Part Two

BACnet® Explained

By Grant Wichenko, P.Eng., Member ASHRAE

As discussed in Part 1 of this article, BACnet started with modeling HVAC points such as a fan sail switch to indicate fan status, which is represented in BACnet as a Binary Input object (ON or OFF). A fully modulating two-way valve is represented as an Analog Output object (0% to 100% open). BACnet provides objects and services so users can perform common tasks such as trending, alarming and scheduling. BACnet also offers special objects and services for the fire alarm and security industries.

This article describes some of the enhancements to BACnet. For example, users can log events with the Event Log object type and trend multiple object values with the same time base with the Trend Log Multiple object type. The BACnet committee developed a suite of value objects to represent integers, dates, times and other values beyond the original binary, analog and multistate values. There are new objects and services to permit device interoperability in the lighting industry. The BACnet community wanted a simplified mechanism for devices to alert other devices that an event had taken place. Finally, there are new more granular device profiles for operator workstation devices.

Event Log Object Type

An Event Log object records event notifications with timestamps and other pertinent data in an internal buffer for subsequent retrieval. Each time stamped buffer entry is called an event log "record." Virtually all building automation system (BAS) vendors historically have provided this function in their front end software in a proprietary manner. What was needed is an interoperable solution for event logs.

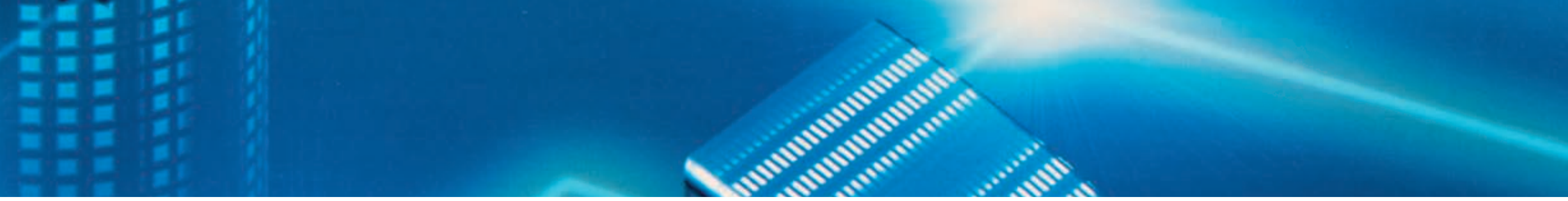
An Event Log object is similar to a Trend Log object. Each Event Log object maintains an internal, optionally fixed-size buffer. This buffer fills or grows as event log records are added. If the buffer becomes full, the least recent records are overwritten when new records are added, or collection may be set to stop. Event log records are transferred as BACnetEventLogRecords using the ReadRange service developed for the Trend Log Object Type. The buffer may be cleared by writing a zero to the Record_Count property. The user can decide what notifications are placed into the log. There is a BACnet Interoperability Building Block (BIBB) that uses this object. More details on this BIBB are available in the standard.

Trend Log Multiple Object Type

The Trend Log Multiple object type works the same way as the original Trend Log object type except that multiple values can be trended at the same time. Before this object type was added to the BACnet standard, each trend log had to

About the Author

Grant Wichenko is president of Appin Associates in Winnipeg, MB, Canada. He is a member of SSPC135, the BACnet Committee and SGPC13, the BAS Specification Committee, and chair of the Application Profile Working Group.



be created individually. Most BAS vendors have supported this trend log multiple functionality in their front ends in a proprietary manner. This new object type allows this functionality to be provided by vendors in an interoperable manner so a user can trend multiple points on a common time period from the same BACnet device or from different devices supplied by multiple vendors.

For example, the Trend Log Multiple object type allows a user to trend outdoor air temperature in one vendor's controller and boiler supply water setpoint in another vendor's controller. A third trend log such as Btu consumption from yet another vendor's BACnet Btu meter can be added to this object. These three trends from different devices are trended with a common time base. This example shows how Trend Log Multiple will help users analyse boiler plant efficiency in an interoperable manner.

There is a suite of BIBBs for this object that is similar to the Trend Log object type.

New Value Object Types

The new suite of Value objects now allow for more than just the binary, analog and multistate values to be represented in a network visible manner to other BACnet devices. Here is a brief description of these new value objects.

CharacterString Value. This value object allows a device manufacturer to represent a character string such as a HIGH DRIVE TEMPERATURE ALARM!!! in a variable speed drive as the present value of an object, instead of as a Multistate Value of say, 27, that needs to be converted into human understandable text for the user.

Large Analog Value. A BACnet device can use a Large Analog Value object to make any kind of double precision data value accessible to other BACnet devices. One excellent use for this object is to represent kWh from an electricity meter.

Integer Value. This object allows one to represent a negative integer such as –1238 or a positive integer such as +8321. This object type can be used to represent positive values for clockwise rpm and negative values for counterclockwise rpm.

Positive Integer Value. This object allows one to represent only a positive integer such as +1238. This object type could represent count of the number of office spaces in a building calling for heating.

Time Value. This object allows one to represent a single specific time value like 12:34:56.77.

Date Value. This object allows one to represent a single specific date like 07-August-2013.

Time Pattern Value. Time Pattern Value objects allow for wildcard characters like the asterisk (*) to represent any value in the field. The wildcard value of the Time Pattern Value object could have a present value of 12.*.*.*. The asterisks between the periods represent any minute, second or hundredths of a second.

Date Pattern Value. The wildcard value of the Date Pattern Value object could have a present value of 07-*-2013. The asterisk in this case represents any day in July.

DateTime Value. This object allows one to represent a single specific date and time like 07-August-2013, 12:34:56.77.

DateTime Pattern Value. By combining the concepts of the Date and Time Value objects together, the present value of this object could look like 07-*-2013, 12.*.*.*. Besides representing any value, the asterisk can represent even or odd days or months.

BitString and OctetString Values are mentioned for completeness but are not covered in this article. The BACnet standard contains more details on these object types.

Global Group Object Type

The Global Group object type defines a standardized means of representing a collection of properties from one or more objects. A Global Group object is used to simplify the exchange of information between BACnet devices by providing a shorthand way to specify all members of the group at once (e.g., all space temperatures in a building). The Global Group also supports intrinsic event reporting, and it exposes a method for sending periodic Change of Value (COV) notifications. This object type can initiate an event notification message conveying the values of all of the members of the group at once.

A Global Group object type differs from a Group object types in that its members are not restricted to the same device where the Group object is located. Global Group members can be from any device on the BACnet internetwork.

This object type also differs from the Structured View object type. The Global Group gives present values of referenced object properties whereas Structured View defines the object location, allowing the client device to find the object and read its present value.

Notification Forwarder Object Type

The Notification Forwarder object type allows devices that can distribute notifications to a small number of destinations to have their notifications received by many destinations. This object types differs from a Notification Class object type in that the Notification Forwarder object type is not used for originating event notifications, but rather is used to forward event notifications to a different and potentially larger number of recipients.

Figure 1 shows how this object works. It can be thought of as a list of recipients who must be notified when the burglar alarm goes off. The building security system sends the alarm to the alarm company. The company (representing the Notification Forwarder object type in the figure) then notifies staff and the police that an alarm has occurred.

This object is ideally suited to small devices, such as VAV boxes. The box sends the notification to a supervisory controller, which in turn sends the event to a Recipient_List in a Notification Class object.

Alert Enrollment Object Type

This object provides an easy way for a device to notify another device that an event has occurred. Alerts give users flexibility as they do not use BACnet's standard event algorithms such as

CHANGE_OF_STATE and OUT_OF_RANGE. "Information alerts" are not a replacement for BACnet alarms. Alerts are suited for sending maintenance notifications to another device.

BACnet Objects for Lighting

The original plan to meet the needs of the lighting industry was to extend the existing BACnet objects like the Binary and Analog Output object types. It soon became clear that this was the wrong approach and that new lighting centric objects were needed. Lighting specific strategies such as "blink-warn," ramping up to a specific brightness level or fading the light to off could not be easily accommodated by enhancing the standard existing BACnet objects.

Lighting Output Object Type

The Lighting Output object type defines a standardized object whose properties represent the externally visible characteristics of a lighting output and includes dedicated functionality specific to lighting control that would otherwise require explicit programming.

The lighting output is analog in nature. The physical output level, or nonnormalized range, is specified as the linearized percentage (0% to 100%).

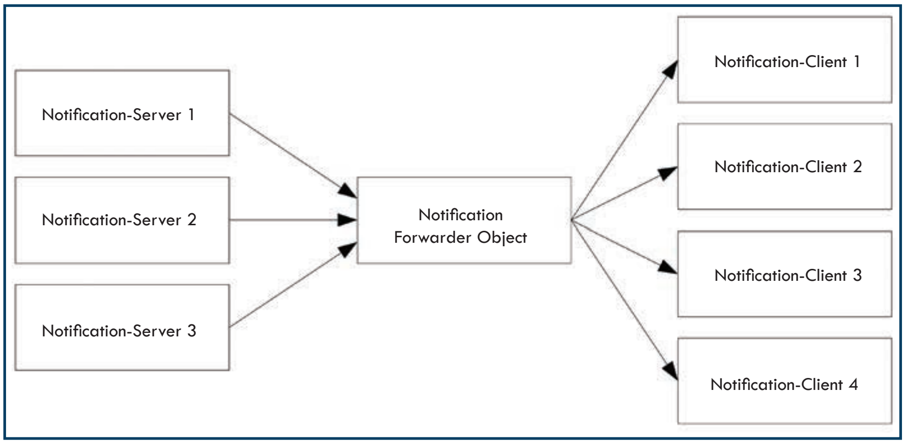


Figure 1: Example of multiple Notification-servers routing to multiple Notification-Clients. (Source: Figure 12-7 from SSPC 135-2012).

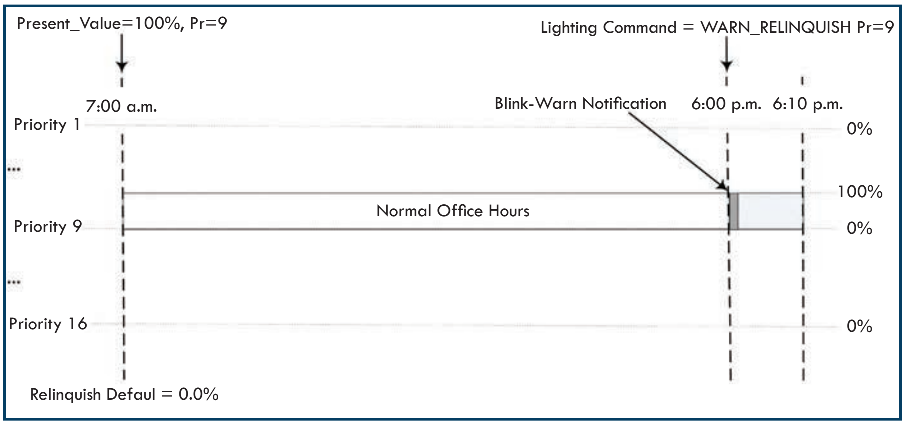


Figure 2: Daily schedule with Blink-Warn example. (Source: Figure 12-14 from SSPC 135-2012).

The actual value may range from 5% to 85%. These values can be defined as a subset of physical output levels defined by Min_Actual_Value and Max_Actual_Value.

The level of the lights can be changed directly to an absolute level by writing to the Present_Value. This property is commandable and uses BACnet's 16 level priority array mechanism to arbitrate between multiple writers to the lighting output. This notion of having more than one device writing to an object is not used much in the HVAC world but is commonplace in the lighting world. The light might start with defined brightness level during occupancy. This level could be reduced by a daylighting sensor if the sun is shining into the office space. There may be a lighting load shedding strategy controlled by a Load Control object that could then reduce the lighting output. The local user normally has the final say and may set the brightness of the room lights through a local controller on the wall.

The Lighting_Command is a writeable property that provides additional lighting functionality with special lighting-specific functions such as ramping, stepping and fading. The Lighting_Command also provides a blink-warn mechanism to notify room occupants that the lights are about to turn off. The blink-warning mechanism is internal to the Lighting Output object type and may cause the physical lights to blink on and off or issue a notification in some other manner.

The following example illustrates how a Lighting Output object type may be used in a typical office scenario.

Prior to 7 a.m. the lights are off as the Lighting Output object type is being controlled at the relinquish default value of 0.0% or off. At 7:00 a.m. a scheduler (e.g., a BACnet Schedule object or other automated process) turns the physical lights on by writing 100% to the Present_Value property at priority 9. At 6 p.m. a WARN_RELINQUISH lighting command is executed at priority 9, which causes an immediate blinkwarn notification to occur but leaves the lights on until the egress timer has expired. Assuming a 10 minute (600 seconds) egress time is specified, the value at priority 9 is relinquished at 6:10 p.m., and the Present_Value returns to 0%. This scenario is shown in *Figure 2*.

Channel Object Type

The Channel Object was needed for the lighting industry to allow for writing a lighting brightness value to as many as dozens of individual lights in many lighting controllers quickly. The existing WritePropertyMultiple service requests could be used but this approach would not provide the fast response needed for the lighting industry.

The BACnet committee developed the WriteGroup service to change properties in objects in multiple devices at the same time. To do this, the WriteGroup is broadcast in a way that allows many devices to hear the same request simultaneously. For this to make sense and not cause chaos, the WriteGroup service includes a group number parameter that restricts the effect to only those receiving devices that are members of that group. The WriteGroup service further restricts the targets for writing to those Channel objects within those devices that are associated with the specified channel number(s).

In *Figure 3*, the writing device is writing a value to Group 27, Channel 11. Devices 1 and 2 receive the write request. Device 2's Channel object is Group 14, Channel 11 so it does not execute the write, whereas Device 1 does execute the write.

Facilities management personnel benefit from a BACnet-based lighting control system as they can operate this system and other BACnet-based systems, such as HVAC, through a common front end. At the device level, a BACnet-based VAV box in a room can read the state of the occupancy sensors provided by the BACnet-based lighting system. The VAV box can reduce the air flow to the space when the space is considered to be unoccupied to achieve part load occupancy savings needed by high performance buildings. Separate HVAC occupancy sensors are not needed as the sensor data can be shared through BACnet.

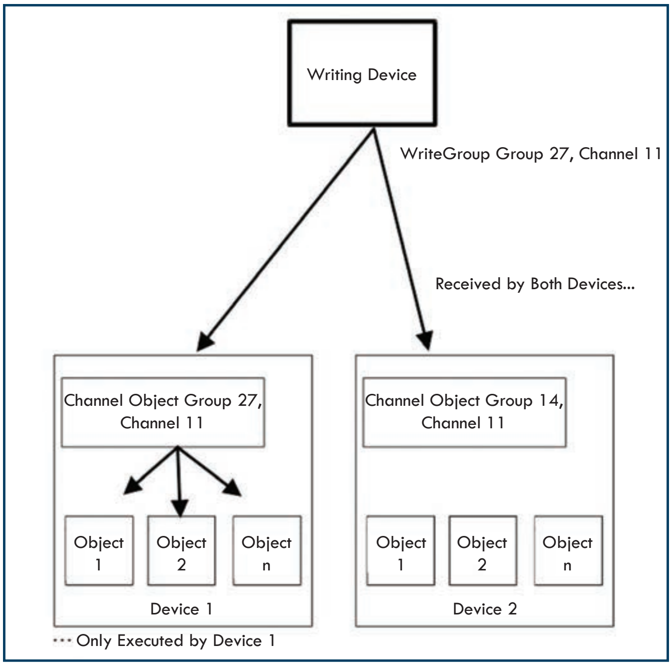

Writing Device
WriteGroup Group 27, Channel 11
Received by Both Devices...
Channel Object Group 27, Channel 11
Object 1
Object 2
Object n
Device 1
Channel Object Group 14, Channel 11
Object 1
Object 2
Object n
Device 2
Only Executed by Device 1

Figure 3: Control Groups limit WriteGroup effect to specific Channel objects across many devices. (Source: Figure 12-11 from SGPC 135-2012).

New Device Profiles

BACnet has had device profiles for some time. These device profiles set the minimum BIBB support requirements and are useful in specifying project requirements.

The major change in device profiles was to separate the original BACnet Operator Work Station (B-OWS) into three device profiles called the B-AWS (Advanced Workstation), B-OWS (Operator Workstation) and the B-OD (Operator Display). This was done because the BACnet community wanted different devices that met specific BIBBs rather than having to support all operator-related BIBBs in one device.

The B-AWS supports all the B-OWS BIBBs, plus the following:

- Event Log BIBB,
- Advanced View Modify BIBBs for data sharing, scheduling, trending and scheduling that allows writing to other object properties not supported in the B-OWS, and
- Automatic Network Mapping, Device Mapping, Object Create and Delete and other device management BIBBs.

The B-OD is a device that is an LCD screen usually attached to BAS panel in a mechanical room. The B-OD supports the ability to read and write data so the user does not have to carry a laptop around and connect to a BAS panel.

Conclusion

Since its release in 1995, ASHRAE BACnet Standard 135 has undergone substantial enhancements. The standard had 538 pages in 1995. The 2012 version has 1,039 pages. The standard was republished with updates every four years. Now, the standard is on a two-year publication cycle. More new BACnet objects and services will be in the 2014 version of the standard. Stay tuned! ■