

Schneider Electric Xenta Driver
for Niagara AX and Niagara 4

Driver Guide

Contents

Introduction	2
Requirements.....	2
Network	2
Devices	3
Points	4
Alarms	6
Schedules	8
Xenta Schedules	9
Niagara implementation	10
Schedule Import.....	12
Schedule Export	13
Xenta Scheduler	14
Time synchronization.....	14
Xenta IP communication.....	15
Xenta IP network in Jace	15
Xenta IP network in Supervisor.....	15
Advanced topics.....	16
Use Niagara and Vista simultaneously.....	16
Determine TAC domain and group	18
Schedule configuration	19
Reading Exported Schedules.....	20
Alarm priorities	20

Introduction

Schneider Electric¹ (formerly Tour Andover Controls or TAC) Xenta range of freely programmable controllers is used for stand-alone HVAC control and for building management systems. The range includes controllers Xenta 281, 282, 283, 301, 302, 401 and their modifications. Separate IO modules Xenta 411, 421A, 451A, 471, 491 with various input and output combinations are used to increase control capacity. Xenta OP is a local display, TAC Vista is a BMS head-end and TAC Menta is a programming tool.

Although Xenta controllers utilize LonWorks communication medium, by default they communicate with a proprietary protocol on top of Lon. This protocol enables point reading and writing, input and output overriding, alarming, scheduling, trending, time synchronization, and remote programming.

Xenta driver for Niagara allows fast, simple and fully-functional integration of Xenta controllers into Niagara AX and Niagara 4 frameworks. The driver implements the following features:

- Niagara automatic configuration
- Device discovery
- Point import from Menta program
- Point discovery from devices
- Adding units of measurement
- Point polling with configurable frequency
- Point writing
- Input and output overriding
- Reading and writing and access to schedules
- Receiving Xenta alarms
- Alarm import from Menta program
- Time synchronization

Requirements

- Niagara AX 3.6² / Niagara 4.0 or later powered device, such as Tridium JACE or any OEM version like Honeywell Hawk, Distech Controls EC-BOS, Trend Controls TONN
- LON adapter card and Tridium license for lonworks driver³
- Xenta driver license

Network

Install lonworks.jar, kitLon.jar and xenta.jar files via Software Manager, start the station and add new **Xenta Network**. This network extends the standard Lon Network, therefore all Lon functions and settings are available there, in addition to Xenta-specific ones.

Open **Xenta Network** properties and enter the license code in **License** slot.

Before the communication with Xenta controllers can commence, underlying Lon network should be set-up according to Xenta system requirements.

1. Reallocate Neuron chip memory in Lon adapter, because proprietary messages are longer than standard LonWorks messages.
2. Set TAC Lon domain

¹ All trademarks or registered trademarks are property of their respective owners.

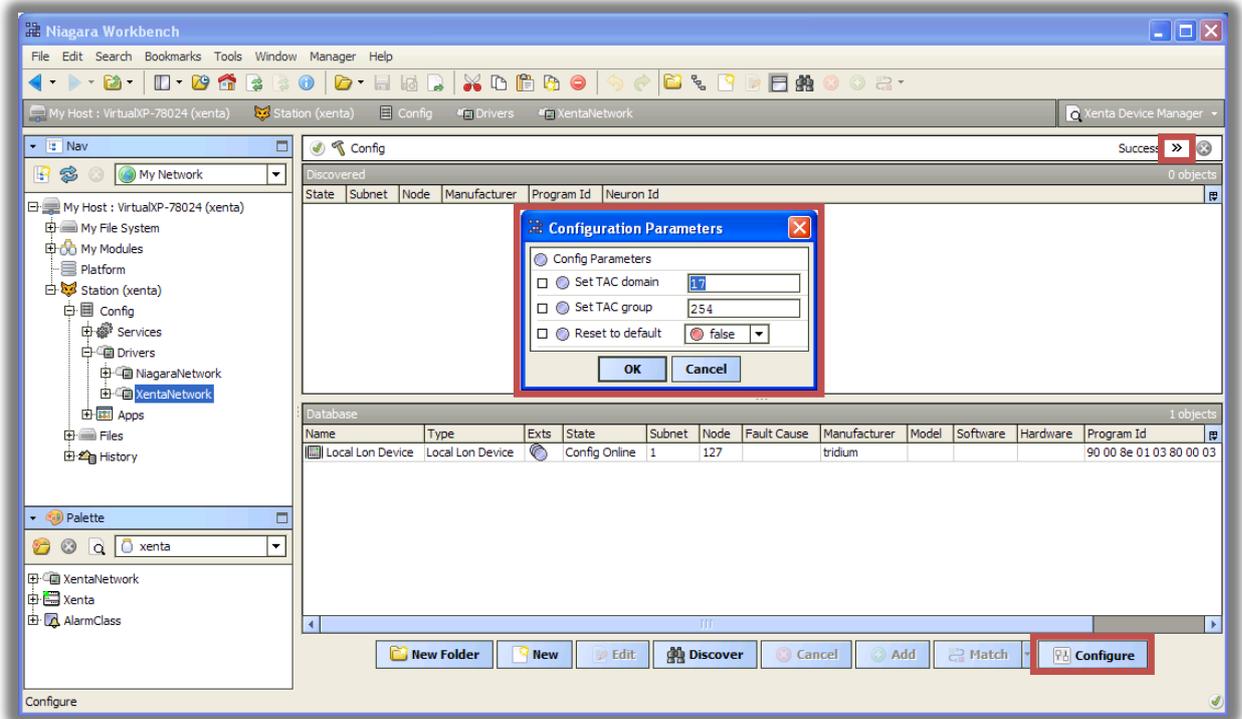
² If support for older Niagara versions is required, please contact the vendor.

³ For IP communication see corresponding section

3. Set TAC Lon group⁴

These settings are configured by opening **Xenta Device Manager** and pressing **Configure** button. The dialog box asks for domain and group values. “Reset to default” option is for reversing the settings in case you stop using Xenta driver: if set to true, all settings will be returned to default ones. After configuration is done, its status could be checked in job log by pressing [>>] symbol in top right corner.

For more information about TAC domain and group, see Determine TAC domain and group.



In addition to setting Neuron buffers, domain and group, the command sets **Max Write Time** to zero to switch off periodical write commands and creates two more tuning policies.

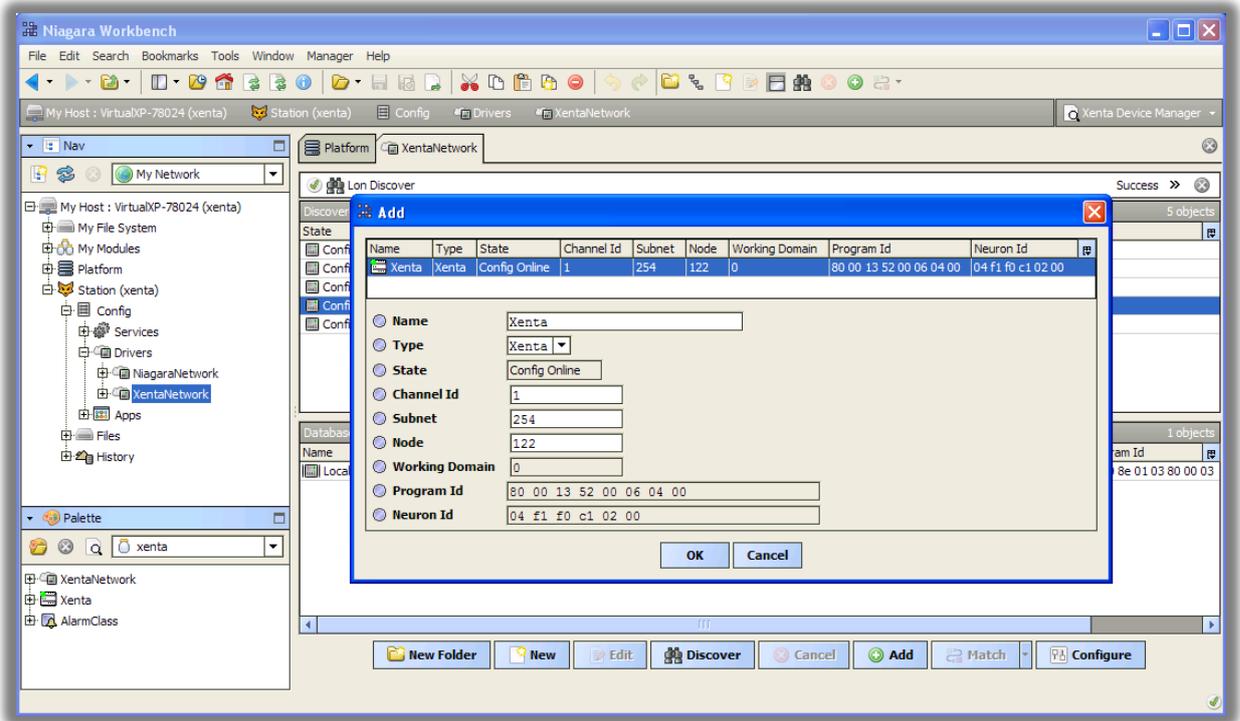
Devices

If Niagara is connected to existing Xenta installation, pressing **Discover** button in **Xenta Device Manager** will find all Lon devices in TAC domain: Xenta controllers and IO modules, displays, 3rd party Lon devices.

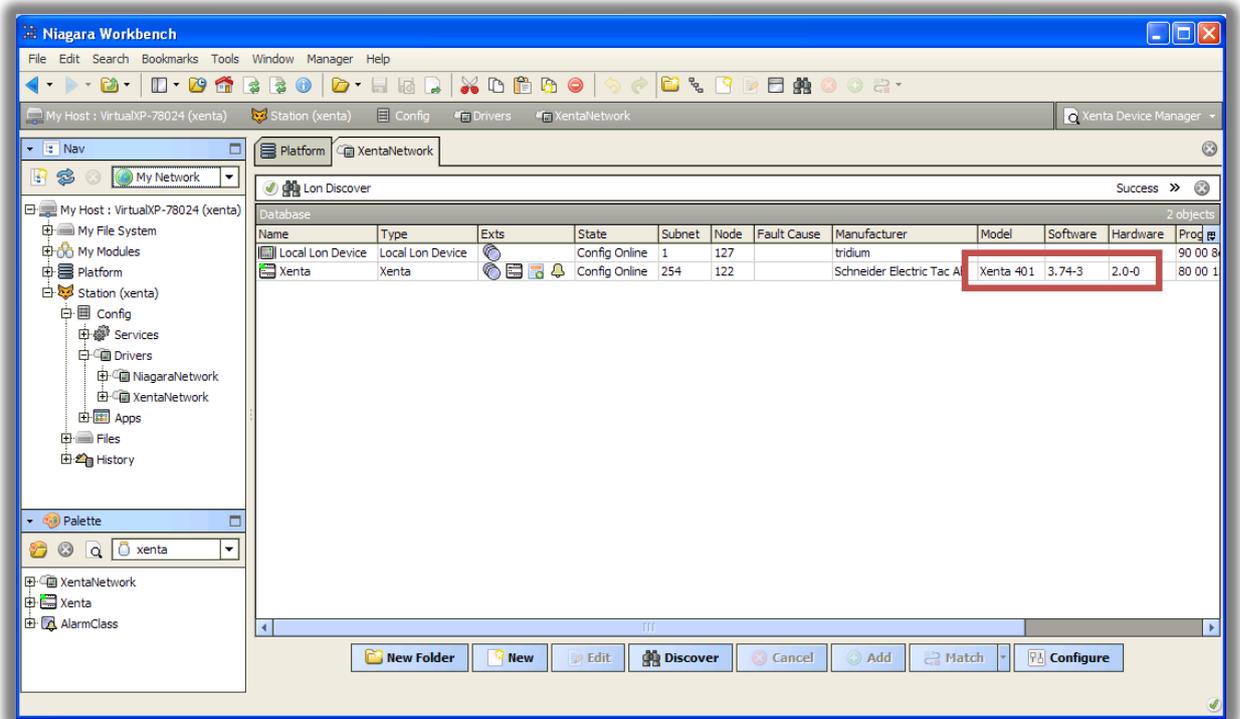
- Xenta controllers are shown with Manufacturer Schneider Electric and when added will have Xenta type.
- Displays and IO modules should not be added to Niagara database – they communicate directly with Xenta controllers, so all information from IO modules will be available via controllers.
- 3rd party devices could be added as generic LonWorks devices if necessary, although in existing systems they most likely are already bound to Xenta controllers, so their data could be obtained via controllers as well.

⁴ If the group is not set, the following message is displayed in Application Director during start-up: “TAC group is not set in Neuron address table. Use Configure button in Xenta Manager”

Do not change Channel Id, Subnet and Node in Add dialog box – this will change addresses in Xenta device, which may break communication between Xenta controllers and modules. Also do not try to Commission any device in existing installation, unless you know what you are doing.



After Xenta controller is added, the driver will read its model, software and hardware versions.



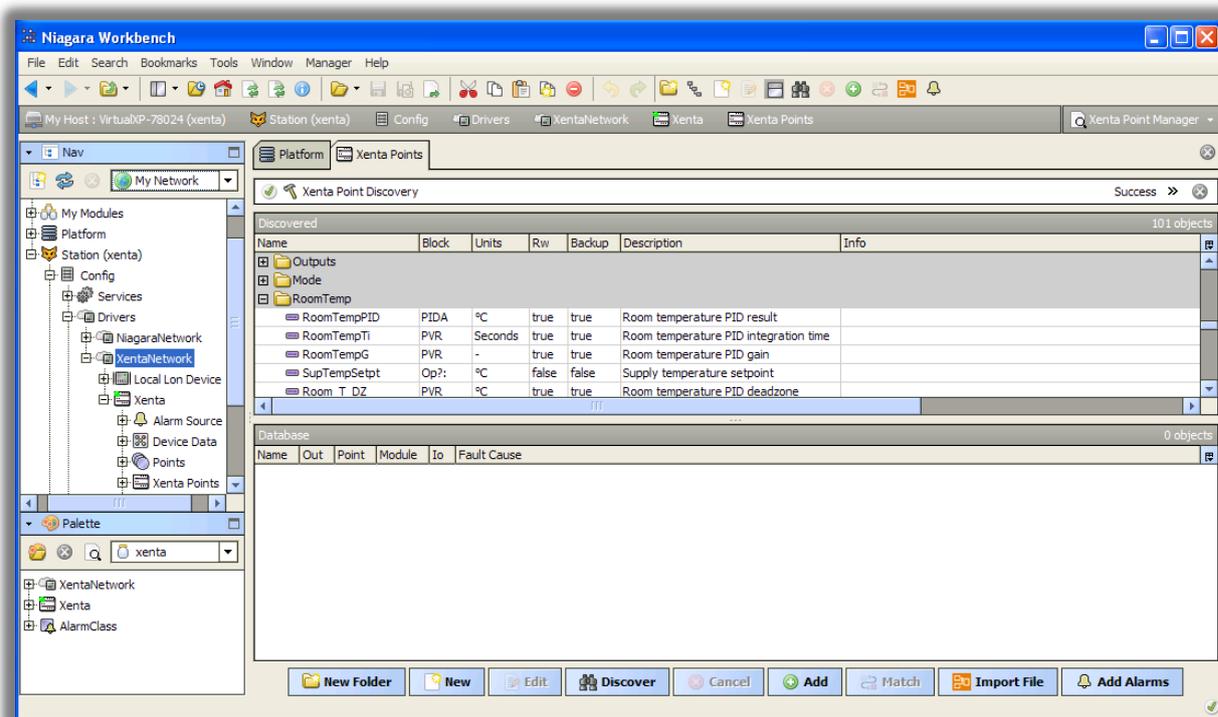
Points

Xenta controllers have two point extensions: standard Lon points and Xenta points.

Xenta controllers could be programmed as LonWorks devices, so they might contain generic Lon points, which could be read and written using standard Lon mechanisms. As the number of these points are limited, an extra programming is required, and not every Xenta point could be represented as LonWorks point, it is advantageous to use Xenta points.

Xenta points extension allows to see all points, which are available for native TAC Vista software. There are few ways how to add points to Niagara database:

- **Import** function allows getting all points from Menta program file (MTA or AUT file extensions). These files should be available in existing installations in Vista head-end. This is the best way to add points, because it provides all possible information: point name, type, program block name, units, read-write status, backup, description. If point is added, Niagara will automatically provide appropriate point types and unit facets.
- **Discover** function could be utilized, if no Menta file for the controller is available. This function will read points from the controller and use heuristic algorithms to determine its type. The process may take a few minutes, depending on program size. The progress bar on the top and job log button [>>] provide information about discovery. As this feature is not available in Vista, and controllers are not designed for point discovery, there are few limitations in this process: no point units and descriptions could be discovered, just point names; sometimes a point could be reported writable even if it is not; very old Xenta firmware may not work – a warning will be shown in this case. Despite these limitations, point discovery is an invaluable tool when the program file is lost or inaccessible.
- **New** point manual creation: point module, name and IO status should be provided.

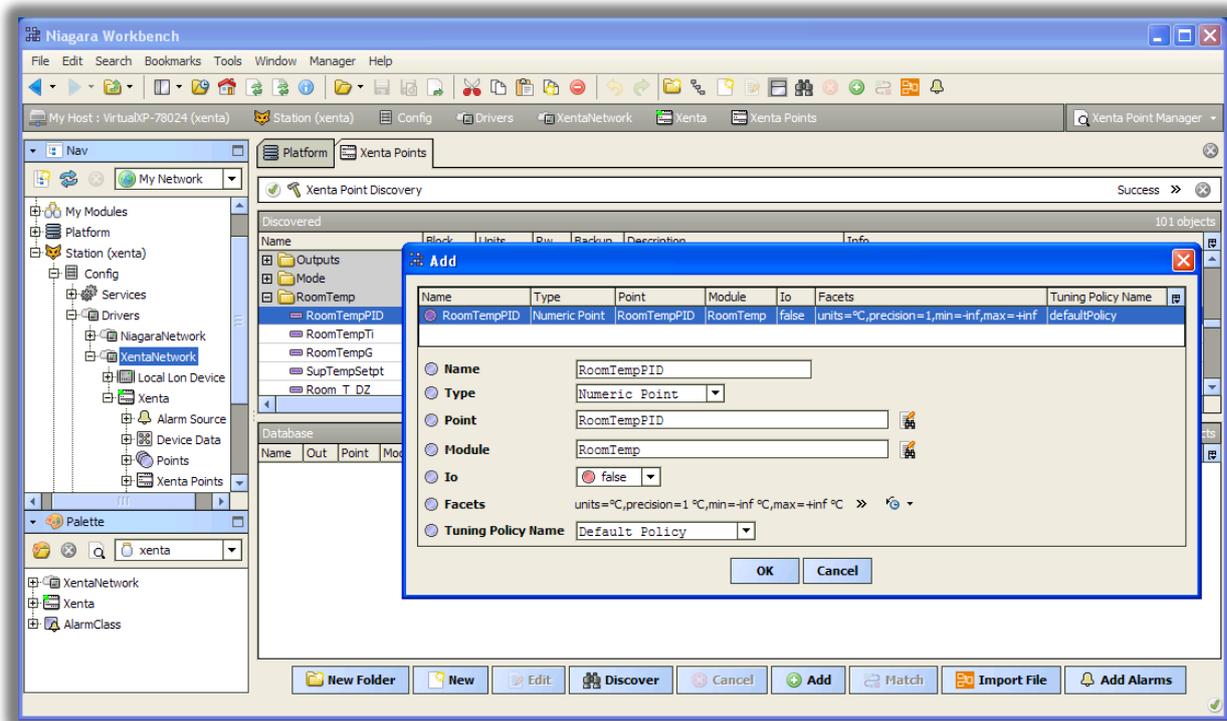


Xenta points are grouped in modules, each module could be expanded and points could be added to Niagara database. The values will be read from controller instantly. By default, all points are read-only. Select writable points to make them read-write. Inputs and outputs could not be overridden.

A standard Niagara polling mechanism is used to control update rate for each point. See **Lon Network >> Xenta Poll Service** to specify fast, normal and slow update rates. By default these are 5, 10, and 30 seconds.

Each point has a **Tuning Policy**, defined in **Lon Network >> Tuning Policies** to specify update rate and writing behaviour. There are 3 policies provided: default (normal), slow and fast. It is possible to create more policies to customize read and write behaviour.

Max Write Time property in Tuning Policy defines how often point value is rewritten. By default this value is set to zero, i.e. to write only once when the action is requested. It might be beneficial in some cases to rewrite some of the points periodically, i.e. to prohibit any change. This will cause extra traffic, so use with caution.



Alarms

Xenta controllers are capable of generating alarms – using ALARM block in Menta program – which will be sent to Vista front-end and could be checked from Xenta OP display. The driver will enable Niagara to receive these alarms, translate them into regular Niagara alarms and send user acknowledgements back to Xenta controller.

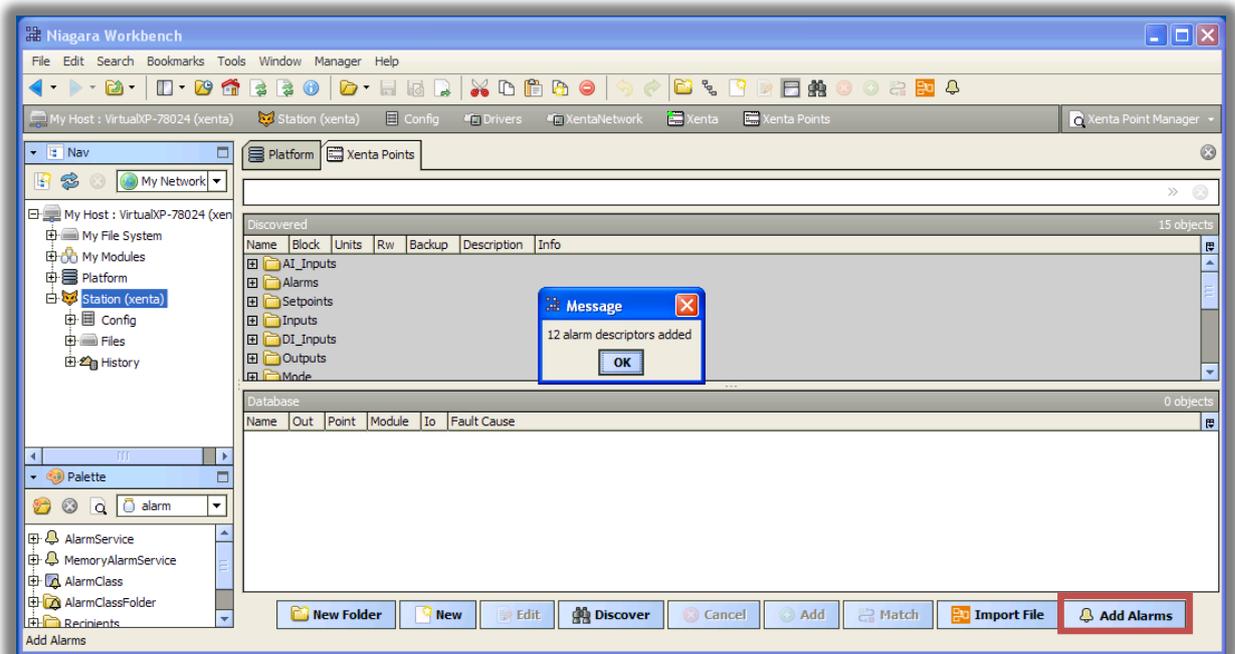
Xenta alarms have the following attributes, which are assigned during Xenta programming:

- Module and point name of the ALARM block – the source of the alarm
- Priority – number from 0 to 9, where 1 is the highest, 9 is the lowest, and 0 is extra-low
- OP alarm text – alarm message displayed on Xenta OP
- Tripped alarm text – alarm message displayed in Vista, when alarm is generated
- Reset alarm text – alarm message displayed in Vista, when alarm returns to normal

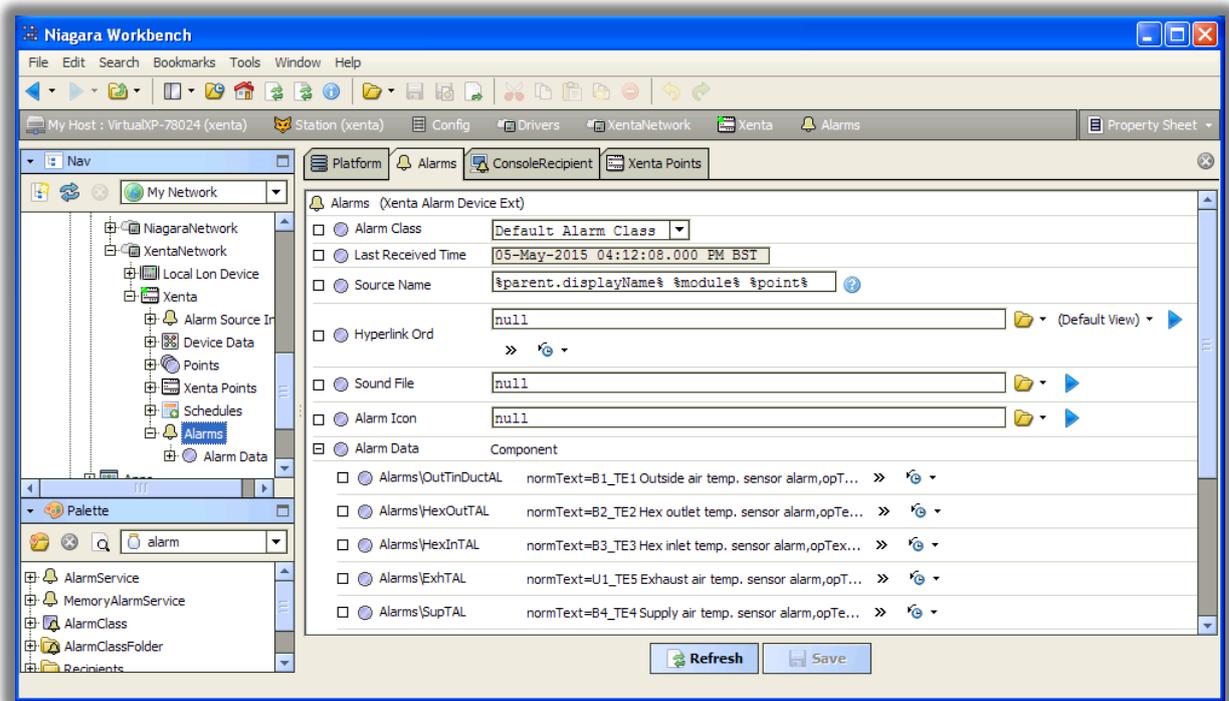
Tripped and reset alarm texts are optional; they are stored in Vista, not in a controller. If these texts are not available, then OP alarm text is displayed for both alarm event and return to normal event.

Niagara could read all attributes directly from controller, except tripped and reset texts. If Menta file is available, then tripped and reset texts could be imported into the station as well. Pressing **Add Alarms** button in **Xenta Point Manager** after Menta file is imported will add all found alarm descriptors as facets in Alarm Data property under Alarms extension. These descriptors will be used in case real alarm events are generated by Xenta. It is possible to manually edit these facets to modify texts and priorities.

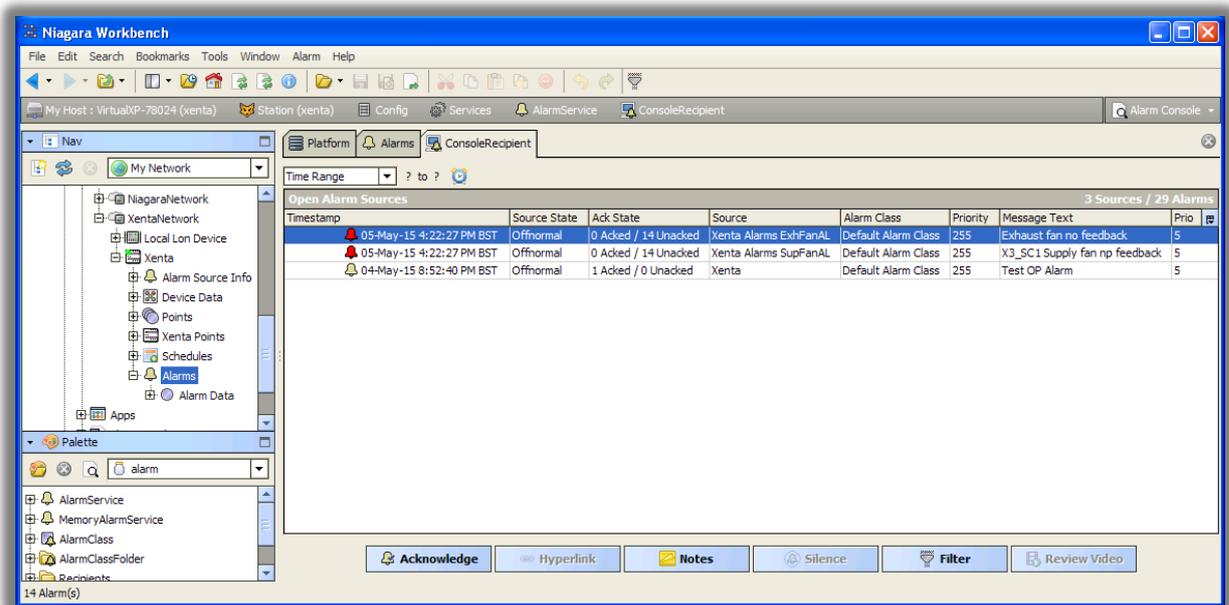
In case Menta file is not available, OP alarm text will be read from controller and displayed as alarm message.



Niagara starts to receive Xenta alarms right after TAC domain and group are configured and heartbeat is enabled; no extra operations are required – see [Use Niagara and Vista simultaneously](#). All alarms are redirected to Niagara alarm class, which is set in Alarms extension. Properties Source Name, Hyperlink Ord, Sound File, Alarm Icon are standard Niagara properties used to assign extra information to alarm records. Special keywords **%module%** and **%point%** will be replaced with real module and point name of the alarm.



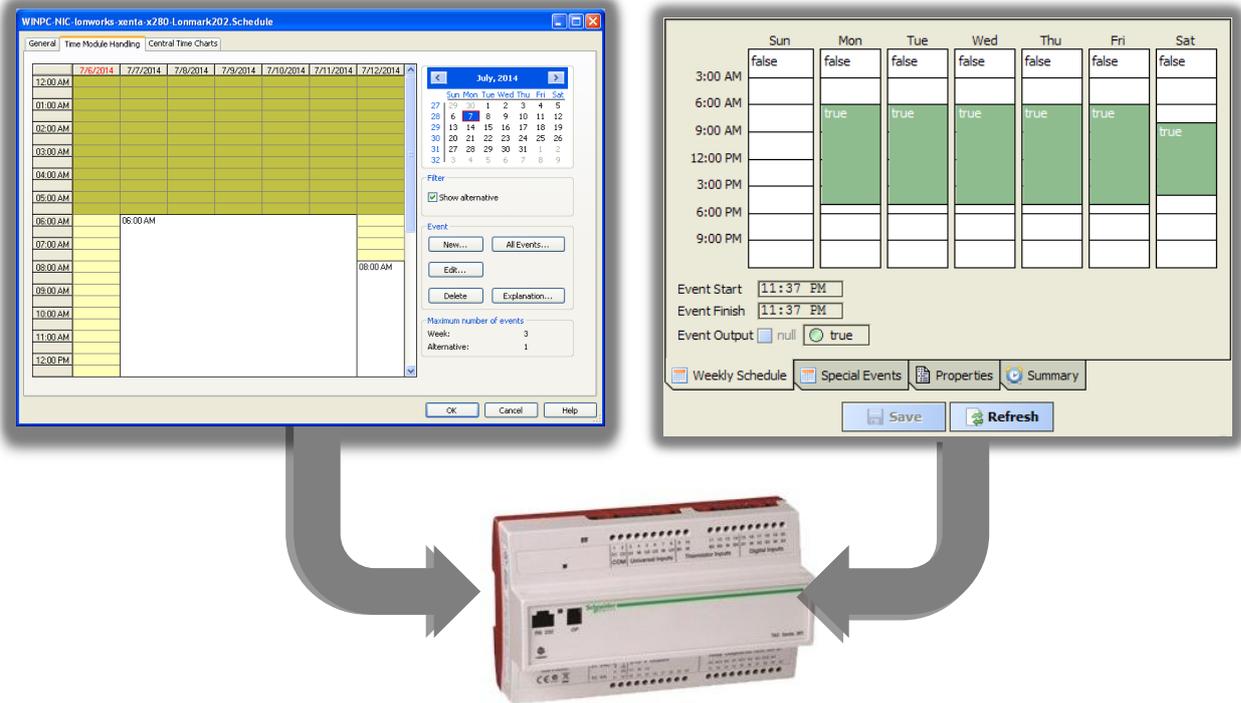
After alarms are redirected to alarm class, they could be displayed with **Console Recipient**, send to Supervisor with **Station Recipient** and printed with **Printer Recipient** as regular Niagara alarms. If user acknowledges any alarm, an acknowledgment is sent to Xenta, so Xenta OP will show this alarm as acknowledged.



For more information about alarming see [Use Niagara and Vista simultaneously](#) and [Alarm priorities](#).

Schedules

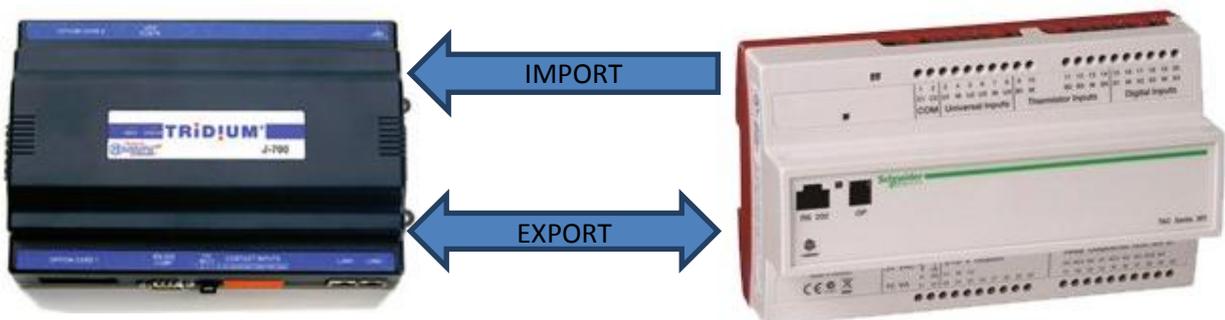
Xenta controllers can store and execute complex time schedules. Users normally modify these schedules with Vista software and with portable display Xenta OP.



The driver provides full control over Xenta schedules from Niagara. Schedules can be imported, modified and exported into controllers. In order to integrate Xenta schedules with Niagara, Xenta programs should not be modified. Xenta schedules could be simultaneously accessed from Niagara, Vista and Xenta OP.

Xenta schedules are modelled in Niagara similarly to BACnet schedules:

- Imported schedules are read-only. They are useful if a schedule should not be modified from Niagara. Imported schedules are generic Niagara Boolean Schedules and could be used for time-control inside of Niagara station.
- Exported schedules are the most useful, because they can be read (imported) from controllers, modified in Niagara UI and then written back (exported) into controller.

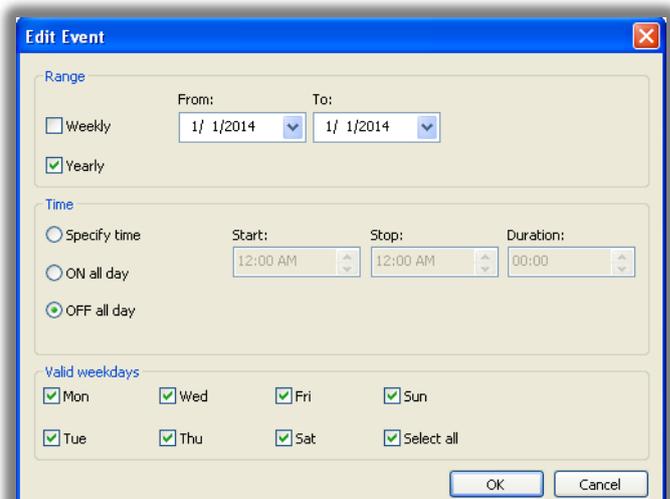
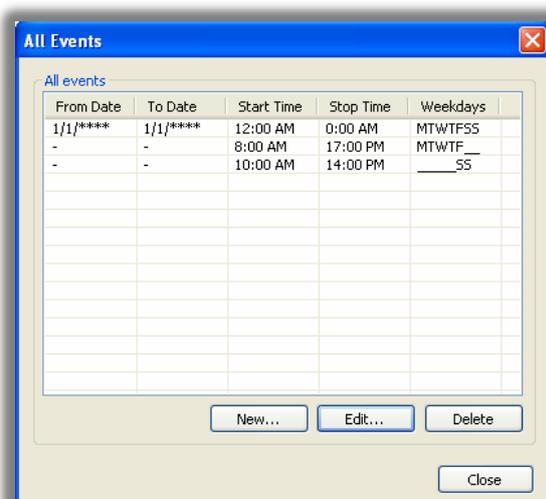
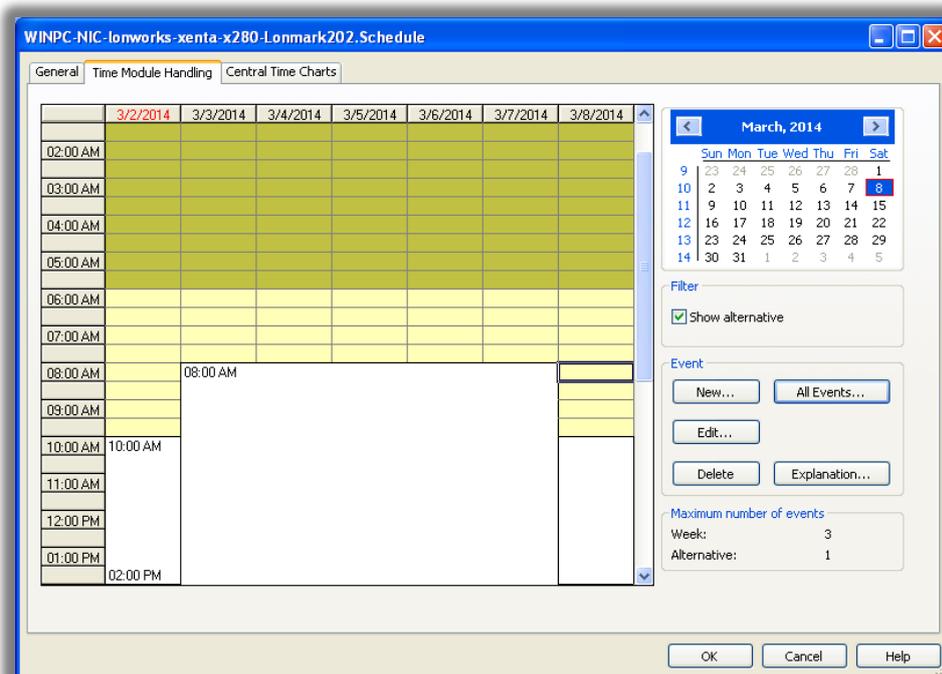


Xenta Schedules

Each programmable Xenta controller can contain one or more schedules. Each schedule consists of a limited number of weekly events and special (alternative) events. Maximum amount of events is defined during programming and cannot be exceeded. Schedule can be in one of two states: active (true) and inactive (false).

Weekly event is a time period repeated on one or more weekdays every week, e.g. Mo-Tu-We-Th-Fr 8:00-17:00 – during this time schedule output will be active. Special event is a date or a date range,

which could be repeated every year, with a time period for each day (e.g. holidays). Special events override weekly periods.



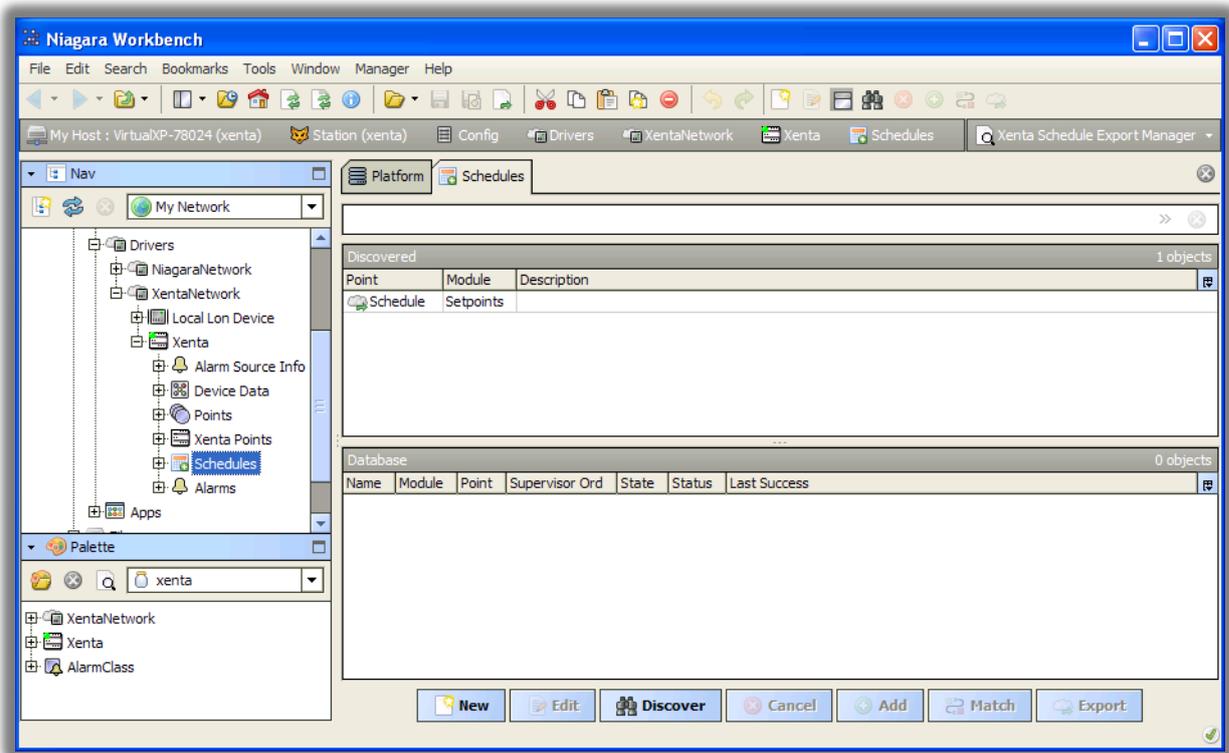
An example of Xenta schedule. Maximum number of events: 3 weekly and 1 alternative (special). There are 2 weekly periods: 8:00-17:00 on working days and 10:00-14:00 on weekend. There is one special event: on 1st of January every year the schedule is inactive.

Niagara implementation

Xenta schedule can be well mapped into Niagara schedules: they also contain two types of events (weekly and special) with special events having higher priority; special events can be a date or date range repeated yearly, etc. There are few limitations which prevents every Xenta schedule to be mapped into Niagara schedule and vice versa – it will be covered later, although these limitations are practically not important.

There are two ways how to add Xenta schedules to Niagara database, depending on availability of Menta program files.

- If previously Xenta points are imported from Menta file, then discovered points with block **TSCH** represent found time schedules⁵. Xenta Schedules extension will display the schedules in Discovered pane, so they could be automatically added to the database.
- If Menta files are unavailable, schedules could be created manually with **New** button by defining modules and point name. Although point discovery does not find schedules automatically, it provides enough information to determine necessary information: look for enum points with appropriate name, e.g. “AHU_Schedule”, note module and point names and try to create **New** schedule import with the same module/point names. The driver will try to read the schedule and you will be able to open it and see time periods.



Schedules extension has two main views: **Xenta Schedule Export Manager** and **Xenta Schedule Import Manager**. To determine which one is needed, select one of use cases for the schedule:

1. Read-only schedule. In this case Xenta schedule will not be modified from Niagara (although it still can be modified from Vista and Xenta OP). The schedule will be imported (periodically or by demand) into Niagara as *read-only* Boolean Schedule to use as a regular Niagara schedule: time control program blocks and outputs in Niagara and external equipment connected via any protocol⁶. This is a rare use case – proceed to **Schedule Import**.
2. Write-only schedule. In this case schedules could be modified exclusively from Niagara UI and the exported (periodically or by demand) into Xenta controller. This is the most common scenario – proceed to **Schedule Export**.
3. Read-write schedule. If the schedule could be modified from multiple sources, i.e. Niagara UI and Xenta OP, when it is necessary to do both import to show existing Xenta schedule to

⁵ Adding TSCH points to station from **Xenta Point Manager** will produce just a **Numeric Point** equal to the number of minutes left until the next state change: positive for OFF-ON change, negative for ON-OFF change.

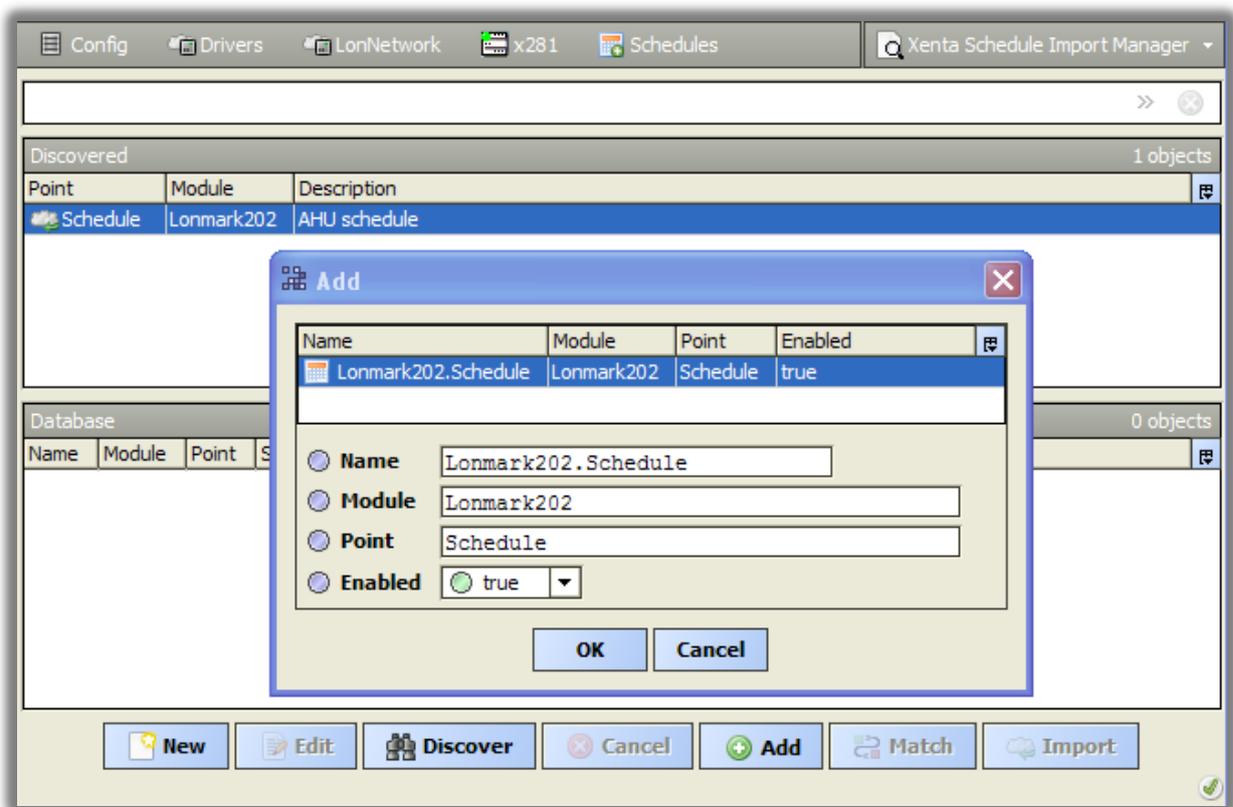
⁶ Xenta schedule import might be useful when Niagara is used not as a front-end, but as a protocol gateway.

Niagara user and export to send modified schedule back to controller. This is the most advanced option – proceed to **Schedule Export** and then **Reading Exported Schedules**.

*When doing periodical import or export you should not set **Execution Time** interval too short to avoid overloading communication network. Schedule objects might be “heavy” compared to regular data points and they do require longer transmission time. Usually there is no need to import schedules more often than once per hour or even daily.*

Schedule Import

When adding Xenta schedule into database in **Xenta Schedule Import Manager**, a **Xenta Schedule** with **Xenta Schedule Import** extension will be created and automatically read from the device. Now the schedule could be opened, viewed (but not modified), embedded into PX graphic and linked to other points in a usual way.



The schedule is imported from Xenta:

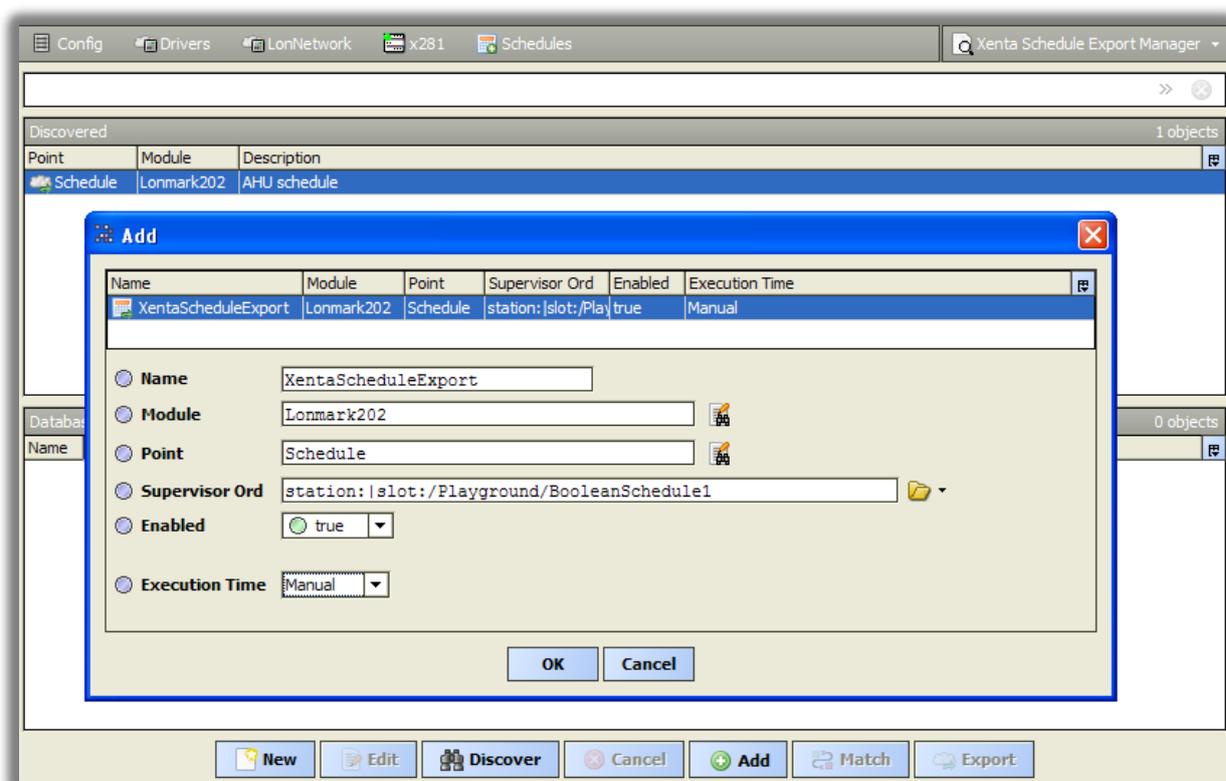
- Periodically
 - The trigger **Execution Time** in **ext** property is set to Interval or Daily.
- By demand
 - The button **Import** in **Xenta Schedule Import Manager** is pressed.
 - The action **Read From Device** on **ext** property in the schedule is launched.

Schedule Export

In order to establish Xenta schedule export, first a **Xenta Schedule**⁷ should be created anywhere in Niagara station (drag & drop from Xenta palette). Xenta Schedule is almost identical to Niagara Boolean Schedule. This schedule will act as a master (supervisor) schedule for the one stored in Xenta controller. After mapping to Xenta is complete, this Xenta Schedule will be exported into Xenta.

When adding **Xenta Export** component into database in **Xenta Schedule Export Manager**, select the master schedule **Ord** in **Supervisor Ord** field. The master schedule will be linked to **Xenta Schedule Export** object, which enables schedule export every time the schedule is changed and **Save** button in the **Scheduler** is pressed.

Then the schedule will be automatically imported from the controller into master schedule. Now the schedule can be opened, viewed, modified, embedded into PX graphic, linked to other points and exported into controller subordinate schedule.



The schedule is exported into Xenta:

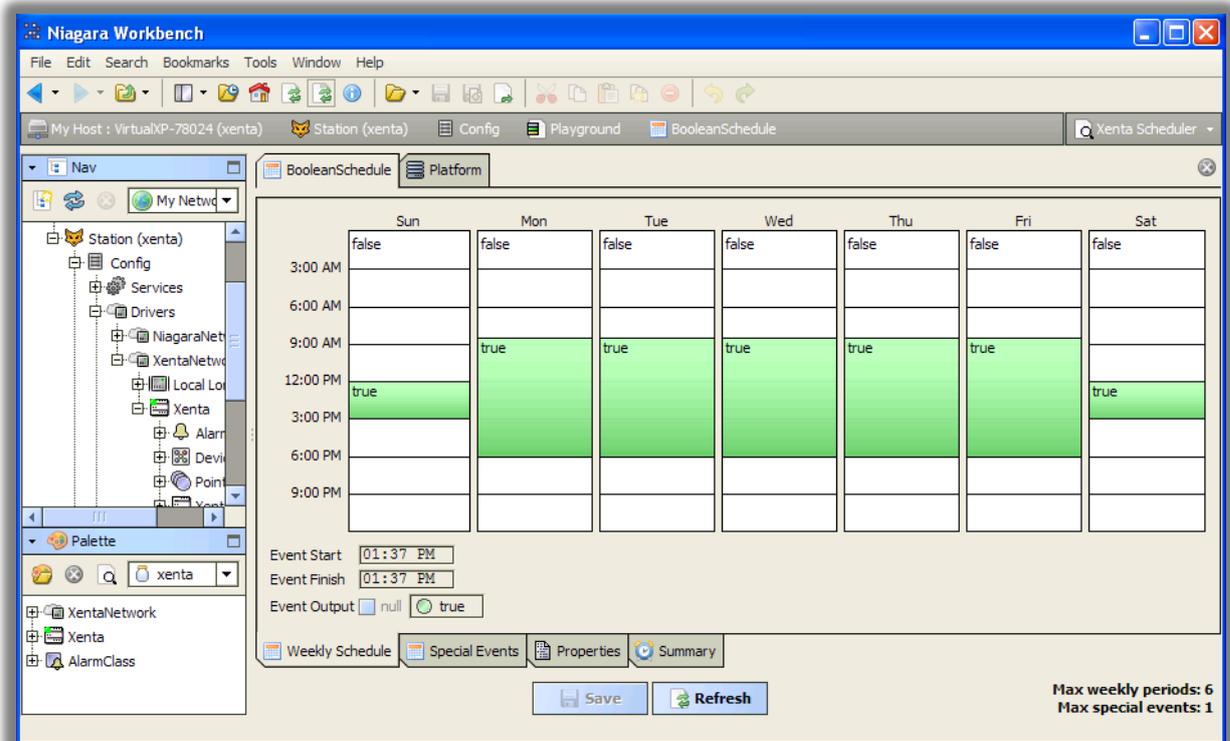
- Periodically
 - The trigger **Execution Time** is set to Interval or Daily.
- By demand
 - The schedule is saved in **Xenta Scheduler**.
 - The button **Export** in **Xenta Schedule Export Manager** is pressed.
 - The action **Write To Device** on **Xenta Schedule Export** is launched.

⁷ Note, in previous driver versions a regular Boolean Schedule has been used instead of Xenta Schedule. In case of driver upgrade, these Boolean Schedules should be replaced with Xenta Schedules.

Xenta Scheduler

After adding a new schedule for import or export, Niagara will import the schedule from the controller to determine the maximum amount of weekly periods and special events allowed for this schedule. These values will be stored as schedule facets called **week** and **special**.

In order to prevent users exceed these limits a new widget called **Xenta Scheduler** could be used. This widget is almost identical in appearance and behaviour to the original Niagara Scheduler.



Maximum weekly periods and special events are displayed on the screen for information.

If the number of periods and events exceed these limits, pressing on Save button will not save the schedule, but instead will show a warning message.

Because of Xenta schedule limitations, the only allowed types of Niagara special events are **Date** and **Date Range**. If other types of events are selected, a warning message will be displayed.

Time synchronization

Xenta controllers need to know the accurate time to correctly execute time schedules and record timestamps on alarms. Normally Vista periodically broadcasts the current time, so all Xenta could set the internal time clock.

The driver could broadcast the time of Niagara-powered device with period set in **Xenta Network >> Xenta Poll Service >> Timesync Interval**, 10 minutes by default. If the time period is set to zero, time sync broadcasts are disabled. The broadcast could be initiated manually using action **TimeSync** in **Xenta Poll Service** component.-

Make sure there is just one time broadcaster per network; see [Use Niagara and Vista simultaneously](#).

To check the local time of Xenta controller use its action **Read Time**. Property **Time Date** displays the received time and property **Time Diff** displays the difference between received Xenta time and Niagara time. It is normal to have few seconds' difference.

Xenta IP communication

It is quite common practice to interconnect Xenta LON TP/FT-10 networks via IP. There are few ways how to accomplish it.

- Use proprietary TAC hardware like Xenta 911 or Xenta 511. These controllers implement their own proprietary IP protocol, which is currently not supported by Xenta driver.
- Use standard LonWorks IP (EIA-852 or CEA-852) routers, like Loytec L-IP. These devices are supported by Xenta driver. It relies on either Tridium or Loytec EIA-852 software for communication.

Some features of Xenta protocol – reading and writing of points and schedules – use unicast or one-to-one communication. Other features – time synchronization and alarming – use broadcast or one-to-many communication.

Xenta IP network in Jace

To connect Jace to Xenta controllers via IP-based EIA-852 network, one should add **Xenta Ip Network** to the station. This network extends Tridium **Lon Ip Network**, thus the Jace should be licensed for Tridium LonIp driver.

Tridium LonIp driver does not implement Lon broadcast messaging, so Xenta time synchronization and Xenta alarms will not work in this scenario.

Configuration of **Xenta Ip Network** is identical to the one of **Lonworks Ip Network** and it is described in details in **Niagara Lonworks Guide**, section **Lon over IP**.

Assuming Loytec L-IP operates as Config Server one should:

- Open L-IP web console, **Config / CEA-852 Server**.
 - Make sure **Config server status** is enabled.
 - Set **Auto members** to on.
 - Open **CEA-852 Ch. List** – there should be at least one member (local).
- In Niagara station **Xenta Ip Network / Ip Channel / Config Server Ip** enter IP address of L-IP. **Is Config Server** property should be false.
- L-IP **CEA-852 Ch. List** should now display Jace IP. In Jace **Member Table** under the network's **Ip Channel** will become dynamically populated.
- From this point forward, other Lonworks functions will perform as if directly connected to a Lon trunk, that is, as if using the standard **Xenta Network**.

Xenta IP network in Supervisor

To connect Supervisor directly to Xenta controllers via IP-based EIA-852 network, it is recommended to use Loytec NIC852. It is a USB dongle and software installed on Supervisor PC, which enables Loytec EIA-852 software stack instead of Tridium one.

Loytec NIC852 software fully implements EIA-852 standard, so all Xenta driver features will work.

Assuming Loytec L-IP operates as Config Server one should:

- Open L-IP web console, **Config => CEA-852 Server**.
 - Make sure **Config server status** is enabled.
 - Set **Auto members** to on.
 - Open **CEA-852 Ch. List** – there should be at least one member (local).
- Install and activate Loytec NIC software on Supervisor PC, insert Loytec NIC852.
 - In **LConfig** utility under **NIC852** tab open **CNIP Configuration** and enter L-IP address in **Conf. Server IP Addr**.
 - Assign **Device Name**, e.g. “Niagara”.
 - Press **Test Device**. Serial number should be displayed.
- L-IP **CEA-852 Ch. List** should now display Supervisor IP.
- Add **Xenta Network** (not **Xenta Ip Network!**) to Supervisor station.
- To determine how to connect **Xenta Network** to NIC852 start Windows Registry Editor (regedit.exe). Open **HKEY_LOCAL_MACHINE / SOFTWARE / Lonworks / DeviceDrivers / NIC_852_1_001_LDV** (this entry might be different) and look for **device name** value, which should be similar to **\\.\LegacyMIP9**.
- Enter this name – **\\.\LegacyMIP9** – in station **Xenta Network / Lon Comm Config / Device Name**.
- From this point forward, other Lonworks functions will perform as if directly connected to a Lon trunk, that is, as if using the standard **Xenta Network**.

Advanced topics

Use Niagara and Vista simultaneously

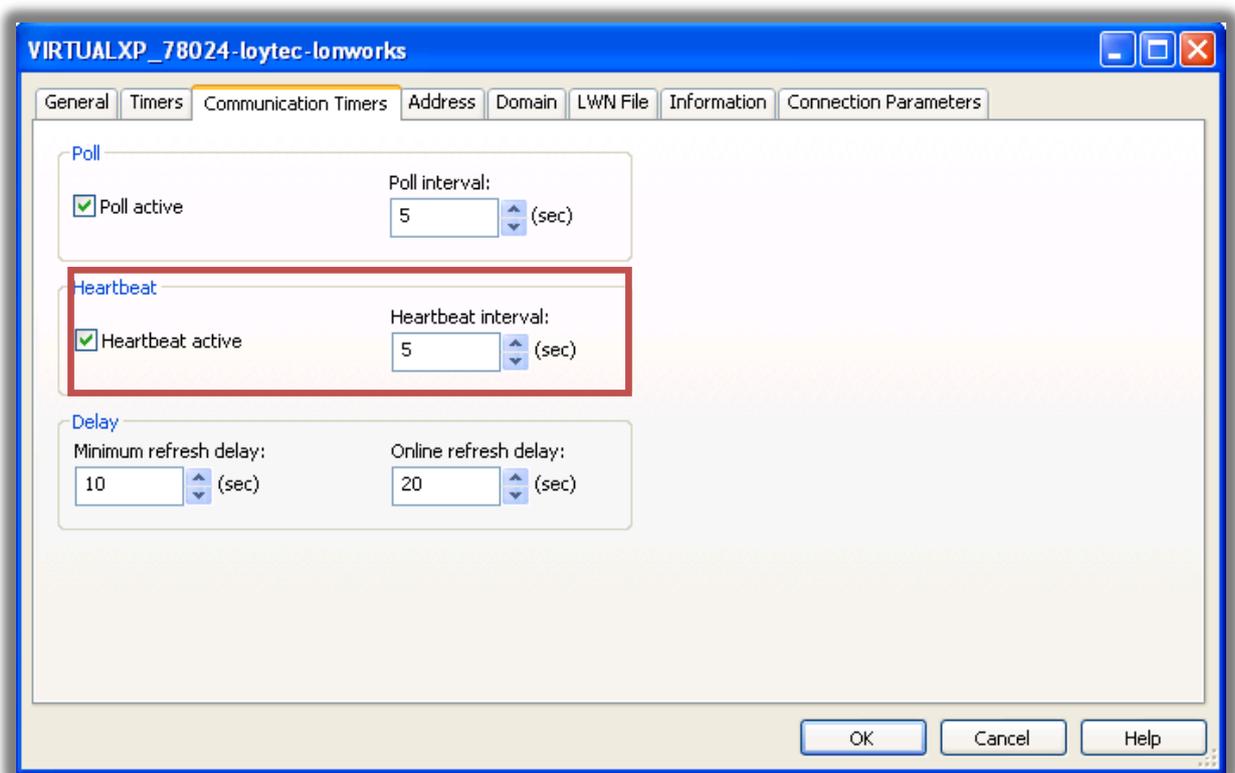
When refurbishing an existing BMS with Vista head-end, it is important to make the transition comfortable for users, so they would not lose control over their system in any time period. Niagara and Vista can successfully coexist in the network, if few configuration rules are applied properly.

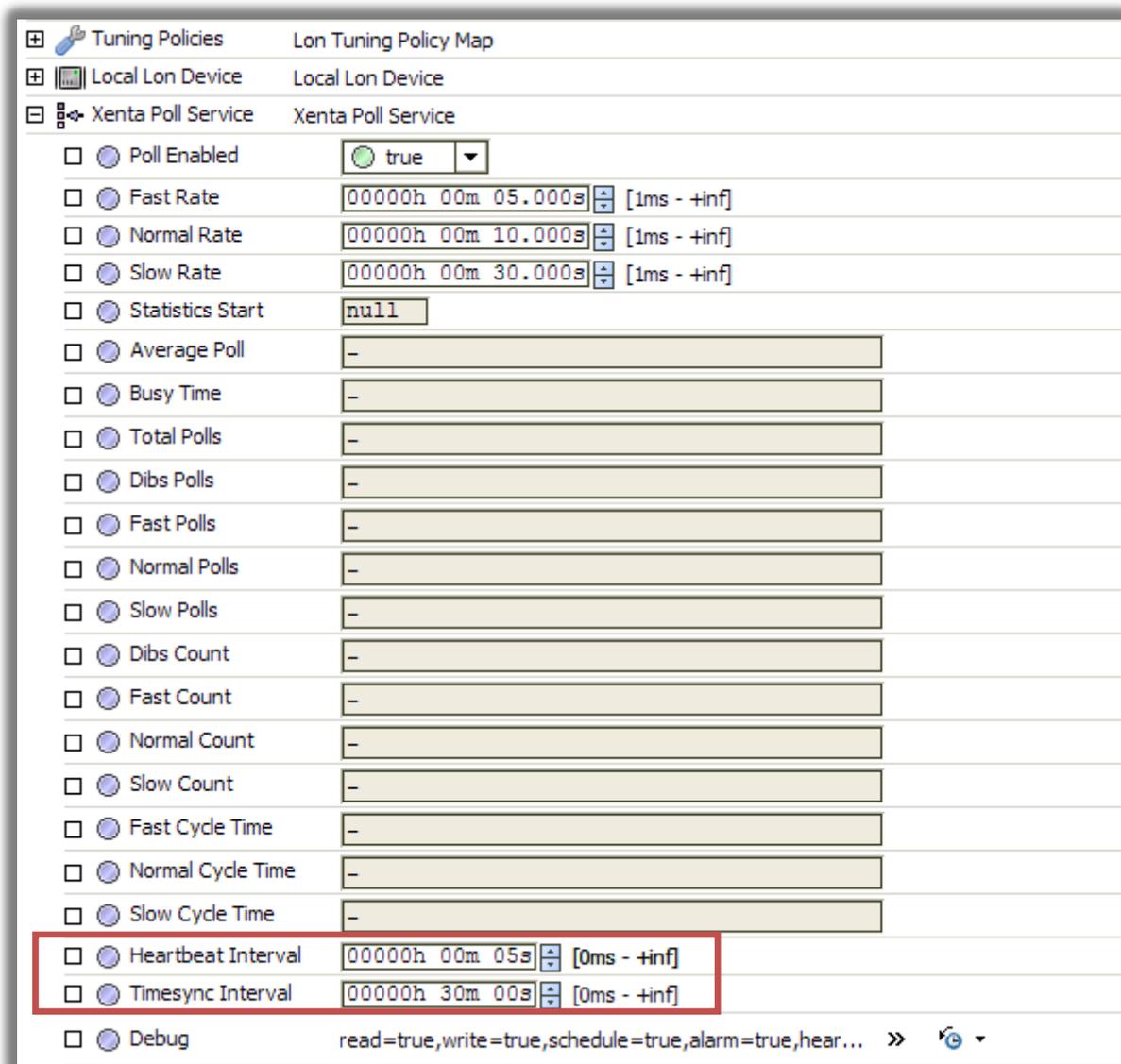
Xenta points and schedules can be read from Niagara and Xenta on the same network. The only consideration is doubling the network traffic, thus polling periods might need to be increased.

Alarming process is more complex. There can be just one master alarm receiver on the network: either Vista or Niagara. It is determined by heartbeat messages, which are periodically sent by the master. If both Vista and Niagara send the heartbeat, some alarms will be sent to Vista and others to Niagara, which will cause confusion. Therefore it is recommended to:

- disable the heartbeat in Niagara during commissioning works: set **Xenta Network >> Xenta Poll Service >> Heartbeat Interval** to 0;
- when the commissioning is completed, re-enable it by setting **Heartbeat Interval** to 5 seconds;
- at the same time disable the heartbeat in Vista: **Lonworks Properties >> Communication Timers >> Heartbeat active**.

Alarms will stop to arrive to Vista and start to arrive to Niagara. Similarly either Vista or Niagara should broadcast time to the network.

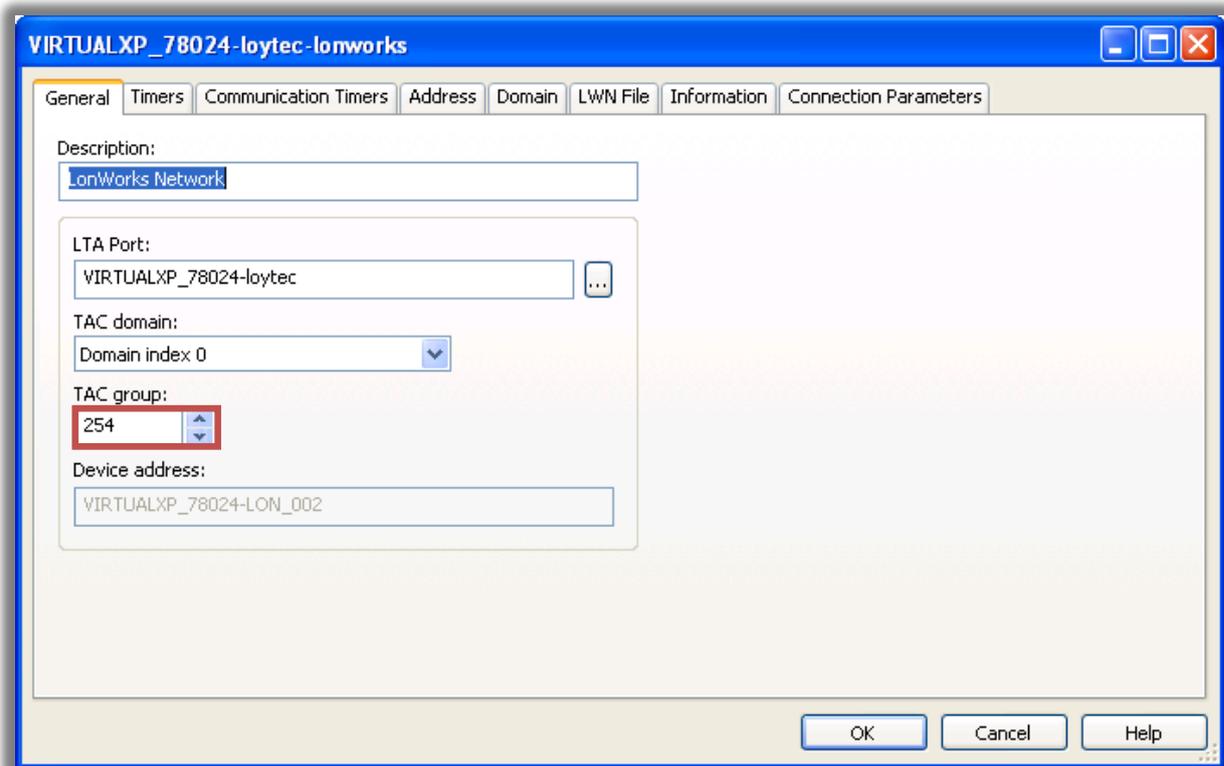
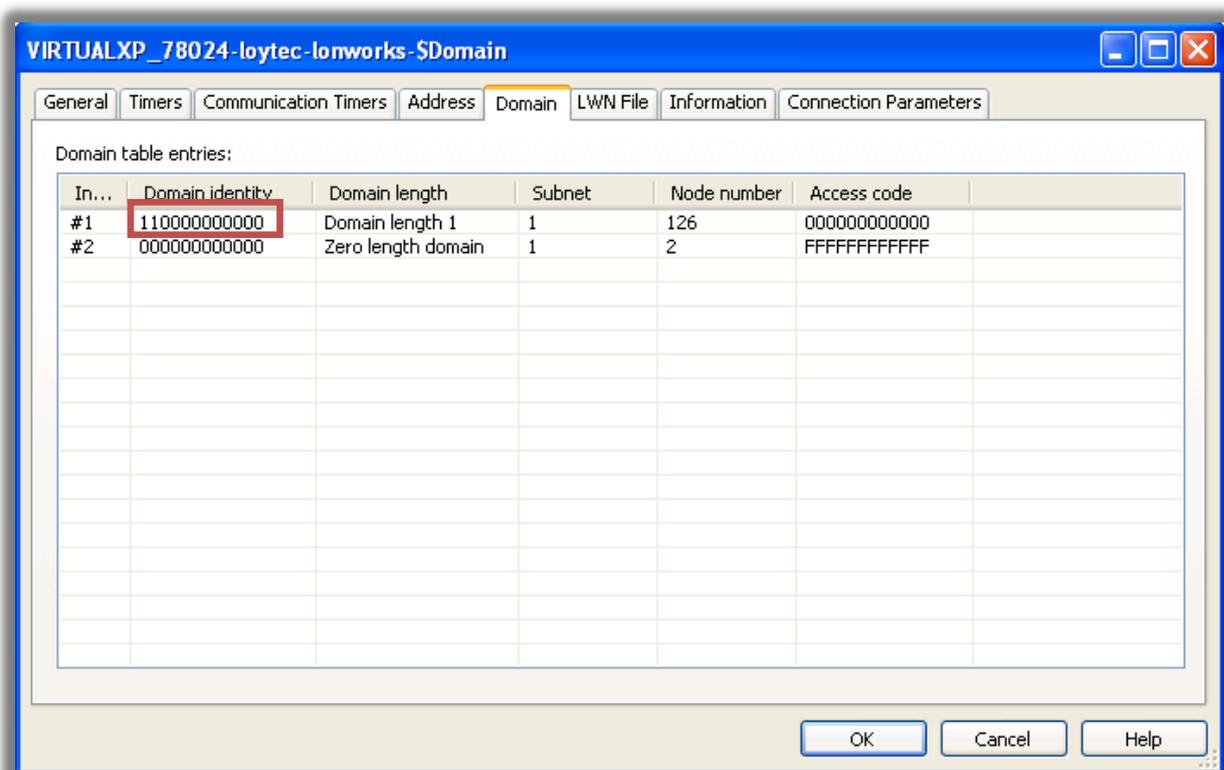




Determine TAC domain and group

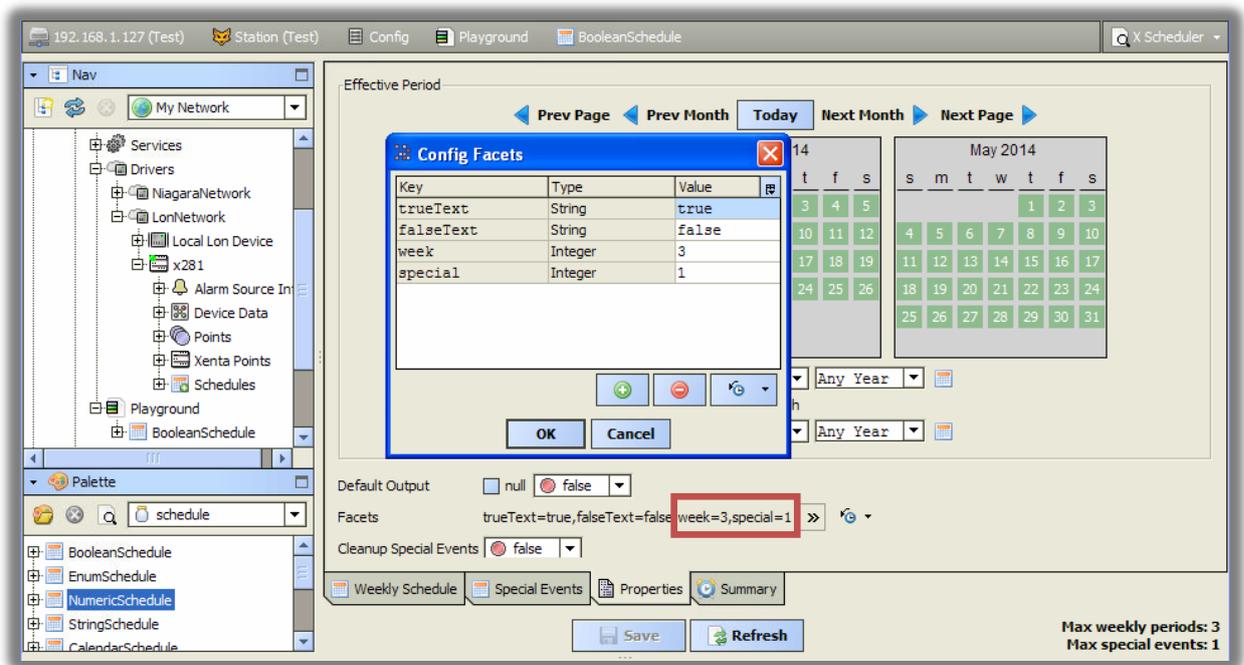
Although Configuration dialog box in **Xenta Device Manager** offers default values for TAC domain ($17_{10} = 11_{16}$) and group (254), on some BMS sites they might be set differently. There are few ways to determine them:

- If already connected to live Xenta network, use Discover to find Xenta controllers (they should be found, because they also have Niagara's default zero-length domain), but don't add them to the database.
Open **Lon Utilities Manager**, select found device, Command: `Data Structs`, SubCommand: `Domain Table` – look for Domain Id with index 0; then SubCommand: `Address Table` – look for Group with index 0.
- Alternatively if there are Vista front-end, check Lonworks properties. Look for Domain identity with index #1 – it is in hexadecimal, translate to decimal; and for TAC group.



Schedule configuration

The maximum amount of weekly periods and special events is determined after the schedule is added to the station. After that the driver will try to read exactly this amount of events every time. If the maximum limits are changed (i.e. Xenta is reprogrammed), delete these facets and launch **Read From Device** action – the driver will determine the new ones automatically during the next import operation.



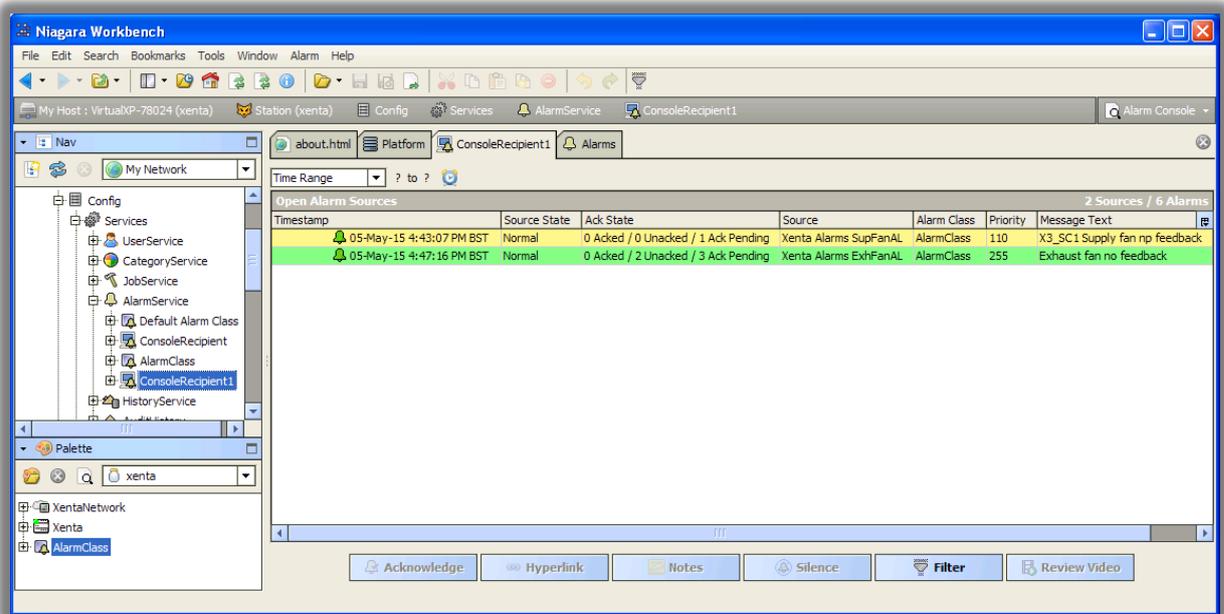
Master Xenta Schedule property **Last Modified** is linked to **Xenta Schedule Export** object action **Write To Device** – this link initiates exporting process as soon as the master schedule is modified via **Xenta Scheduler**. If you don't need this behaviour (e.g. you prefer periodical exports only), delete this link (**Xenta Schedule Export** >> **Link Sheet**).

Reading Exported Schedules

If the schedule can be modified from multiple sources, i.e. Niagara and Xenta OP, then it might be reasonable to periodically import Xenta schedule into master Xenta Schedule. In order to do it, insert **Interval** or **Daily** trigger from **kitControl** palette into the station and link it to **Read From Device** action of **Xenta Schedule Export**. Do not set the interval too short.

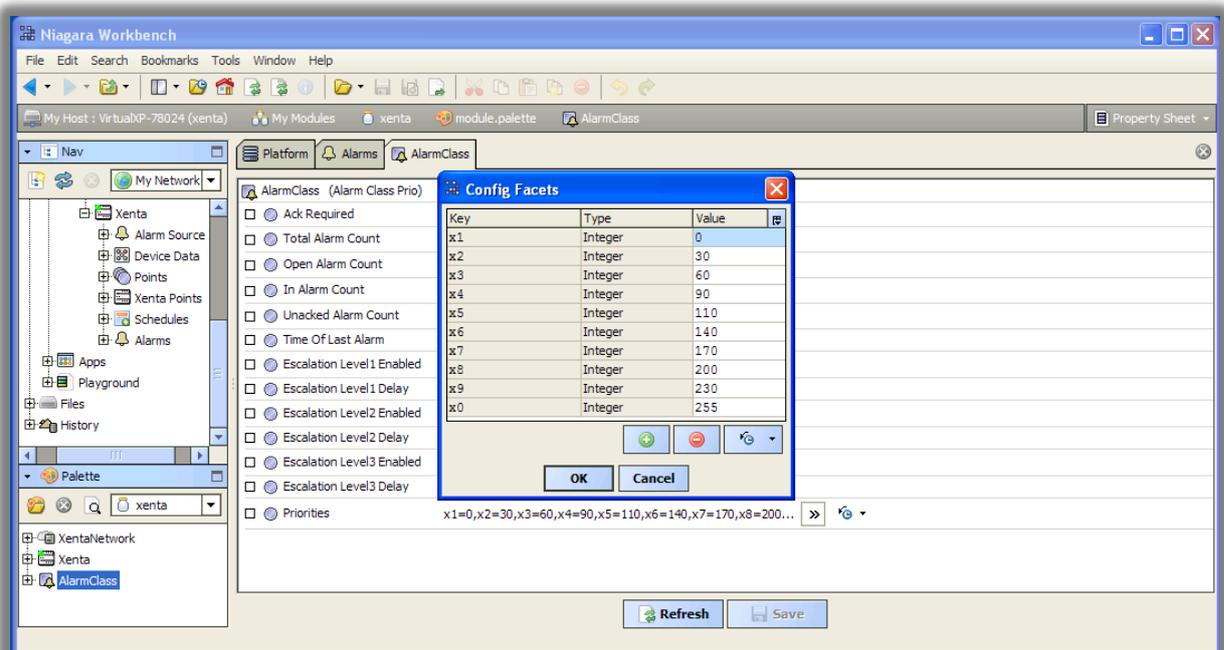
Alarm priorities

As described in **Alarms** section, Xenta alarms have priorities from 0 to 9, which determine the importance of alarm. Niagara alarms have priorities as well, from 0 to 255. Niagara priority is displayed in **Alarm Console** and might be used for colour-coding: see menu **Alarms** >> **Priority Colors**.



Niagara standard Alarm class assigns priorities to alarm records according to its settings, thus Xenta priorities could not be propagated. To avoid it, Xenta priority is stored in alarm record's **Prio** property, so it is still could be displayed in Alarm Console.

In order to use Xenta alarm priorities for colour-coding and recalculate them from Xenta scale (0-9) to Niagara scale (0-255), a special Alarm Class is introduced. It is located in Xenta palette and is identical to Niagara alarm class, except it does not reassign priorities, just recalculates them according to **Priorities** facets: Xenta priority is a key and Niagara priority is a value.



Drag & drop **AlarmClass** from xenta palette to **Alarm Service** and assign this class in **Xenta >> Alarms extension**.