



BACnet Alarm Management



APPLICATION NOTE

October 2015

Description: Guide to understanding, setting up, and processing BACnet Alarms in GENESIS64.

General Requirement: Basic GENESIS64 and BACnet knowledge.

Introduction

The BACnet protocol offers Alarm management features including algorithmic-based generation of real-time alarms (similar to AlarmWorX64 Server), logging these alarms to internal buffer (similar to AlarmWorX64 Logger), and then the forwarding of generated alarm messages to clients (such as GENESIS64). BACnet alarms are generated by BACnet devices, typically controllers. GENESIS64 is a client that can be configured to subscribe to all or subsets of alarm/event messages generated by the BACnet devices. GENESIS64 provide the ability to visualize, acknowledge, and log alarm/event messages that it receives.

BACnet Alarms Overview

A critical component of the BACnet alarm processing is the BACnet Notification Class Object. Any device capable of generating and forwarding BACnet alarms must have at least one Notification Class object. Instances of this object type act as "gateways" for reporting about new real-time alarms. Clients (in our case AlarmWorX64 Viewer or Logger) subscribe to a selected Notification Class. When a new event occurs the Notification Class sends a notification to all subscribed clients. BACnet devices can contain one or more Notification Class objects.

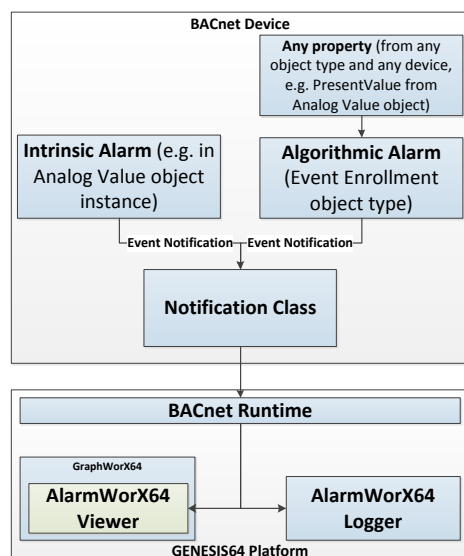


Figure 1 - BACnet Alarms/Event Infrastructure

Intrinsic (Real-time) Alarms

Some object types (Analog Input/Output/Value, Binary Input/Output/Value) support intrinsic alarming. Intrinsic alarming means the object instance itself can be configured to generate event notifications based on object-specific rules (Analog object types are based on Low, Normal and High Limit notifications, Binary object types can create event notifications based on Binary object state).

Algorithmic (Real-time) Alarms – Event Enrollment object

Event Enrollment is an object type allowing you to reference any property from any object instance, located in any BACnet device as a datasource. The Event Enrollment object executes selected algorithms to evaluate alarm conditions on the configured data source. If these specific conditions are met, an event notification is sent to a chosen Notification Class Object. Then it is distributed to subscribers. An Event Enrollment object implements alarm analysis such as high limit, low limit, evaluation on selected values of other objects.

Historical Alarms – Event Log object type

If a BACnet device supports Event Logs, it is able to store a specific number of BACnet events from a Notification Class local to the device. . AlarmWorX64 Viewer can be then used for visualization and navigation through the buffer (historical alarms).

Event States

In the BACnet universe there are 3 standardized event states:

- **Normal** – value is within specified boundaries.
- **Off Normal** – value is out of specified boundaries.
- **Fault** – a problem prevents the device from providing correct event evaluation (e.g. sensor malfunctioned, thus cannot reliably provide a value).

All events generated must be of one of these types.

Subscribing to BACnet Alarms

The following steps are necessary to configure a GENESIS64 system to subscribe to BACnet alarms or events from one or more devices.

1. **Create Notification Class** object in the generating device (if it does not already exist).
2. **Configure Objects** intended to generate alarms.
3. **Add** the BACnet Device **to Alarm Cache**.



BACnet Alarm Management



APPLICATION NOTE

October 2015

4. **Subscribe** for Alarms in AlarmWorX Viewer Control or Alarm Logger.

Creating Notification Class

In order to create a Notification Class please follow these steps:

1. Launch Workbench.
2. Switch to the BACnet provider.
3. Find the desired device in the object tree, right-click it and select the "Add Object to Runtime" option.
4. Choose "Notification Class" in the drop-down list.
5. A configuration form will be opened where an instance ID is automatically assigned (you can overwrite it). Next you need to define state priorities, and whether acknowledgement is required. You can also specify a list of recipients in Recipient List pane – by this you will explicitly tell which BACnet clients (BACnet Runtimes) should receive notifications. By default it will be configured automatically after adding BACnet Device to the GENESIS64 Alarm Cache.
6. Click "Apply" to create a Notification Class.

A device can contain multiple Notification Classes. If you are familiar with the concept of AlarmWorX64 Server, the Notification Classes can be compared to Areas – they can be used to distinguish alarms/events from specific sets of alarm/event sources. You can subscribe to a notification class of a subset of alarms/events. For instance imagine that you have two rooms, each with a set of BACnet-enabled sensors and detectors (temperature, humidity, light intensity, etc.), all of these sensors are connected to a single central BACnet device. You will define that alarms/events from sensors located in Room A will notify through Notification Class instance NC 1 and sensors located in Room B through NC 2. This will allow you to separate and clearly distinguish events raised in separate rooms. Another application of multiple notification class objects is to generate Alarm priorities. A Notification class generates alarms at only one specified priority. All alarms generated by it will be at a specific priority. Consequently a Notification Class Object will be created for each alarm priority and objects that generate alarms will be configured to utilize the appropriate notification class object that generates the alarm at the appropriate priority.

Configuring the Object

For common use please follow next chapter "Setting up intrinsic alarms" covers standard use-case.

Example: Setting up intrinsic (real-time) alarm

In order to have intrinsic alarming working as expected it is required to have a Notification Class already created in the device (you have to know its instance number). Follow the next steps to set it up:

1. Launch Workbench.
2. Switch to the BACnet provider.
3. Find an object supporting intrinsic alarming and select it, For example, select an Analog Value provided by ICONICS BACnet Simulator located on a remote machine.
4. The Runtime tab of the selected object will be opened and you should be able to read and write values of all properties. You have to configure specific values for the following properties (under BACnet / BACnet Intrinsic Alarming section):
 - a. **Notification Class** – instance number (ID) of the Notification Class you want to use for alarm notifications.
 - b. **Time Delay** – how long (in seconds) the value must stay over/under the specified limit before an Event is generated (for instant notification set to 0).
 - c. **Event Enable** – which event transitions will be notified (check at least To Normal and To Off Normal events).
 - d. **Limit Enable** – this property is Analog object type specific – specify which limit should be monitored in order to generate events (in our case check both – High Limit Enable and Low Limit Enable)
 - e. **High Limit** – specify value of High Limit (set 300)
 - f. **Low Limit** – specify value of Low Limit (set 100)

Please note that other Object Types can have different Intrinsic Alarm settings depending on object's purpose (e.g. Binary object types will not have Low and High Limits). But Notification Class, Time Delay, and Event Enable properties should be used for all.

Setting up algorithmic alarm

In this example it is assumed that the device already has a Notification Class and the monitored property already created in the device. An algorithmic alarm is created by creating an Event Enrollment object type configured to analyze the desired alarm/event source property, to apply the desired algorithm to evaluate the alarm condition, and specified to report via the desired Notification Class. To configure please follow these steps:

1. Launch Workbench
2. Switch to the BACnet provider.
3. Find the desired device in the tree,
4. Right-click it and select the "Add Object" item.



BACnet Alarm Management



APPLICATION NOTE

October 2015

5. Choose "Event Enrollment" in the drop-down list and confirm by clicking OK (don't have "Create Empty Object" checkbox checked).
6. In opened form you have to fill in following:
 - a. **Notification Class** - instance number of NC you want to use for alarm notifications.
 - b. **Event Type** - here you choose an algorithm type used for evaluation. Please refer to BACnet specification for details, in our case please set "Out Of Range" Event Type
 - c. **Event Parameters** - Click the "..." button next to "Event Parameters" text box to open the dialog specific for "Out Of Range" event type in order to set its parameters - it uses same parameters as we described in "Setting up intrinsic alarm" section so please refer it for the details.
 - d. **Referenced Object** - Click the "..." button next to "Referenced Object" text box to open the dialog to browse for referenced property whose value is used as a data source, or set Referenced Object values manually using combo boxes and numeric fields.
7. Confirm Event Enrollment object creation by clicking Apply button.

Add BACnet Device to Alarm Cache

In order to receive BACnet Alarms/Events from a device you must configure all such devices to be included in the **Alarm Cache**. You can add devices to the alarm cache during the Device Discovery process by clicking on buttons with alarm indicators as shown in Figure 3.

Another option for devices already added to BACnet cache is right-click on specific device and select "Add to Alarm Cache". You can see whether the device is added by looking at the icon in Figure 2.

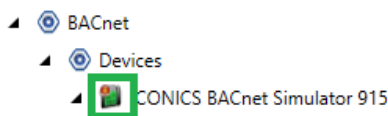


Figure 2 - Icon for device in Alarm Cache

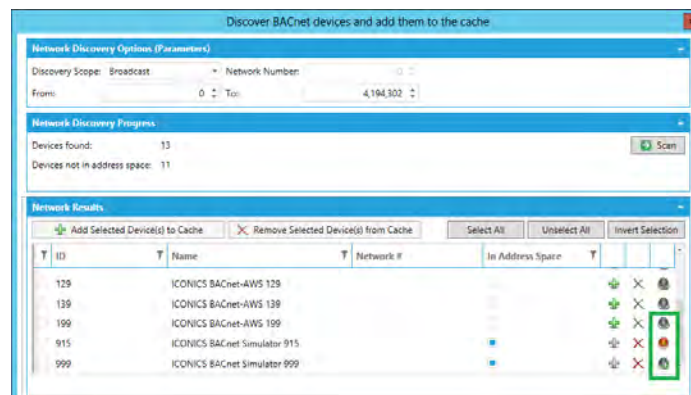


Figure 3 - Subscribing to alarms during device discovery

Subscribing Real-Time BACnet Alarms using AlarmWorX64 Viewer

In order to display BACnet Alarms in AlarmWorX64 Viewer do the following:

1. Launch GraphWorX64.
2. Insert an AlarmWorX64 Viewer control
3. Double-click the viewer control in order to open its properties and settings.

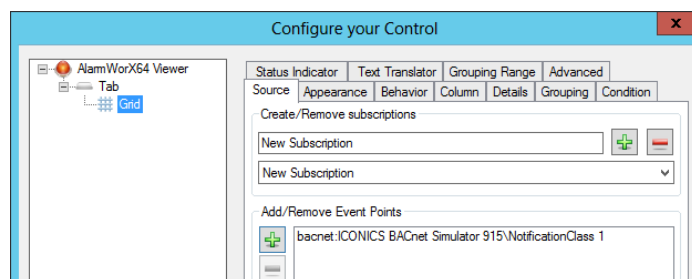


Figure 4 - Configuring BACnet Alarm Viewer

4. In the opened dialog, select the "Grid" tree item.
5. Click the button with the "+" sign in the "Create/Remove Subscriptions" section and choose "Real Time Subscription" from the opened context menu.
6. Select the "Local Alarm Server" item in the "Add/Remove Event Points" section and click the "-" sign in the same section to remove the existing subscription.
7. Now press the "+" button in the "Add/Remove Event Points" section. This will open the Data Browser.
8. In the Data Browser expand My Computer / Data Sources and find BACnet node.
9. Choose your BACnet device from the list to see its provided objects.



BACnet Alarm Management



APPLICATION NOTE

October 2015

10. In the list of objects, find a Notification Class providing notifications about alarms (in Figure 4, it is "Notification Class 1").
11. Confirm selection by clicking "OK" in the Data Browser. This will add the Notification Class subscription to the AlarmWorX64 Viewer.
12. After subscribing to all desired Notification Classes, you can close the dialog window for AlarmWorX64 Viewer's configuration.

You can also **subscribe to a device or the complete BACnet network** in the Data Browser. In this way you can subscribe to all notification classes in the device and for all BACnet devices in the alarm cache.

Your subscription syntax can look like:

BACnet network: bacnet:

Device: bacnet:ICONICS BACnet Simulator 915

Specific NC: bacnet:ICONICS BACnet Simulator 915\NotificationClass 1

NOTE: For easier configuration you can use the Drag and Drop feature. In GraphWorX Configuration mode, open the Data Browser panel and browse for the BACnet node. Then you can select "BACnet" → device → notification class and drag it to the display. This will add an AlarmWorX64 Viewer control that is automatically subscribed to the given notification class.

Subscribing Historical BACnet Alarms using AlarmWorX64 Viewer

In order to display BACnet Historical Alarms - **Event Log object**, in AlarmWorX64 Viewer hosted in GraphWorX64, do the following:

1. Launch GraphWorX64.
2. Insert an AlarmWorX64 Viewer control.
3. Double-click the viewer control in order to open its properties and settings.
4. In the opened dialog, select the "Grid" tree item.
5. Now click the button with the "+" sign in the "Create/Remove Subscriptions" section and choose "Historical Subscription" from the opened context menu.
6. Select the "Local Alarm Server" list box item in the "Add/Remove Event Points" section and click the "-" sign in the same section to remove the existing subscription.
7. Press "+" button in the "Add/Remove Event Points" section. This will open the Data Browser.
8. In the Data Browser expand My Computer / Data Sources and find BACnet node.
9. Choose your BACnet device from the list to see its provided objects.

10. In the list of objects, find an Event Log object providing historical alarms.
11. Confirm your selection by clicking "OK" in the Data Browser. This will add the Event Log historical subscription to the AlarmWorX64 Viewer.
12. After listing all desired Event Log objects, you can close the dialog window for AlarmWorX64 Viewer's configuration.
13. Switch GraphWorX64 to Runtime mode in order to see historical alarms provided by the Event Log object(s).

NOTE: The current implementation supports getting the historical data by polling method only; AlarmWorX64 Viewer specifically asks for new data. This can be forced manually by performing a "Refresh" action on the AlarmWorX64 Viewer control. You can enable automatic polling in AlarmWorX64 Viewer's configuration dialog. Go to the "Behavior" tab, click the checkbox and fill in the input field "Automatically update Historical data after".

Logging BACnet Alarms to SQL Database

You can log BACnet Alarms using AlarmWorX64 Logger. Just select Alarm Subscription: entire BACnet device or specific BACnet device /Notification Classes.

In short, the AlarmWorX Logger's "Alarm Subscription" should look like subscription for Real-time Alarms (see the section entitled "Subscribing Real-Time BACnet Alarms using AlarmWorX64 Viewer", point 12).

Notification Class Subscriptions settings

BACnet Runtime allows user to configure automatic configuration of Notification Class Subscriptions. Under Devices → Communication tab, find the "Notification Class Subscriptions" section. The important options are:

- **Manage Recipients Lists** – if enabled this GENESIS64 server will be automatically added to "Recipients List" of Notification Class objects on the BACnet Device.
- **Valid Days** and **Time Range** – specification when accepting of event notifications will be allowed (e.g. NC2 only on weekend).
- **Accept All / Process ID's** event notifications – Process ID can be a specific "pairing key" with specific alarm/event generating functions in alarm/event generating device. If "only accept ID" is enabled, only notification comes from NC with the same IDs will be accepted. The typical setting is "Accept All"