetwork*



Name	Exts	Master Node L2 Addr	Slave Node L3 Addr
McQuayDevice		0	255

As needed, from this view use buttons **`New Folder`** and **`New`** to make McQuayDeviceFolders and McQuayDevices, respectively. The figure below shows the dialogs when making a New device.

Figure 3-4 *New McQuayDevice dialogs*



Type to Add: Mc Quay Device
Number to Add: 1

Name	Master Node L2 Addr	Slave Node L3 Addr
McQuayDevice	0	255

○ Name: McQuayDevice
○ Master Node L2 Addr: 0 [0 - 255]
○ Slave Node L3 Addr: 255 [0 - 255]

Defaults are like those shown in the Figure 3-4 dialog above, where:

- Name is `McQuayDevice` or `McQuayDevice`*n* (must be unique in the current parent container), and
- Master and slave L2 and L3 addresses appear as 0 and 255, respectively.

Change these values as needed to match the various MicroTech controllers you are adding to the network.

After adding one or more new devices, you can also reselect them in this view, and use the **Edit** button to change name and address values in a similar **Edit** dialog, if needed. Also see "About the McQuayDevice" on page 3-9 for more details on this device component.

In this view, go to the points manager of any McQuayDevice shown by double-clicking on its (Points) icon under the **Exts** column. See "McQuay Point Manager" on page 3-10.

About the McQuayDevice

A McQuayDevice represents a McQuay MicroTech controller, and is mapped to a specific device by its combination of its "master node" L2 address and "slave node" L3 address. One McQuayDevice also represents the McQuay OPM device itself.

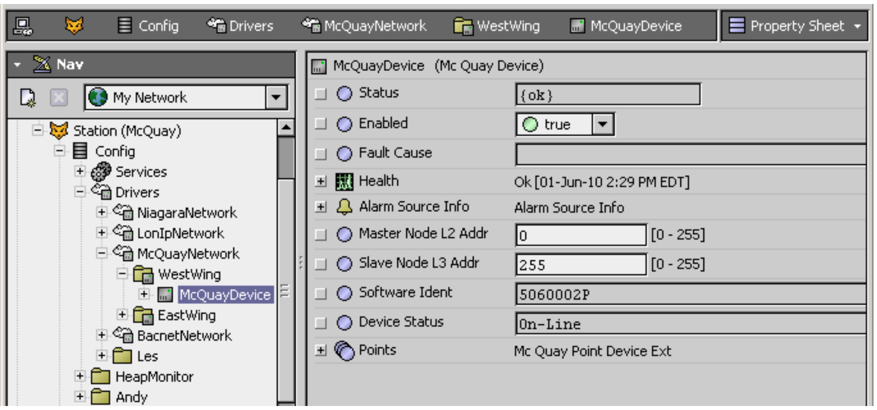
The following subsections provides more details:

- McQuayDevice properties
- McQuayDevice action

McQuayDevice properties

The McQuayDevice property sheet is shown in Figure 3-5.

Figure 3-5 *McQuayDevice property sheet*



McQuayDevice (Mc Quay Device)

Status	{ok}
Enabled	true
Fault Cause	
Health	Ok [01-Jun-10 2:29 PM EDT]
Alarm Source Info	Alarm Source Info
Master Node L2 Addr	0 [0 - 255]
Slave Node L3 Addr	255 [0 - 255]
Software Ident	5060002P
Device Status	On-Line
Points	Mc Quay Point Device Ext

These properties are described as follows:

- **Status, Enabled, Fault Cause, Health, Alarm Source Info**
  These properties operate the same as for a device object in most drivers. see "Common device components" in the *Drivers Guide* for general information on these properties.
- **Master Node L2 Addr**
  The first "L2" part of a McQuay device's network address, from 0 to 255 using decimal notation. In the actual device, this address may be expresses in hexadecimal, from 00 to FF.
- **Slave Node L3 Addr**
  The second "L3" part of a McQuay device's network address, from 0 to 255 using decimal notation. In the actual device, this address may be expresses in hexadecimal, from 00 to FF.
- **Software Ident**
  The McQuay software IDENT (identity) string of the MicroTech controller, for instance "5060002P". This is populated when proper communications to the OPM have been established.
- **Device Status**
  Descriptive device status string such as "On-Line" or "Off-Line". If proper communications to the OPM have been established, this should be "On-Line".
- **Points**
  The standard Points device extension (container) for all data items (attributes) in this McQuay device that need to be polled for data.

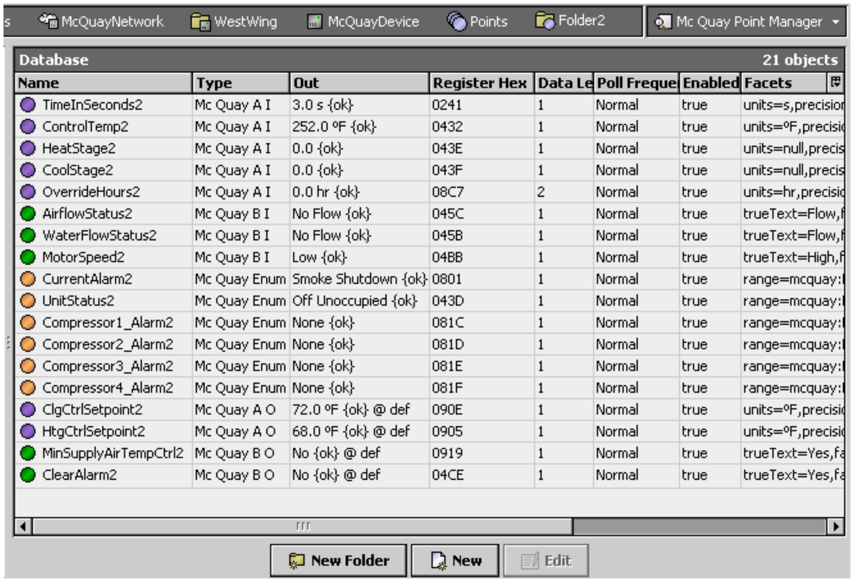
McQuayDevice action

The McQuayDevice has a single action: Ping. It reconfirms communications to the target MicroTech controller through the OPM network.

McQuay Point Manager

The McQuay Point Manager (Figure 3-6) is the default view for the “**Points**” device extension for any McQuayDevice, as well as any McQuayPointFolder, and is similar to other point managers that *do not* support online discovery. See “About the Point Manager” in the *Drivers Guide* for general information.

Figure 3-6 McQuay Point Manager is default view for Points or McQuayPointFolder



Database 21 objects

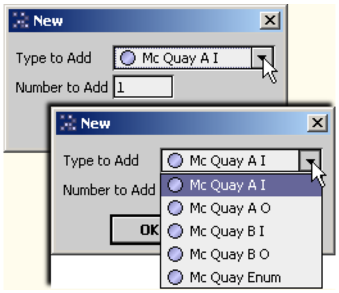
Name	Type	Out	Register Hex	Data Le	Poll Freque	Enabled	Facets
TimeInSeconds2	Mc Quay A I	3.0 s {ok}	0241	1	Normal	true	units=s,precisior
ControlTemp2	Mc Quay A I	252.0 °F {ok}	0432	1	Normal	true	units=°F,precisi
HeatStage2	Mc Quay A I	0.0 {ok}	043E	1	Normal	true	units=null,precis
CoolStage2	Mc Quay A I	0.0 {ok}	043F	1	Normal	true	units=null,precis
OverrideHours2	Mc Quay A I	0.0 hr {ok}	08C7	2	Normal	true	units=hr,precisi
AirflowStatus2	Mc Quay B I	No Flow {ok}	045C	1	Normal	true	trueText=Flow,f
WaterFlowStatus2	Mc Quay B I	No Flow {ok}	045B	1	Normal	true	trueText=Flow,f
MotorSpeed2	Mc Quay B I	Low {ok}	04BB	1	Normal	true	trueText=High,f
CurrentAlarm2	Mc Quay Enum	Smoke Shutdown {ok}	0801	1	Normal	true	range=mcquay:
UnitStatus2	Mc Quay Enum	Off Unoccupied {ok}	043D	1	Normal	true	range=mcquay:
Compressor1_Alarm2	Mc Quay Enum	None {ok}	081C	1	Normal	true	range=mcquay:
Compressor2_Alarm2	Mc Quay Enum	None {ok}	081D	1	Normal	true	range=mcquay:
Compressor3_Alarm2	Mc Quay Enum	None {ok}	081E	1	Normal	true	range=mcquay:
Compressor4_Alarm2	Mc Quay Enum	None {ok}	081F	1	Normal	true	range=mcquay:
ClgCtrlSetpoint2	Mc Quay A O	72.0 °F {ok} @ def	090E	1	Normal	true	units=°F,precisi
HtgCtrlSetpoint2	Mc Quay A O	68.0 °F {ok} @ def	0905	1	Normal	true	units=°F,precisi
MinSupplyAirTempCtrl2	Mc Quay B O	No {ok} @ def	0919	1	Normal	true	trueText=Yes,fa
ClearAlarm2	Mc Quay B O	No {ok} @ def	04CE	1	Normal	true	trueText=Yes,fa

As needed, from this view use buttons **New Folder** and **New** to add McQuayPointFolders and McQuay proxy points, respectively. See the next section “Adding McQuay proxy points” for more details. Additional proxy point details are in “About McQuay proxy points” on page 3-12

Adding McQuay proxy points

The figure below shows the *first* **New** dialog when adding a proxy point from the McQuay Point Manager.

Figure 3-7 New McQuay proxy point (initial) dialog



Type to Add: Mc Quay A I
Number to Add: 1

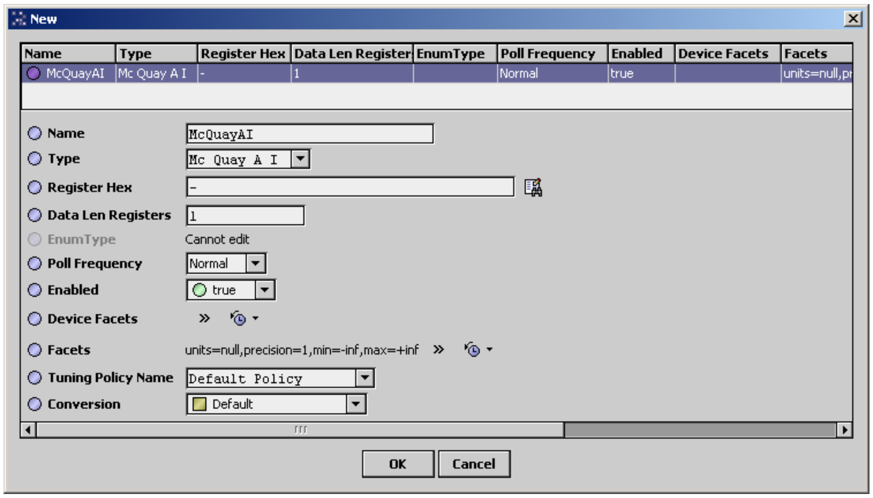
As shown in Figure 3-7, click the **Type to Add** drop-down control to select from these types:

- McQuay AI — use for any numeric read-only data item, or for monitor-only access of any read-write numeric data item.
- McQuay AO — use for any numeric read-write data item that you wish to monitor and control from Niagara, meaning to invoke action or link to input.
- McQuay BI — use for any Boolean (two-state) read-only data item, or for monitor-only access of any read-write Boolean data item.
- McQuay BO — use for any Boolean (two-state) read-write data item that you wish to monitor and control from Niagara, meaning to invoke action or link to input.
- McQuay Enum — for monitor-only access to any one of the following McQuay data items:
  - Current Alarm
  - Unit Status
  - Compressor Alarm 1
  - Compressor Alarm 2
  - Compressor Alarm 3
  - Compressor Alarm 4

The figure below shows the *second* **`New`** dialog when adding a New proxy point.

Note: *A similar dialog appears if using the* **`Edit`** *feature for an existing McQuay proxy point.*

Figure 3-8 *New proxy point (second) dialog*



Name	Type	Register Hex	Data Len Register	EnumType	Poll Frequency	Enabled	Device Facets	Facets
○ McQuayAI	Mc Quay A I	-	1		Normal	true		units=null,pr

○ Name	McQuayAI
○ Type	Mc Quay A I ▾
○ Register Hex	-
○ Data Len Registers	1
○ EnumType	Cannot edit
○ Poll Frequency	Normal ▾
○ Enabled	○ true ▾
○ Device Facets	»
○ Facets	units=null,precision=1,min=-inf,max=+inf »
○ Tuning Policy Name	Default Policy ▾
○ Conversion	Default ▾

Figure 3-8 shows the various fields available and the example default (new point) values. Change values as needed, using the guidelines below.

- **Name**
  Type in to replace default "McQuay*TypeN*" name, where each name must be unique in the parent container for the proxy point.
- **Type**
  Already set, and if in the **`Edit`** dialog, read-only (cannot change if proxy point already created).
- **Register Hex**
  Enter the documented register address for the data item, using hexadecimal notation.
  For example, if "Dew Point Temperature" is listed at address "$042F", enter `042F`
- **Data Len Registers**
  Enter the documented number of registers for the data item, typically either 1 (default) or 2.
  For example, if "Operating Hours" address is listed as "$08C7 - $08C8", enter `2`
- **Enum Type**
  (Unavailable unless a McQuay Enum point) Select from the available enum types.
  Note: *In this case, do not enter (disregard) values for Register Hex and Data Len Registers—these values are automatically resolved upon point creation, overwriting any entered values.*
- **Poll Frequency**
  Accept the default "Normal" poll frequency, or else change to "Slow" or "Fast".
- **Enabled**
  Accept the default "true" (if set to "false" point status is set to disabled, and the item is not polled).
- **Device Facets**
  Accept defaults—usage is not applicable in McQuay driver.
- **Facets**
  Edit as needed for proper unit display and decimal handling of the selected data item.
- **Tuning Policy Name**
  Accept the standard "Default Policy", or if additional tuning policies have been added to the McQuayNetwork, select another tuning policy if needed.
- **Conversion**
  As needed, change from the "Default" conversion for proper handling of the selected data item. Typically, this applies to any:
  - AI or AO point where the McQuay raw data value, typically integer 0-255, requires scaling. In this case, select **`Linear`**, then enter a value in **`Scale`** (replacing the default 1) as documented by McQuay. Usage of **`Offset`** (from default 0 value) is optional.
  - BI or BO point where the McQuay raw data value, typically integer 0 or 1, requires "polarity reversal", meaning that 0 should be evaluated as "true" (On) and 1 should be evaluated as "false" (Off). In this case, select **`Reverse Polarity`**.

About McQuay proxy points

McQuay proxy points are similar to other driver's proxy points, as "point-level" components in the NiagaraAX architecture. See "About proxy points" in the *Drivers Guide* for general information.

The following sections provide driver-specific details about McQuay proxy points:

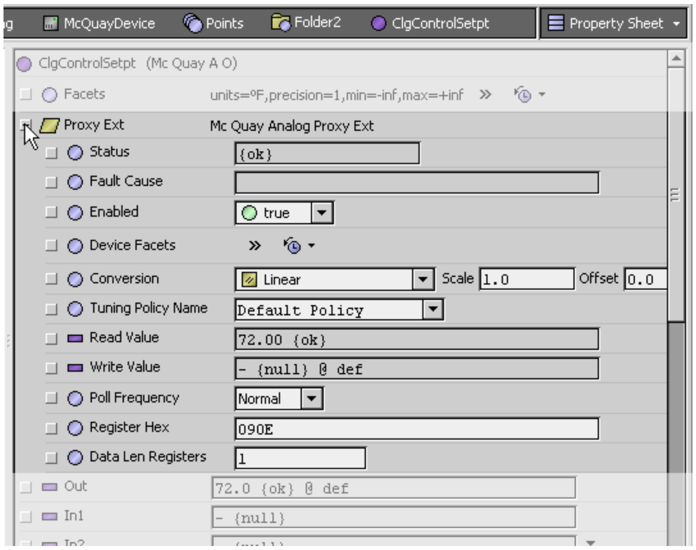
- McQuayProxyExt properties
- McQuay proxy point actions

McQuayProxyExt properties

All McQuayProxyExt types (McQuayAnalogProxyExt, McQuayBinaryProxyExt, McQuayEnumProxyExt) share the same set of configuration properties. Any one of the McQuayProxyExt types is a proxy for one piece of data of interest in a McQuay controller. This single piece of data is specified by its combination of register address and number of address registers (register length).

Figure 3-9 shows the property sheet of an example McQuayProxyExt.

Figure 3-9 *McQuayProxyExt example expanded in property sheet*



ClgControlSetpt (Mc Quay A O)	
Facets	units=°F,precision=1,min=-inf,max=+inf
Proxy Ext	Mc Quay Analog Proxy Ext
Status	{ok}
Fault Cause	
Enabled	true
Device Facets	
Conversion	Linear Scale 1.0 Offset 0.0
Tuning Policy Name	Default Policy
Read Value	72.00 {ok}
Write Value	- {null} @ def
Poll Frequency	Normal
Register Hex	090E
Data Len Registers	1
Out	72.0 {ok} @ def
In1	- {null}

In addition to typical ProxyExt properties, the proxy extension in any McQuay proxy point includes these specific properties:

- **Register Hex**
  Four digit hexadecimal address for the register (or starting register) where the data item resides.
- **Data Len Registers**
  Number of registers used for the data item, typically either 1 (the default) or 2.

Refer to the appropriate McQuay documentation for the target McQuay controller, and enter this needed register address information for each proxy point.

Note: *The single exception to this is for a McQuay Enum point, where you do not enter register address information, but instead select from the several available "Enum Types".*

For related details, see "Adding McQuay proxy points" on page 3-10.

McQuay proxy point actions

Unlike proxy points in some other drivers, no "special" actions are available on McQuay proxy points, or on their ProxyExt (McQuay*Type*ProxyExt). Only point types McQuayAO and McQuayBO have actions, which are the same actions available on standard NumericWritable and BooleanWritable components.

As such, if needed you can go to the slot sheet of these proxy points to edit display names for actions. Or, edit config flags for actions to hide or change access (permission) level, if needed.

CHAPTER 4

McQuay Plugin Guides

Plugins provide *views* of components, and can be accessed many ways—for example, double-click a component in the tree for its *default* view. In addition, you can right-click a component, and select from its **`Views`** menu. For summary documentation on any view, select **`Help > On View`** (F1) from the Workbench menu, or press F1 while the view is open.

Summary information is provided here about the different McQuay views.

McQuay Plugin Guides Summary

Summary information is provided on views specific to components in the `mcquay` module, with views listed in alphabetical order as follows:

- McQuay Device Manager
- McQuay Point Manager

mcquay-McQuay Device Manager

The **`McQuay Device Manager`** is the default view of a McQuayNetwork. Use the McQuay Device Manager to add, edit, and access McQuay device components (McQuayDevices). As with a few other device managers (say, Modbus drivers), there is no online “discovery” of devices. Instead, you manually add new devices using the **`New`** button (or by dragging McQuayDevices from the `mcquay` palette).

For general information, see “About the Device Manager” in the *Drivers Guide*. For specific details, see “McQuay Device Manager” on page 3-8.

Added McQuayDevices appear in the view’s table. By default, the following columns appear in this view:

- **Name**
  Name of the device-level component that represents the McQuay device, or the McQuayDeviceFolder for organizing McQuayDevices.
- **Exts**
  Shows an icon for the only (points) de vice extension of a McQuayDevice—double-click the icon for the McQuay Points Manager.
- **Master Node L2 address**
  Reflects the first “L2” part of a device’s McQuay network address, from 00 to 255 in decimal.
- **Slave Node L3 address**
  Reflects the second “L3” part of a device’s McQuay network address, from 00 to 255 in decimal.

In addition, using the table options control, the following additional data columns are available:

- **Path**
  Station path of the device or device folder component, relative to the root.
- **Type**
  Niagara type of component, as either a McQuay Device Folder or a McQuay Device component.

mcquay-McQuay Point Manager

Use the McQuay Point Manager to add, edit, and access McQuay proxy points under the Points extension of a selected McQuayDevice, or in a McQuayDeviceFolder. The McQuay Point Manager is the default view on both these components. To view, *double-click* the Points extension or McQuay-PointFolder, or right-click and select **`Views > McQuay Point Manager`**.

For more details, see "McQuay Point Manager" on page 3-10.

By default, the following columns appear in this point manager:

- **Name**
  Niagara name of the McQuay proxy point or points folder.
- **Type**
  Niagara type of component, as either a McQuay Point Folder (for a folder) or a type of McQuay proxy point (either McQuay AI, McQuay AO, McQuay BI, McQuay BO, or McQuay Enum).
- **Out**
  Current last polled value (out slot) of a proxy point, reflecting status and facets.
- **Register Hex**
  Address register (in hexadecimal format) in the McQuay device where a data item resides.
- **Data Len Registers**
  Number of registers included for a data item (typically either 1 or 2).
- **Poll Frequency**
  Defined polling frequency, as either Normal (default), Slow, or Fast.
- **Enabled**
  Whether enabled (true, default) or disabled (false).
- **Facets**
  Assigned units (AI or AO), trueText and falseText (BI or BO), or enum range (Enum).
- **Tuning Policy Name**
  Assigned tuning policy (Default Policy, or other if available and assigned).

In addition, using the table options control, the following additional data columns are available:

- **Path**
  Station path of the proxy point or point folder component, relative to the root.
- **Enum Type**
  If the point type is McQuay Enum, reflects the selected McQuay enum type.
- **Device Facets**
  Not applicable (will be blank, as all raw McQuay data is integer and "unitless").
- **Conversion**
  Niagara conversion type used, such as Linear, Default, or Reverse Polarity.
- **Fault Cause**
  If the point is in fault, explains why.

CHAPTER 5

McQuay Component Guides

These component guides provides summary help on McQuay components.

McQuay Component Guides Summary

Summary information is provided on components specific to the mcquay module, listed in alphabetical order as follows:

- McQuayAI
- McQuayAO
- McQuayAnalogProxyExt
- McQuayBI
- McQuayBO
- McQuayBooleanProxyExt
- McQuayDevice
- McQuayDeviceFolder
- McQuayEnum
- McQuayEnumProxyExt
- McQuayNetwork
- McQuayPointDeviceExt
- McQuayPointFolder

mquay-McQuayAI

McQuayAI provides read-only access to a numeric data item in a McQuay device. It is similar to a standard NumericPoint, but with a McQuayAnalogProxyExt.

Select a McQuayAI for any numeric read-only data item in a McQuay device.

The McQuayAI is selectable when adding new proxy points using the McQuay Point Manager. It is also available for copy (or drag and drop) from the mcquay palette.

mquay-McQuayAO

McQuayAO provides read-write access to a numeric data item in a McQuay device. It is similar to a standard NumericWritable, but with a McQuayAnalogProxyExt. Refer to the section “About writable points” in the *User Guide* for general details on writable points.

Select a McQuayAO for any numeric read-writable data item in a McQuay device that you also wish to control from Niagara (by invoking an action, and/or by linking to an input). If control from Niagara is not needed, alternatively you can select a McQuayAI proxy point instead.

The McQuayAO is selectable when adding new proxy points using the McQuay Point Manager. It is also available for copy (or drag and drop) from the mcquay palette.

mcquay-McQuayAnalogProxyExt

McQuayAnalogProxyExt is the proxy extension for either a McQuay AI (read-only) proxy point or a McQuay AO (writable) proxy point. Like any McQuay*Type*ProxyExt, it represents a single piece of data defined by a combination of the register address and the data register length, and has standard proxy extension properties such as Status and Enabled, among others (see "ProxyExt properties" in the *Drivers Guide* for related details). For additional details, see "About McQuay proxy points" on page 3-12.

mcquay-McQuayBI

McQuayBI provides read-only access to a Boolean (two state) data item in a McQuay device. It is similar to a standard BooleanPoint, but with a McQuayBooleanProxyExt.

Select a McQuayBI for any Boolean read-only data item in a McQuay device.

The McQuayBI is selectable when adding new proxy points using the McQuay Point Manager. It is also available for copy (or drag and drop) from the `mcquay` palette.

mcquay-McQuayBO

McQuayBO provides read-write access to a Boolean (two state) data item in a McQuay device. It is similar to a standard BooleanWritable, but with a McQuayBooleanProxyExt. Refer to the section "About writable points" in the *User Guide* for general details on writable points.

Select a McQuayBO for any Boolean read-writable data item in a McQuay device that you also wish to control from Niagara (by invoking an action, and/or by linking to an input). If control from Niagara is not needed, alternatively you can select a McQuayBI proxy point instead.

The McQuayBO is selectable when adding new proxy points using the McQuay Point Manager. It is also available for copy (or drag and drop) from the `mcquay` palette.

mcquay-McQuayBooleanProxyExt

McQuayBooleanProxyExt is the proxy extension for either a McQuay BI (read-only) proxy point or a McQuay BO (writable) proxy point. Like any McQuay*Type*ProxyExt, it represents a single piece of data defined by a combination of the register address and the data register length, and has standard proxy extension properties such as Status and Enabled, among others (see "ProxyExt properties" in the *Drivers Guide* for related details). For additional details, see "About McQuay proxy points" on page 3-12.

mquay-McQuayDevice

McQuayDevice is the "device-level" component in a McQuayNetwork, and represents a specific McQuay MicroTech controller. It contains the two-part network address (L2 and L3) necessary for the driver to communicate with the device through the McQuay OPM. A McQuayDevice has a Points device extension (McQuayPointDeviceExt) that contains all proxy points for polling.

A McQuayDevice has the standard device component properties such as status and enabled (see "Common device components" in the *Drivers Guide* for general information). The default view for a McQuayDevice is its property sheet. For more details, see "About the McQuayDevice" on page 3-9.

A "Ping" action is available to send ping monitor request to verify device "health. "

mcquay-McQuayDeviceFolder

McQuayDeviceFolder is the McQuay driver implementation of a folder under a McQuayNetwork. Usage is optional. Each McQuayDeviceFolder has its own McQuay Device Manager view.

You can use the **`New Folder`** button in the McQuay Device Manager to add a McQuayDeviceFolder. It is also available in the **`mcquay`** palette.

mcquay-McQuayEnum

McQuayEnum provides read-only access to one of several pre-defined multistate (enumerated) data item in a McQuay device. It is similar to a standard EnumPoint, but with a McQuayEnumProxyExt.

Select a McQuayEnum for monitor access to any one of the following McQuay data items:

- Current Alarm
- Unit Status
- Compressor Alarm 1
- Compressor Alarm 2
- Compressor Alarm 3
- Compressor Alarm 4

Where the selected Enum Type automatically configures the proxy point's Facets with the appropriate range and units.

mcquay-McQuayEnumProxyExt

McQuayEnumProxyExt is the proxy extension for a McQuay Enum (read-only) proxy point. (Currently, there are no writable Enum points in the McQuay driver). Like any McQuay-*Type*ProxyExt, it represents a single piece of data defined by a combination of the register address and the data register length, and has standard proxy extension properties such as Status and Enabled, among others. See "ProxyExt properties" in the *Drivers Guide* for related details. For additional details, see "About McQuay proxy points" on page 3-12.

mcquay-McQuayNetwork

McQuayNetwork is the top-level component for the McQuay driver in a station. It provides configuration parameters necessary for the driver to communicate with a McQuay OPM device, and through that OPM to other networked McQuay MicroTech controllers.

The McQuayNetwork component has the typical collection of slots and properties as most other network components. For details, See "Common network components" in the *Drivers Guide*. In addition, the McQuayNetwork has properties unique to operation in a McQuay system. For more details, see "About the McQuay Network" on page 3-6.

mcquay-McQuayPointDeviceExt

McQuayPointDeviceExt (default name `Points`) is the container for McQuay proxy points under a McQuayDevice. Proxy points represent data items in the device that need to be polled for data. It operates as in most other drivers; see "About the Points extension" in the *Drivers Guide* for general information. The default and primary view for the Points extension is the McQuay Point Manager.

mcquay-McQuayPointFolder

McQuayPointFolder is an optional container for McQuay proxy points. You can use the **`New Folder`** button in the McQuay Point Manager view to add a McQuayPointFolder. It is also available in the **`mcquay`** palette. Each McQuayPointFolder has its own McQuay Point Manager view.