

GCM Driver User Guide

GCM Driver User Guide

Table of Contents

Installation	1
Requirement	2
Quick Start	3
GCM Component Guide	3
GcmNetwork	4
GcmDevice	6
GcmProxyExt	7
Gcm View	9
Gcm Device Manager	9
Gcm Point Manager	11
Special Consideration	13
Gcm Driver Tuning Performance	15
Gcm Driver Terminal Mode	16
Installing The Client-Side Software	16
Connecting To The GCM	18
Licensing	22

Installation

Install `ibmsGcm.jar` on the computer where Niagara AX Workbench will run. To install, place a copy of the file in the modules directory of your Niagara AX installation. This is typically `C:\Niagara\Niagara-3.n.nnn\modules`.

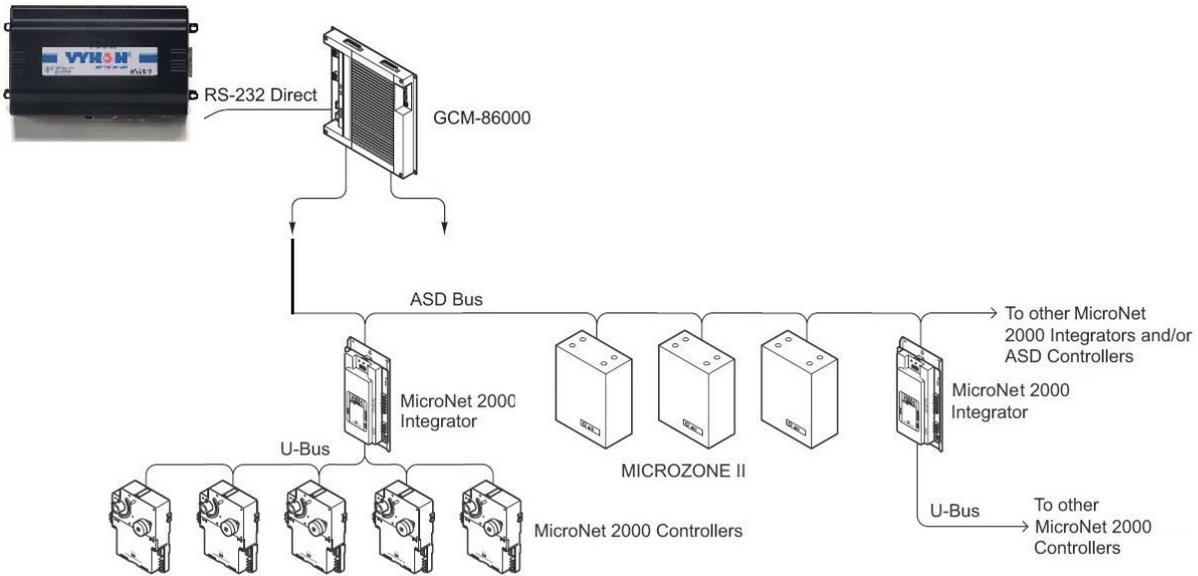
Install `ibmsGcm` module on the target station. Using a Niagara AX workbench where the module has already been installed, connect to the stations platform service. Go to the software manager and install `ibmsGcm`.

Apart from installing the *3.n.nnn* version of the Niagara distribution files in the JACE, make sure to install the `ibmsGcm` module too (if not already present, or upgrade if an older revision). For more details, see “About the Commissioning Wizard” in the JACE Niagara AX Install and Startup Guide.

Following this, the station is now ready for GCM software integration, as described in the rest of this document.

Requirement

- Workbench 3.6.xx or higher.
 1. Serial communication RS232 Port. (Required “serial” feature in the license).
- AX platform JACE support:
 1. JACE 2xx.
 2. JACE 4xx.
 3. JACE 5xx.
 4. JACE 6xx.
 5. JACE 7xx.
- GCM device support:
 1. GCM-84221 (Metal casing)
 2. GCM-86120
 3. GCM-LNC-100



Quick Start

- Follow the installation and configuration instructions preceding this.
- This driver use serial communication RS232 Port. (Required “serial” feature in the license).
- Open the `ibmsGcm` palette and find the `GcmNetwork` object.
- Paste a `GcmNetwork` object under the driver’s node in your station.
 - Edit the Serial Port Config properties as needed.
- Open the `GcmDeviceManager` view by double-clicking the `GcmNetwork` object just added to the station.
 - Add the `gcm` device and setup the host address with the `gcm` address.
- Once a device is added, navigate to the “points” folder under the device and double click the point’s folder to display the `GcmPointManager` view.
 - Press the “Discover” button.
 - Select and add GCM points (`gcm` attributes) to your database.

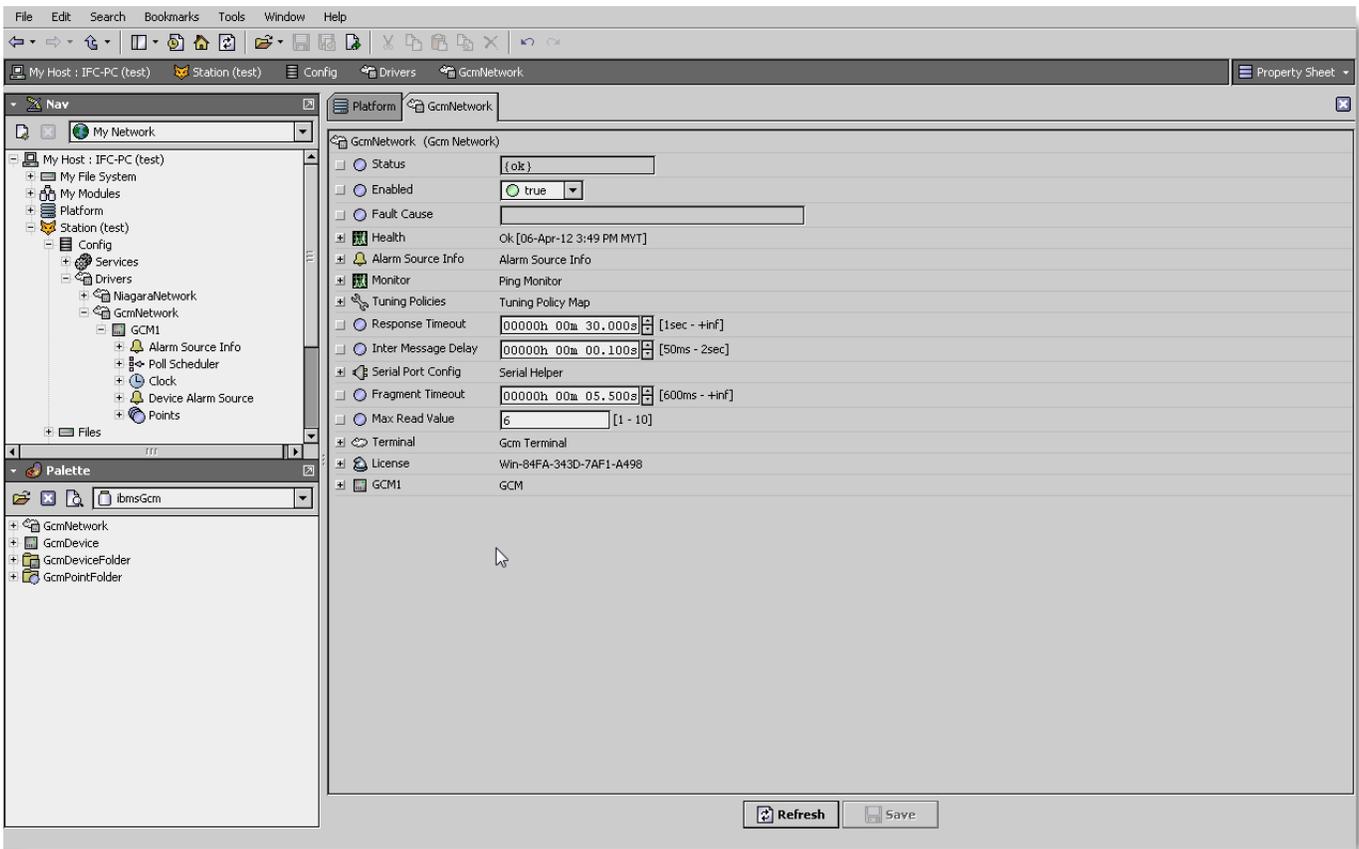
GCM Component Guide

- [GcmNetwork](#)
- [GcmDevice](#)
- [GcmProxyExt](#)

GcmNetwork

The GcmNetwork provides all the configuration parameters necessary to allow the driver to communicate with a network of GCM devices. The GcmNetwork can be thought of as the “host” GCM device in network of GCM devices.

The GcmNetwork is the "network-level" component in the NiagaraAX architecture. It has the standard network component properties such as status and enabled (see "Driver Architecture / Common network components" in the NiagaraAX Users Guide for more information), as well as properties unique to operation as a host in a GCM system.



NOTE: In the following properties, the properties are inherited from the base AX driver classes, and as such are only touched on here. For a full explanation, refer to the AX Driver Architecture document.

- Status – The status of the network will normally be {ok}. A value of {fault} usually indicates an incorrect Serial Port Config setting.

- Enabled – Enables or Disables the Gcm Driver.
- Fault Cause – if the “Status” is fault, the fault cause will appear here. For instance, if a non-existent COM port is configured into the “Serial Port Config” property, then this will say “Could not enable serial communication (javax.baja.serial.PortClosedException)”.
- Monitor – container for monitor (ping) properties.
- Tuning Policies - A container for tuning policies which determines how and when proxy points are read and written.
- Response Timeout - Specifies the maximum time to wait for a response to a GCM message is sent. If a response is not received before the timeout.
- Inter Message Delay – The minimum amount of time to wait between receiving a message on the GCM bus, and sending the next request. This gives time for some GCM devices to prepare for receiving messages again. Note that this setting this value to non-zero has a negative impact on overall throughput, but may be necessary if a slow-to-turn-around GCM device is on the network, minimum setting can be set is 100 ms.
- Serial Port Config – A container for the serial port properties.
 - Status - {ok} or {fault}
 - Port Name – Common name of a serial port, such as “COM1”
 - Baud Rate – 9600 is a standard baud rate.
 - Data Bits – select 5, 6, 7, or 8 bits – Use 8 bits for GCM protocol.
 - Stop Bits – select 1 or 2 bits – Use 1 stop bit for GCM protocol.
 - Parity – select None, Odd, Even, Mark, or Space – Use None for GCM protocol.
 - Flow Control Mode – Do not select any flow control for GCM protocol.

Note : Using a standard “null-modem” or “straight-through” cable connect the JACE serial port used by the GCM network to be configure GCM panel port. If using a null-modem cable, set the GCM port switch to “DTE”. If using a straight-thru cable, set the GCM port switch to “DCE”.

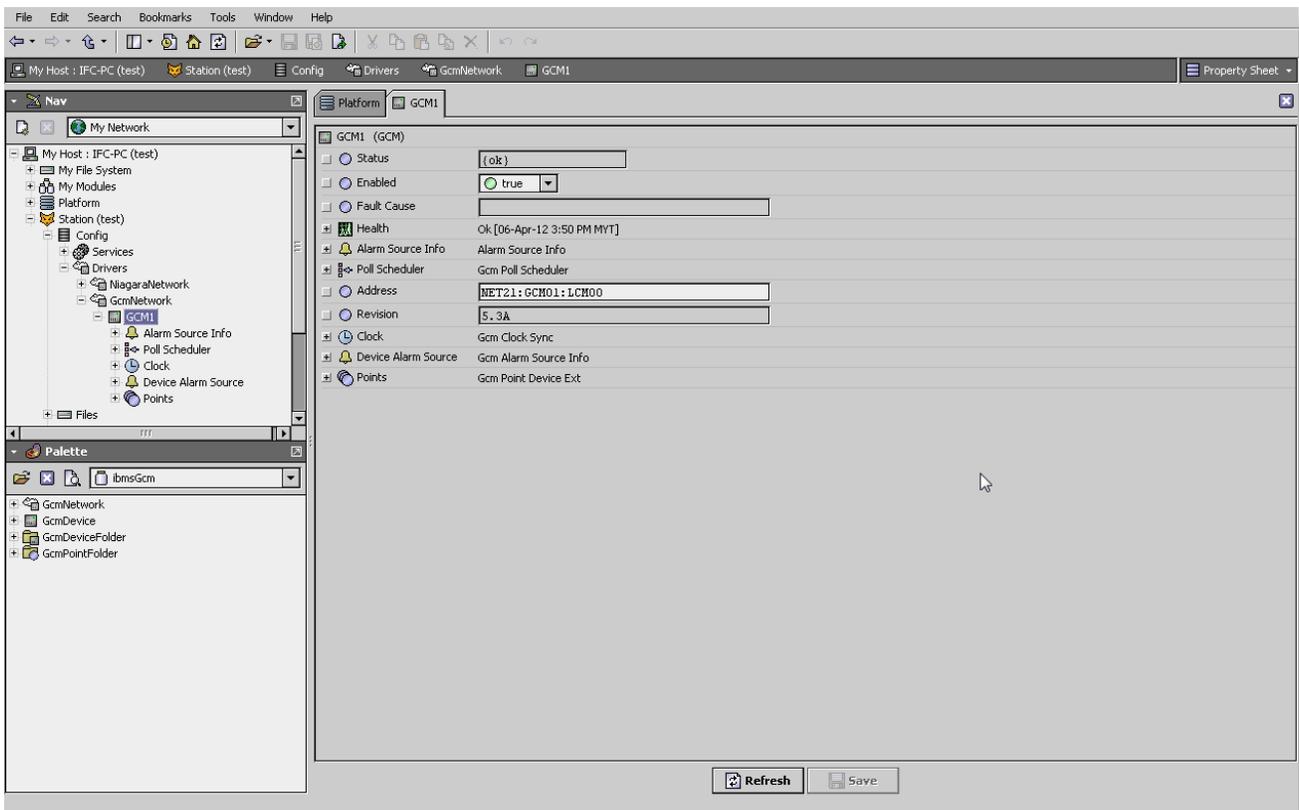
- Timeout Gcm Fragment- Specifies the maximum time to wait for a response to a GCM fragment message is sent. If a response is not received before the timeout.
- Max Read Values – The maximum number of GCM block per-request.
- Terminal – Specifies the information about the GCM terminal mode.
 - Enabled – Enabled/Disabled the tunnel options.
 - Connections – indicated the current total count of terminal access.
 - Status - {ok} or {fault}.

- Identifier – The tunnel identifier to access. The GCM network will setup the identifier automatically when started. This can be access by using Windows hyper terminal software, PSI software or any software that support standard hyper terminal protocol.
- License - The GCM license information. The driver could run with the time trial license in 2 hour. If the time trial expired the driver will stop working and user required to restart the station to make it work again.

GcmDevice

The GcmDevice provides all the configuration parameters necessary to allow the driver to communicate with a given Gcm device. The GcmDevices are always children of a GcmNetwork. GcmDevices also serve as a container for all of the data points that need to be polled for data (a Gcm attribute under a channel in a device).

The GcmDevice is the "device-level" component in the NiagaraAX architecture.



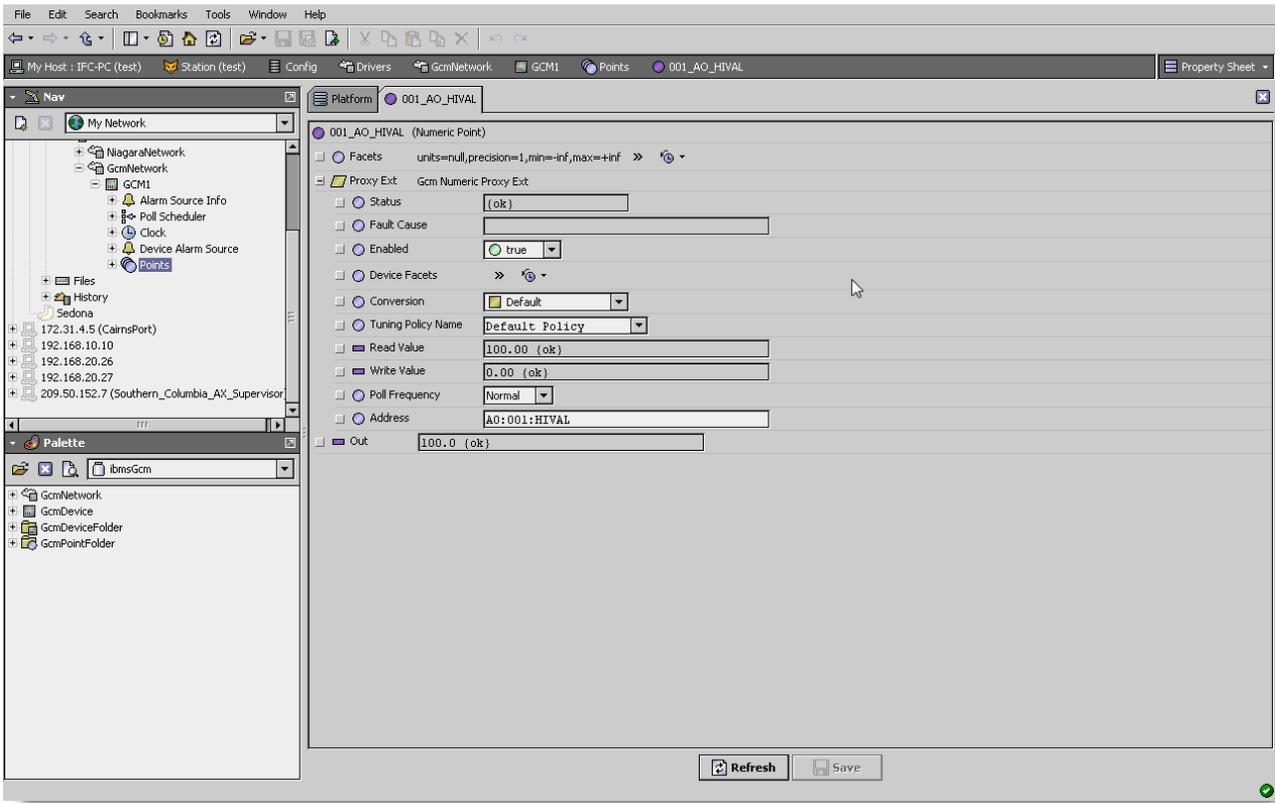
NOTE: In the following properties, the properties are inherited from the base AX driver classes, and as such are only touched on here. For a full explanation, refer to the AX Driver Architecture document.

- Status – The status of the device. Will normally be {ok}. A value of {down} indicates that the last ping to the device was not answered (the “Report Exception Message” is used as the ping message).
- Enabled – Enables or Disables communication to the associated device from Gcm Driver.
- Fault Cause – if the “Status” property value is {fault}, the fault cause is displayed here.
- Heath – contains metadata about the health of this device on the network:
 - Down – indicates if this device is down – should be false under normal operation.
 - Alarm – indicates if this device is in alarm – should be false under normal operation
 - Last OK Time – the last time of successful communication to this device
 - Last Fail Time – the last time of unsuccessful communication to this device
 - Last Fail Cause – the reason of the last comm. Failure (example – “bad checksum) .
- Alarm Source Info – configuration items for alarms generated from the ping process (device up and device down events)
- Gcm Alarm Source – configuration data to apply to alarms that originate in GCM devices and are retrieved using the “Report Exception” message. GCM native alarms are normalized and passed to the AX alarming subsystems which then display manage them. Acknowledgments are in turn transmitted back down to the GCM device that originated them.
- Poll Scheduler - The basic poll scheduler enables/disables polling, determines the fast/normal/slow poll rates, and maintains statistics about proxy extension polls.
- Address – Gcm Host Address and Gcm Address, this must match with the GCM device to communicate.
- Sync Clock – Enabled / Disabled the GCM clock auto synchronization.
- Points – a container for all data items (attributes) in this device which need to be polled for data.

GcmProxyExt

The GcmProxyExt types take on the readable-writable personality of the control point they are attached to. For example, a GcmProxyExt, when used as an extension on a NumericPoint has “read only” functionality, but when used on as an extension on a NumericWritable can read and write the attribute values.

The GcmProxyExt types are the "point-level" component in the NiagaraAX architecture.



NOTE: In the following properties, the properties are inherited from the base AX driver classes, and as such are only touched on here. For a full explanation, refer to the AX Driver Architecture document.

- Status – The status of the device. Will normally be {ok}. A value of {down} indicates that the last ping to the device was not answered (the “Report Exception Message” is used as the ping message).
- Fault Cause – if the “Status” property value is {fault}, the fault cause is displayed here.
- Enabled – Enables or Disables communication to the associated device from Gcm Driver
- Device Facets – contains metadata about the health of this device on the network:
- Conversion
- Tuning Policy Name
- Read Value
- Write Value

- Poll Frequency – the frequency to poll this point in the controller. Choices are either fast/normal/slow. Fast/Normal/Slow rates are determined by the “Poll Scheduler” property on the GcmNetwork component.
- Block Type, Block Name, attribute – represent the identification for the Gcm block to communicate from the Gcm devices.

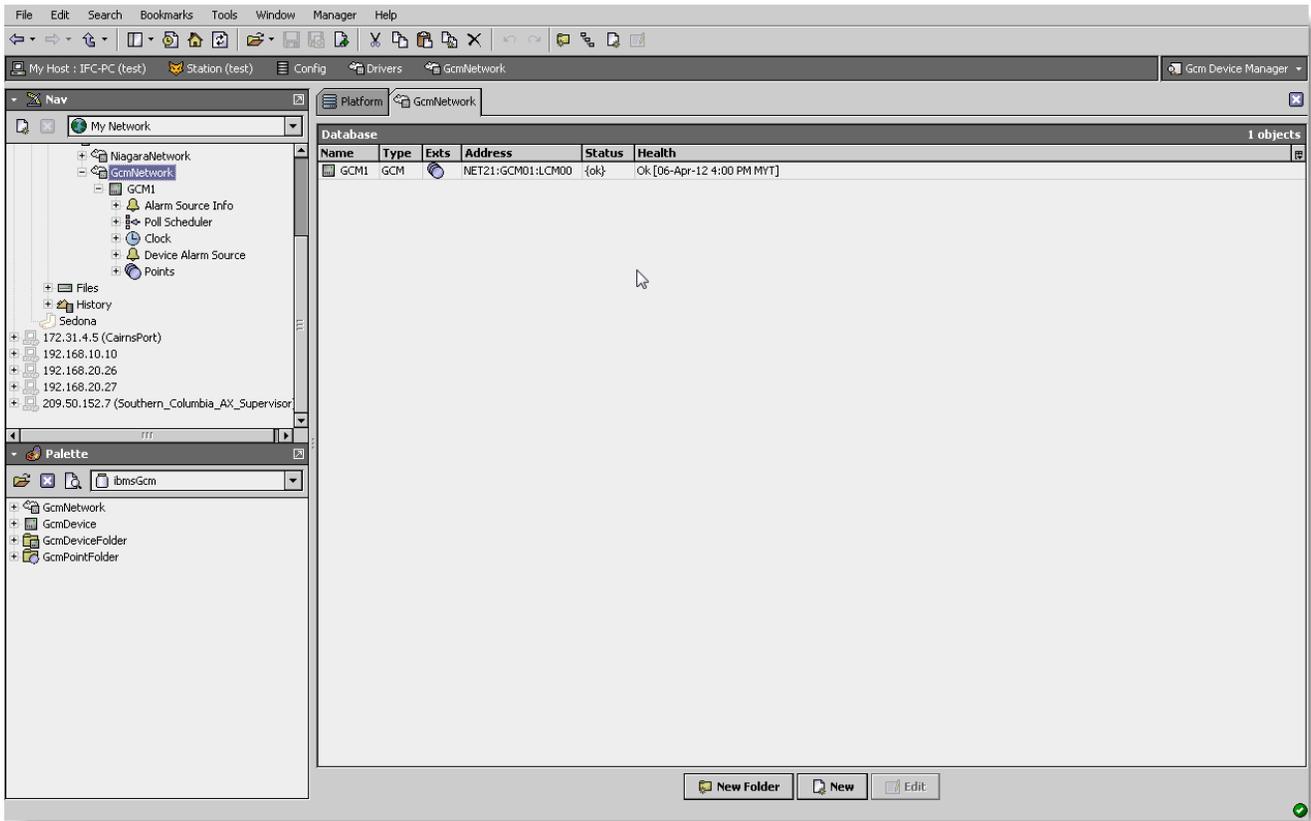
Gcm View

- [Gcm Device Manager](#)
- [Gcm Point Manager](#)
- [Gcm Histories Manager](#)

Gcm Device Manager

The Gcm Device Manager is the default view when you double-click on a Gcm Network in the Nav tree. This manager view provides a quick and easy way to display.

The Gcm Device Manager is the default view for any Gcm Network container. The Gcm Device Manager is a table-based view, where each row represents a unique device. When building a network in the station, you use this view to create, edit, and delete device-level components. Below is an example Gcm Device Manager view.

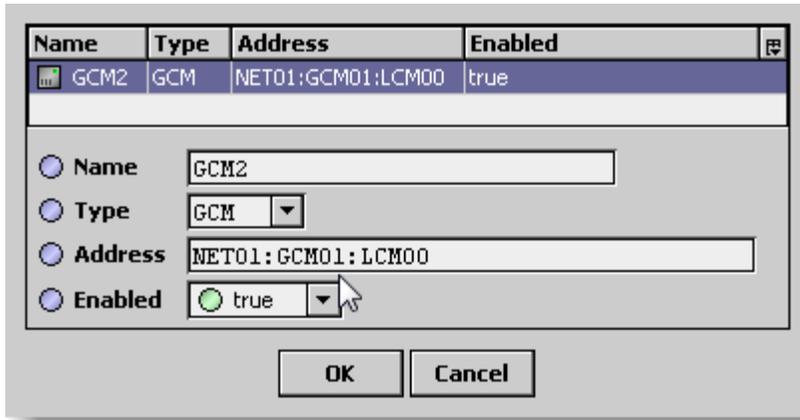


The view above shows a typical Gcm Device Manager view.

The “New Folder”, “New”, and “Edit” buttons are not unique to the Gcm Device Manager, and are explained in the “Niagara AX User’s Guide” in the “Driver Architecture” section. The “Match” button is not used for the Gcm driver.

You can now add the devices to the station database by clicking the “Add” button. This will pop up the “Add” dialog box:



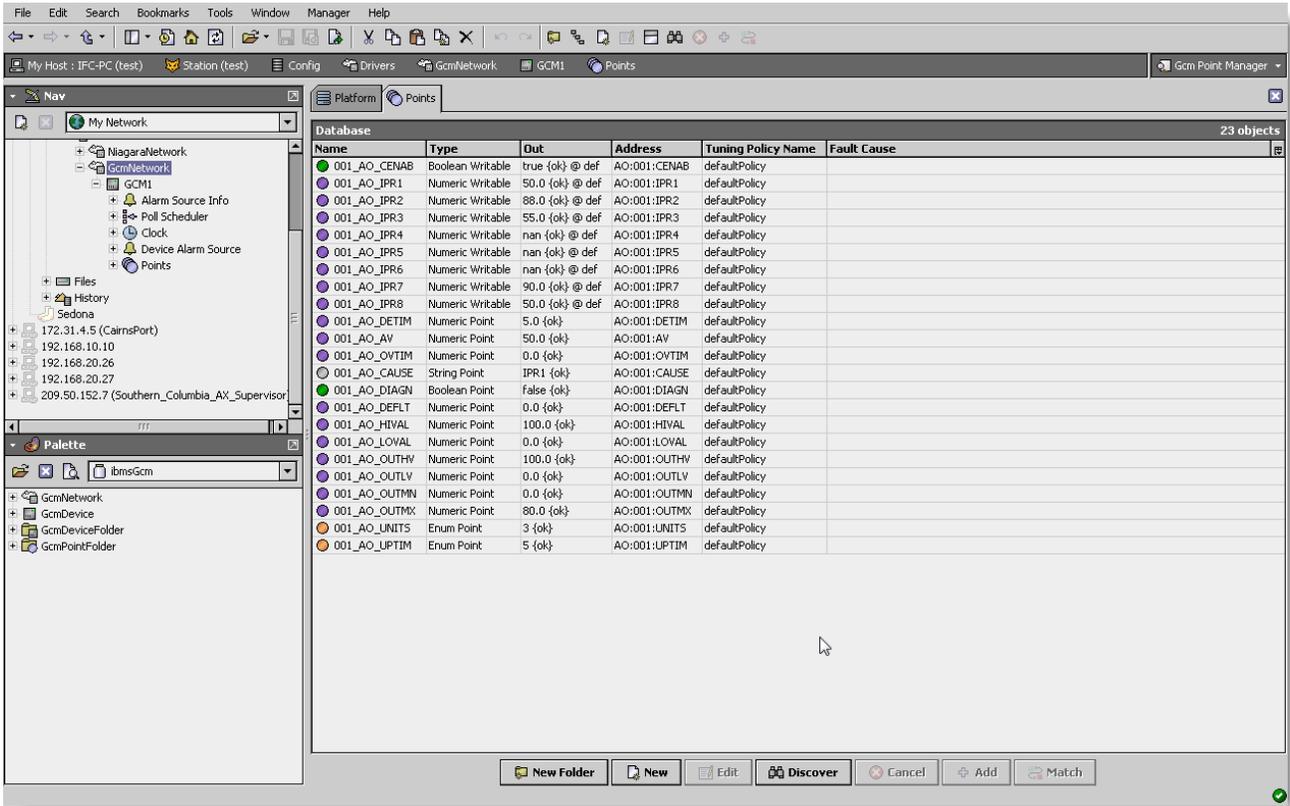


The “Add” dialog box affords you the opportunity to tweak the display name, enabled state, and/or address of each of the selected devices. Click the “OK” button to add the devices to the database, or click “Cancel” to bail out.

Gcm Point Manager

The Gcm Point Manager is the default view when you double-click on a “points” folder (a GcmPointDeviceExt type folder) under a GcmDevice in the Nav tree. This manager view provides a quick and easy way to display and learn Gcm points in a Gcm device.

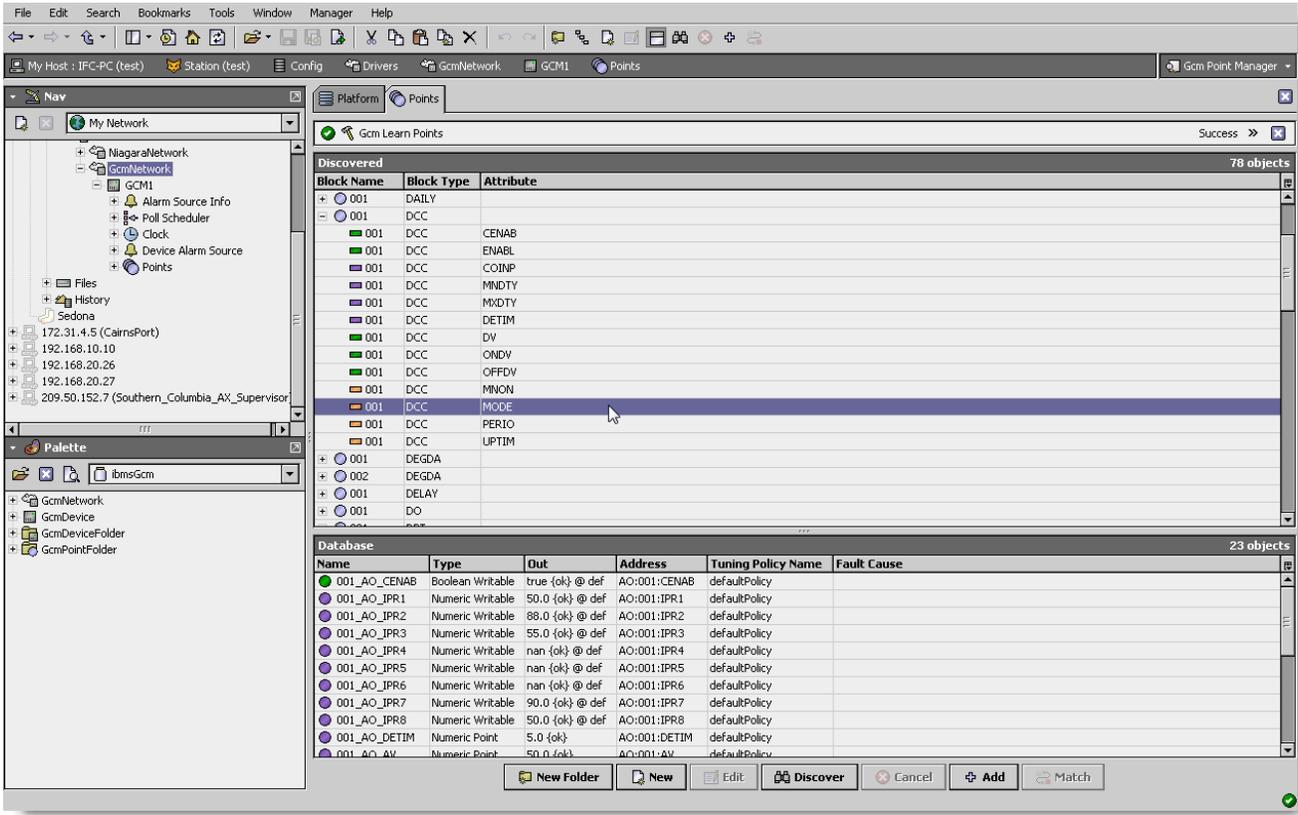
The Gcm Point Manager is the default view for any GcmPointDeviceExt container. The Gcm Point Manager is a table-based view, where each row represents a unique Gcm attribute within a device. Below is an example Gcm Point Manager view.



The “New Folder”, “New”, and “Edit” buttons are not unique to the Gcm Device Manager, and are explained in the “Niagara AX User’s Guide” in the “Driver Architecture” section. The “Match” button is not used for the Gcm driver.

The “Discover” button implements functionality that is unique and tailored to discovering Gcm devices. By clicking the “Discover” button, the “learn” mode of the manager is invoked (the panes will be split, and a “discovery” table will be displayed in the top pane).

When you click Discover, the Gcm Device will be searched for all attributes, and the results will be displayed in the top pane of the following split-pane format:



Note that the first (default) attribute of each channel is listed, with a “+” mark in front of the attribute number if there are additional attributes in that block. To access the additional attributes, click on the “+” sign to expand:

Single or multiple attributes can be added as control points with GcmProxyExt extensions by selecting the discovered row(s) in the top pane, and clicking add. Doing so will cause the “Add” dialog box to appear:

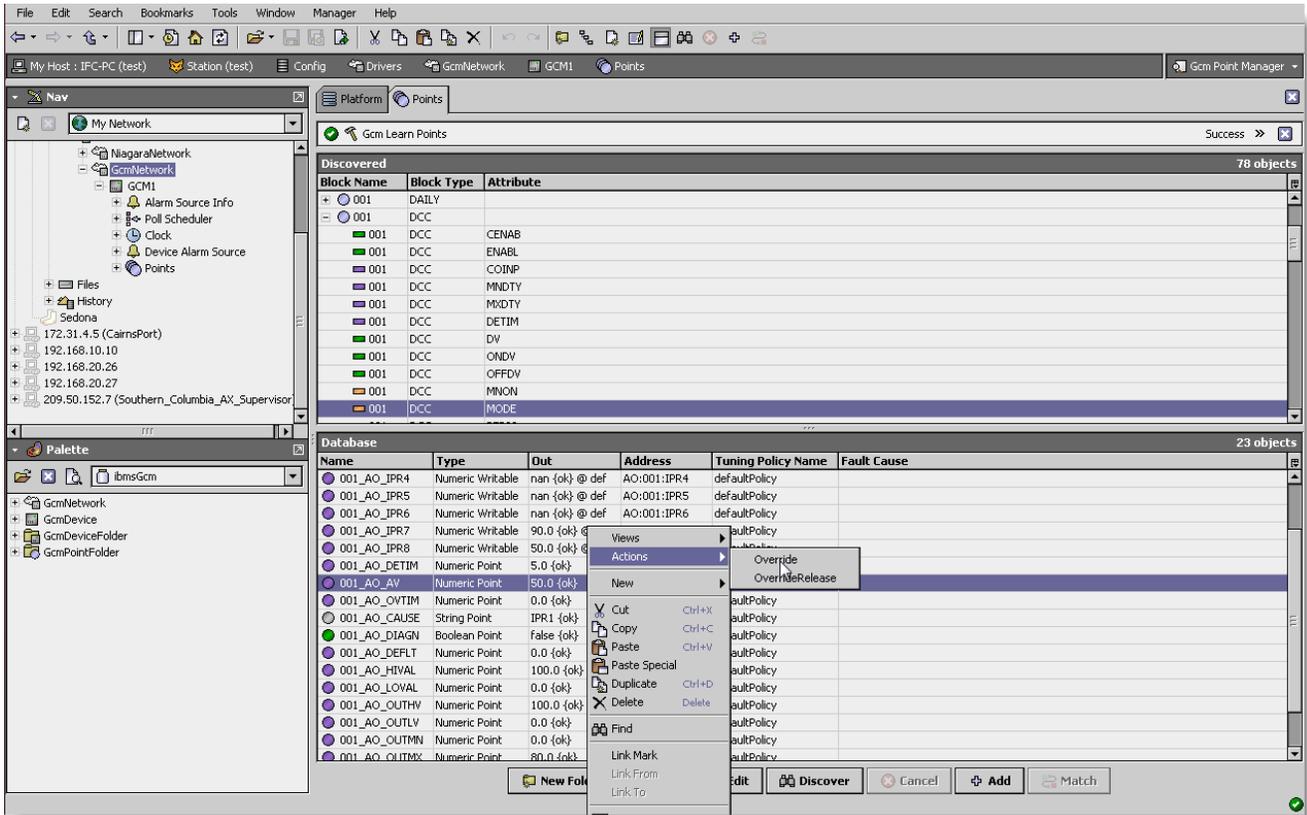
Once the point(s) are satisfactorily edited, click “OK” to create the proxy points corresponding to the block attribute.

Special Consideration

Whenever a Gcm proxy extension is added or configured to have an Attribute property value and the base. Type is none writable Point as shown below, there is additional special behavior attached to the point:

In this case, the additional behavior is the addition of a command attach for override value, clear override change value to the point.

NIAGARA AX GCM DRIVER USER GUIDE



Gcm Driver Tuning Performance

By selecting the GcmNetwork properties sheet their few properties required to setup.

- Max Read Values – specifies the global maximum number of Gcm block attribute to ask for in one message. The larger the number, the longer (time) it will take to response to the request. The new setting is intended to minimize the effects of the communication messages being fragment. As fragmentation occurs, the response time goes significantly.

Gcm Driver Terminal Mode

GCM integration in Niagara is intended for GCM system previously engineered using typical GCM programming tool, such as terminal mode or a program such as a Windows Hyper Terminal or XPSI. The GcmNetwork provide GCM terminal mode access as well.

Note that after integration in Niagara, GCM programming tools **remain necessary** for database saves and downloads of GCM panels and controllers. Note also that direct ASD device access by a XPSI can be done in this terminal.

Working in Terminal Mode view is like using a standard communications programming with VT-100 emulations, for example, Hyper Terminal. Depending on the GCM sign on level used, adds, deletes, and changes to blocks and other GCM system parameter are possible.

Whenever blocks are added or deleted to a GCM (Whether using Terminal Mode or another interface to the GCM), the block lost should be updated in the Gcm Point Manager view for that GCM object.

Note : While the GcmNetwork is used for Terminal Mode access all the request, polling and progress will stop, until user terminate from the Terminal mode or the Terminal Mode is IDLE between 2 minutes.

Installing The Client-Side Software

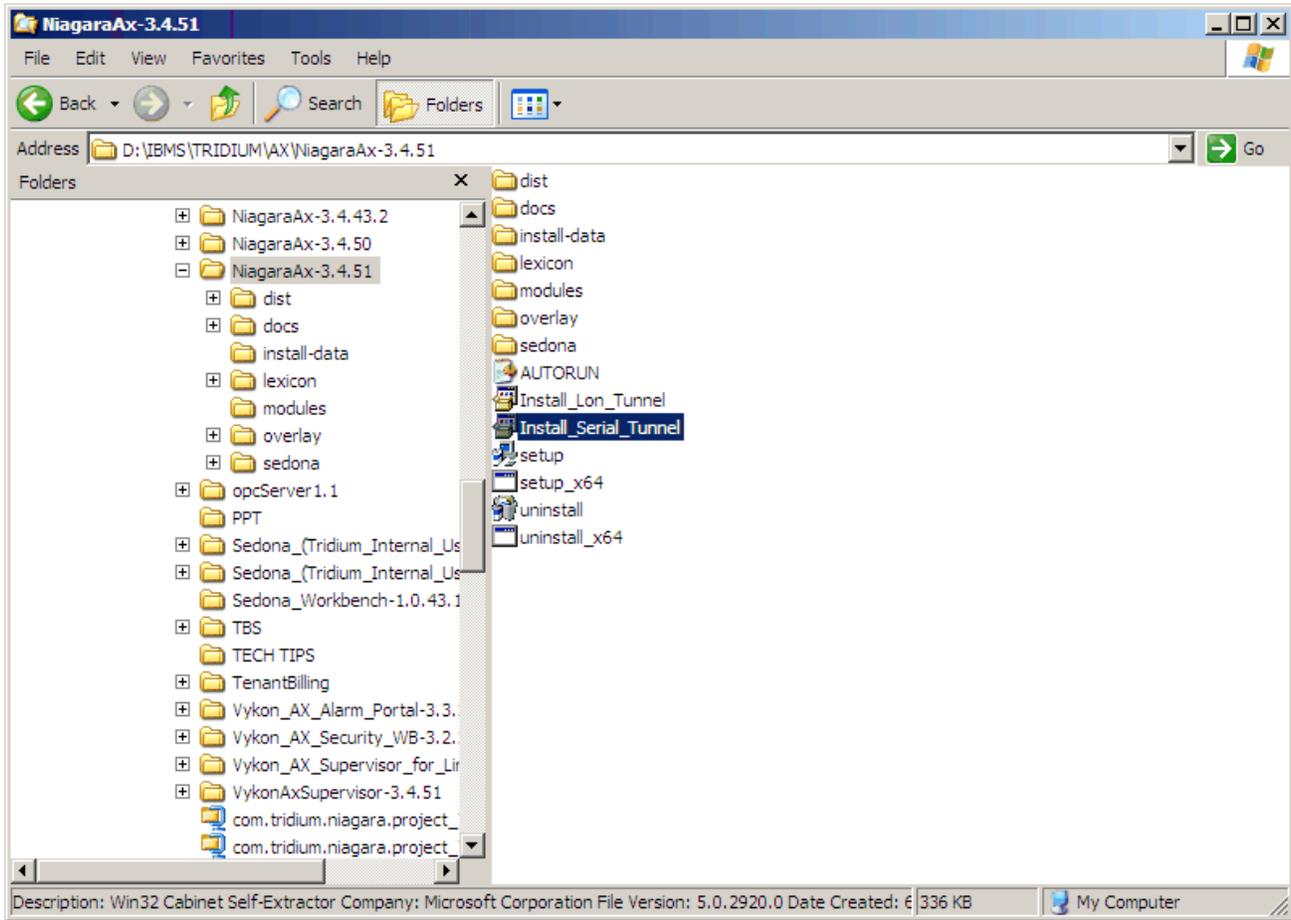
Niagara AX Serial Tunnel “client-side” software exist as a single self-installing file on the Niagara AX Web Supervisor CD. PC requirement are for Windows NT 4.0, WinXP or newest operating system that supporting the VT-100 with 32 bit OS.

Note : Windows with 64 bit OS is not supporting tunnel.

To install the client-side software:

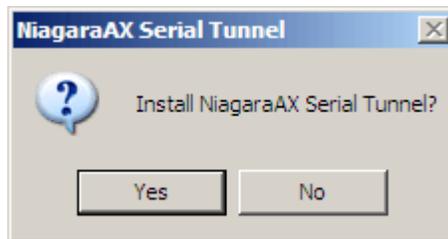
- Close all other applications on the PC.
- Insert the Niagara Ax Web Supervisor CD. If the “autorun” feature starts the Niagara Ax Framework setup, Exit from setup.

- Open Windows Explorer and locate the file on the CD: *Install_Serial_Tunnel.exe*.

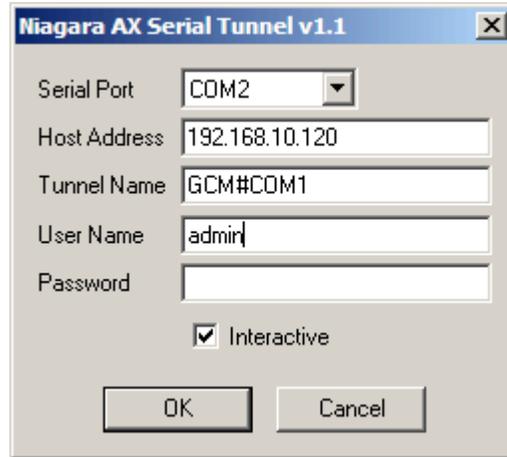
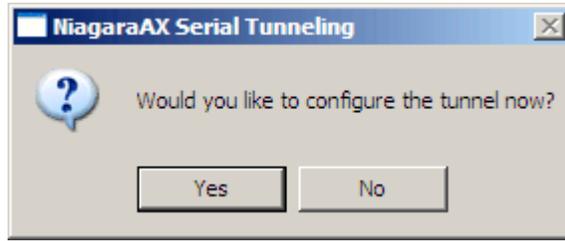


- Double-click on *Install_Serial_Tunnel.exe*.

This start some self-extraction and the Niagara Ax Serial Tunnel Setup program



- Click Yes to continue. This will pop-up the Port setup screen.



- Select a PC COM port to associate with Niagara Ax Serial Tunneling.

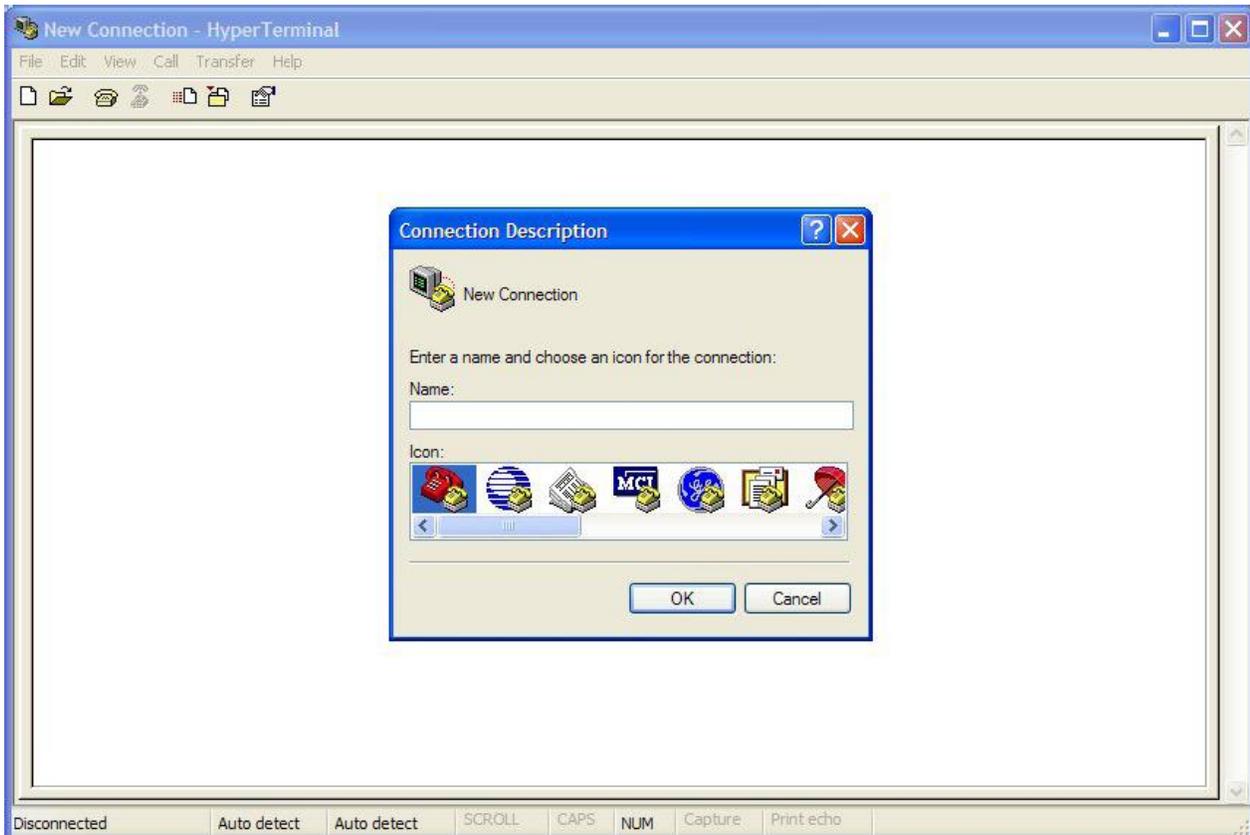
Do not select a port that physically exists in the PC.

- Host Address – The JACE IP address.
- Tunnel Name – The identification at the GcmNetwork terminal.
- User Name – Station user name.
- Passwords – Station passwords.
- Click Ok to finish setup selections.

The necessary file is installed on the PC. You will need to restart the PC to enable the client-side software.

Connecting To The GCM

With the Virtual Serial Tunneling client software installed and configured, start the XPSI or Windows Hyper terminal tool in the normal manner. (Hyper Terminal, a simple terminal program, is shown being used in this procedure).

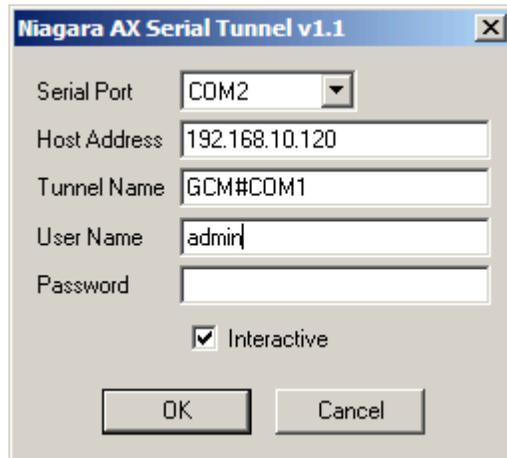


- Fill the required information at the windows hyper terminal e.g. GCM Terminal.
- Select the correct port number same as the tunnel setup, COM2.



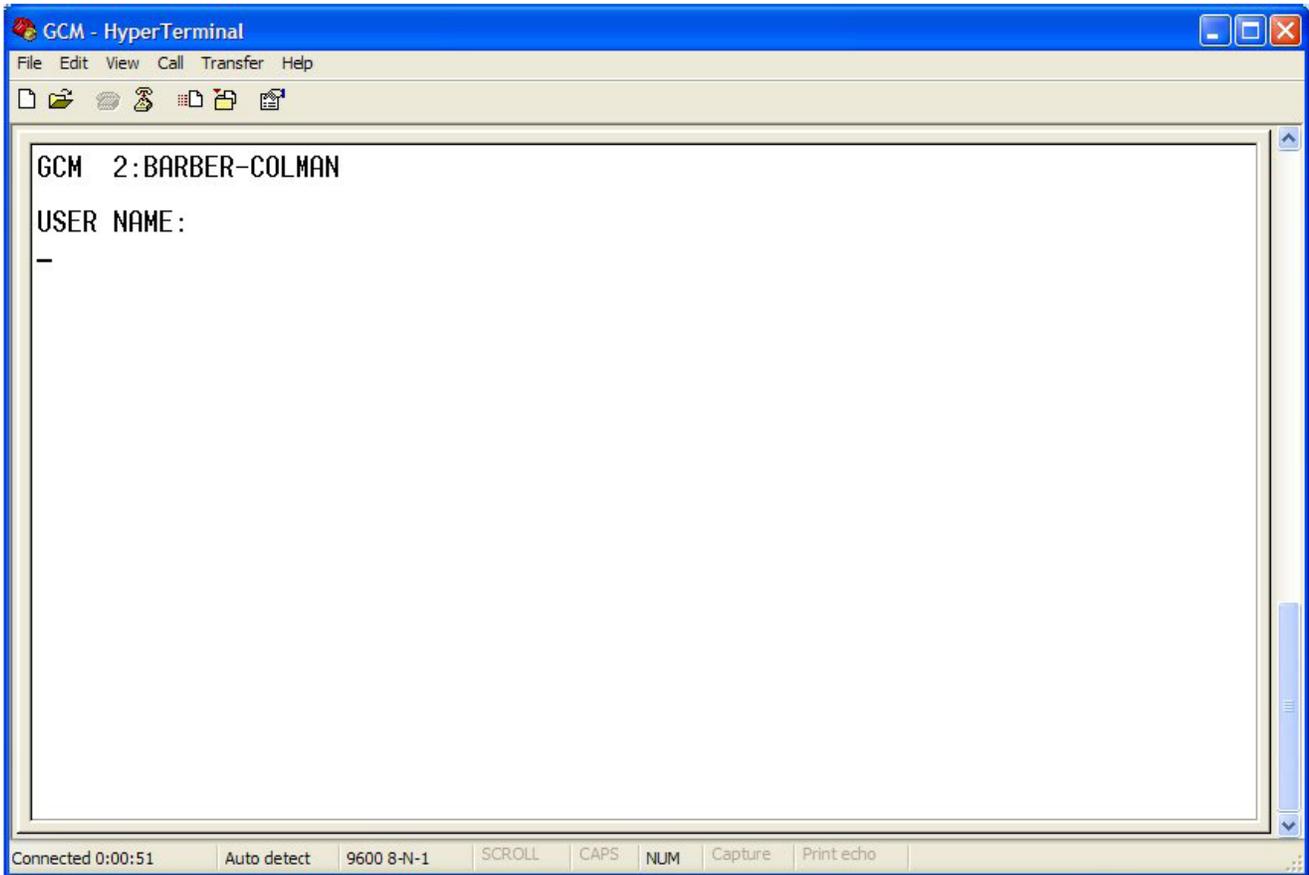
- Set the baud rate = 9600, DataBits = 8, Parity = None, Stop Bits = 1 and Flow Control = None.

- Whenever the serial tunnel start to access, the Virtual Tunnel Logon appears.



- If the JACE host address, station user name and password are correct, click OK to establish the IP connection. Otherwise, edit these items as needed.

A successful connection re-establishes the sign on screen with the virtual port shown configured.



- Enter the valid GCM user ID and password.

You are now connected to the GCM via the Niagara Virtual Serial Communication.

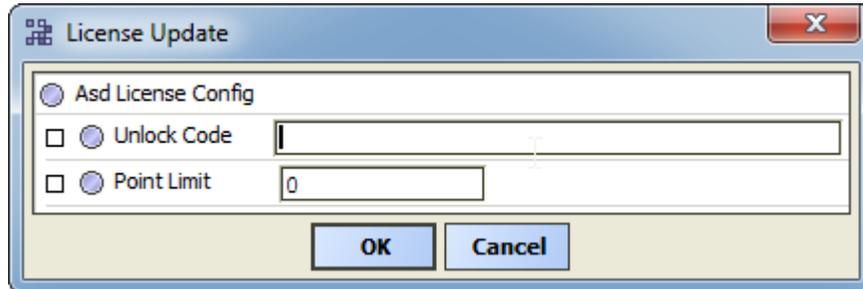


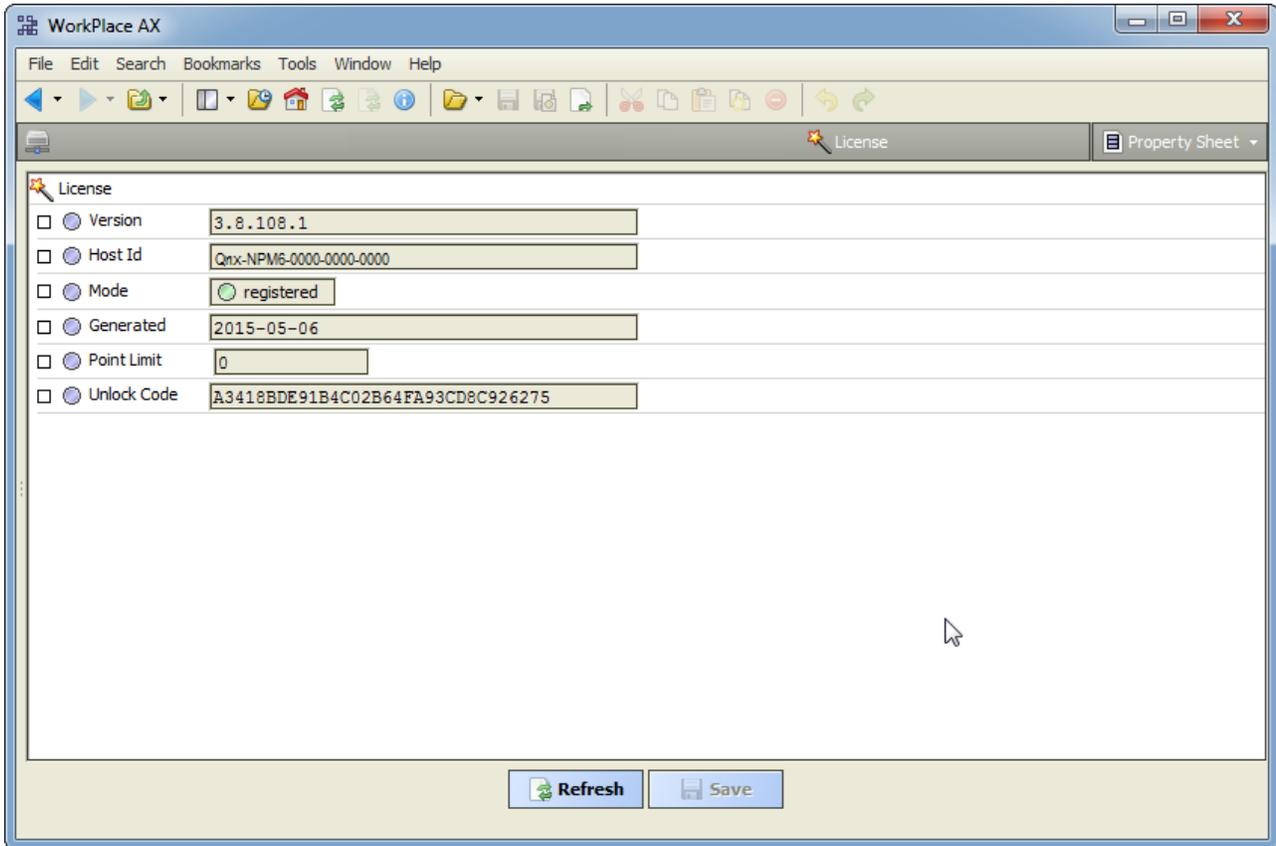
Licensing

Gcm driver License is running independently from the Tridium license, it has no restriction to run to any of the existing license vendor.

The Gcm license provide 2 hour demo license without the unlock code. After the demo license expired the Gcm Driver communication will stop automatically and will show fatal fault at the Gcm Network level. "ibmsGcm license expired". To extend the demo period user required to restart the station or reboot the JACE.

To request the license please submit the JACE/Web Supervisor host ID, to unlock the driver simply go to the license property action and invoke command "License Update", the dialog "License Update" will appear. Place the unlock code at the "Unlock Code" property. Please take note the license only provide by authorize vendor. If the unlock code is successful enter the property "Mode" will automatically change to "registered" and user is required to reboot the JACE to applied the change.





*** End of document