

How to Specify BACnet-Based Systems

Larry K. Haakenstad

Alerton Technologies, Inc.

BACnet is an ANSI/ASHRAE standard that specifies a common communication protocol that allows building systems to communicate with each other using a common language. By implementing BACnet, systems made by different companies can communicate with each other. This technology provides the building owner with more choices when it comes to choosing and integrating whole building systems, and even individual building automation components when new automation systems are to be installed or added to a building or campus-wide system.

There are a number of issues that must be considered when specifying a BACnet-based system. These issues focus on selecting BACnet options, which include system architectures. A very important choice in the selection of system architecture is whether or not to allow a gateway to a non-BACnet-based system, or system components, and how to specify this BACnet gateway. The basics of any control system also must not be forgotten; that is, the sequence of operations of the equipment, in addition to the look and feel of the control system.

Specifying a BACnet system cannot be done by simply stating "System shall use BACnet communication protocols." This is a very ambiguous statement that can result in a system that has a completely unexpected architecture. It is very important to give details regarding architecture and system expectations when specifying a BACnet-based system.

Issues That BACnet Addresses

BACnet defines, in detail, the communication standard for the control system. BACnet defines specific LAN types and many details of the communication protocol. These details include—message structure, communication services, objects, and properties of objects. These specifics allow proprietary services and objects to be passed over the BACnet LANs without interfering with standard BACnet items. Some optional items must be chosen for the system to interface with the building and other building components. Specifying LAN types is very important and usually straightforward. Specifying details within the communication protocol is also important, but the resulting ramifications are not always apparent without some thought and consideration.

Issues That BACnet Does Not Address

BACnet addresses all communication issues in a project, but it does not address a number of other issues that are crucial. A complete specification must address the look and feel of the system, which physical points are connected to the system, the sequence of operation for the controlled equipment, and the system architecture. These issues are addressed in current specifications and must still be included in a specification for a BACnet system.

Programming and setup tools are important parts of the specifications for a BACnet system and must be included for components or systems from each vendor. System commissioning and future additions will go smoothly if all tools are included for every vendor's components or systems. Specifying tools that are simple and easy to use in a graphical environment reduces commissioning and set-up time. A common operating system like Windows[®], or Windows NT[®], allow operation of all tools on one computer, with no lapse in system operation or performance.

All of the issues not addressed by the BACnet standard are already included in any complete specification that is written for a legacy or proprietary system. These items will still need to be included when specifying

a BACnet system. However, there are other issues that BACnet does specify as optional, and it is important to specify them completely.

Choosing a System Architecture Using BACnet

There are many ways to lay out a system using BACnet components throughout. The entire system can use components from a single vendor. This has the advantages of sole-source responsibility and compatibility with future systems, and gives you the choice of adding equipment from other vendors in the future. However, this approach does not necessarily take advantage of all of the BACnet benefits.

A system from one vendor may be supplied with interfaces to a few other controllers, such as chiller controllers or boiler controllers. This provides the advantage of sole-source responsibility, without necessary custom translation hardware for interfaces to those crucial pieces of equipment. A system architecture configured this way takes advantage of one of the biggest reasons to use BACnet devices on a project: protocol compatibility between manufacturers without additional hardware based on custom or seldom-used software.

BACnet may also be used to interface the HVAC control system to other building systems, such as fire systems, access control systems, security systems, etc. This architecture allows a building owner to choose the preferred systems of his/her choice, and BACnet allows integration of these systems without developing or installing custom gateways. All information may be directly passed between the different systems over the BACnet LAN interconnection.

A system can be comprised of components from multiple vendors. This has the advantage of the best feature-to-cost ratio. However, the specification must clearly address who is responsible for integrating each component into the system and who will provide all the necessary set-up and programming tools for each component. If this is not defined, there can be holes in the startup and commissioning phase that leave a system partially operating.

A system can also use an architecture that is BACnet-based at higher levels and non-BACnet-based at lower levels. This lets a system interface to other BACnet equipment at the higher levels, but requires a gateway device to translate data between the non-BACnet protocol and BACnet. Generally, the gateway complicates the system design and increases costs because of the extra hardware, set-up, and programming time involved with a gateway device.

This possible variety of system architectures is why it is so important to specify the type of architecture that fits your particular project. The objectives of the system must be examined carefully, like current systems, before defining the architecture. BACnet provides this flexibility.

Specifying BACnet LAN Types

The BACnet standard defines five LAN types for system communications: Ethernet, ARCNET, Master Slave/Token Passing (MS/TP), Point-to-Point (PTP) and LonTalk. Ethernet and ARCNET are used as backbones for the system. They are standard LANs—typically used in computer networks—that can transfer large amounts of data quickly. MS/TP and LonTalk are typically used as interfaces to field controllers because of the lower installation cost. PTP is used as a direct connection point for computers and modems. Details of LAN types can be found in the BACnet standard and in some of the many articles written on the subject.

The first decision regarding LAN types concerns the backbone of the system. This decision is influenced by the existing network, or whether the owner wants to use the general computer network as the automation system backbone. Items to consider include the reliability of the computer network, and how critical its reliability is to control system performance. Installation costs of the BACnet LAN types vary and the choice between coax installation or twisted-wire connections must also be considered. In a campus-wide system,

using an existing LAN significantly reduces costs and is almost a necessity for quick and efficient system installation.

Choosing the LAN type for other parts of the control system is generally not as critical or complicated; in many architecture scenarios, it can be left to the supplier of the system. MS/TP is a low-cost LAN installed using a twisted pair of wires from unit to unit. LonTalk supports a number of different transport media—twisted pair and coax, for example—that the supplier or specifier can select. However, do not confuse LonTalk with LonWorks. LonWorks is a complete protocol specification, while LonTalk is only the transport medium for the system. LonTalk is part of the BACnet standard, while LonWorks is not.

Direct connection points using a computer are made through the point-to-point LAN. The specifier should detail locations of these PTP connections and also whether modem connections are desired. Typically one modem connection is necessary for each system. Laptop computer connections are provided at each global panel.

The last consideration when specifying LAN types involves interfaces between different manufacturers' systems. One manufacturer may use MS/TP for its controller interface, while another may use ARCnet. It is simple to add a router for integration between these two LAN types. Most important in a BACnet-based specification is to define who is responsible for providing systems components, such as, routers, set-up tools, and start-up/job commissioning.

BACnet Services and Messages

"Services are the means by which one BACnet device acquires information from another device, commands another device to perform certain actions, or announces to one or more devices that some event has taken place." In other words, services control the activity throughout the BACnet LANs to assure that messages and commands arrive at the intended destination. One service may read a single piece of information from a controller, while another may tell a device to shut down and then start up again.

The important thing to remember in specifying a BACnet system is that if one BACnet device supports a given service and another device does not, the two devices cannot use that particular service to communicate with each other. This may or may not be important, depending on what service is not supported. If both devices support another service that will serve a similar role, then this common service can be used.

For example, one device might support the ReadPropertyMultiple service and another device might not. (The ReadPropertyMultiple service reads multiple BACnet objects and properties from a device using one command.) Using multiple ReadProperty services that return the value of one property of one object can duplicate the function of ReadPropertyMultiple service. However, both devices need to support the ReadProperty service or there will be more activity over the LAN because the ReadProperty service must be sent multiple times.

It is tempting to specify that every BACnet device in a system should support every BACnet service. The BACnet standard addresses all system components from the operator workstation to unitary controllers and even smart sensors, so there are many services that need to be implemented at the higher level but are unnecessary at the unitary controller level. Implementing services at the lower levels has a direct effect on processing power and memory, which directly affects the cost of these products.

Most specifications need a general way to specify the services required at each system level. To do this, the BACnet standard defines six general conformance classes. Each conformance class has a minimum number of services that must be implemented for the device to be in its specific class. A specification should define the desired conformance class of each BACnet device in the system. This will assure that devices from two different vendors will be able to communicate with each other.

BACnet Objects and Properties of Objects

BACnet defines each physical point, or software value, in a system as an object. The most commonly used objects are analog input, analog output, analog value, binary input, binary output, and binary value objects. All objects have properties associated with them including, present value, description, status, units, and so on.^{iv} Each object has required and optional properties. These are detailed in Appendix C of the BACnet standard for every object type.

The specification, or drawing set, should list each physical connection that will be made to the system, just like the point list in specifications today. Ideally, this list would also include any software parameters associated with each connection point such as, alarms, alarm limits, setpoints, etc. Each of the items listed is treated as an object, with its associated properties within the BACnet system. Again, all required properties for a given object type must be supported by the BACnet device.

A vital aspect of specifying objects emerges when interfacing controllers from two or more different manufacturers. If the building control system is to communicate with a BACnet-based controller on the chiller, the objects to be passed between the chiller controller and the building control system must be defined. Specifying all objects that are to be passed to and from the controllers assures that the BACnet objects needed for system implementation will be provided by the manufacturers at their respective interfaces. The same philosophy should be adhered to when systems are being integrated.

A specification might ignore support of optional properties; in many cases this is appropriate, because the optional properties are not needed. However, if optional properties are necessary for control or if optional properties are desired because of function, the specification must include those properties in the definition of each object type. If optional properties are not needed on a project, but are supported by a BACnet device, they do not degrade system performance.

Proprietary Services and Objects

BACnet lets manufacturers define proprietary services and objects as an extension to the BACnet standard. This BACnet feature allows manufacturers to add special functions to their systems without impacting any of the standard BACnet communication that occurs on the BACnet LAN. However, implementing and relying on a proprietary service or object means that other manufacturers cannot access functions that use the proprietary service or object. If the vendor utilizing proprietary BACnet objects, services, and so on, provides detailed descriptions in BACnet-standard format, then other vendors can also implement these extensions. This should be carefully specified.

The specifier must decide whether proprietary services and objects are allowed for a project. This should be clearly defined in the specification. If proprietary services or objects are allowed, the specification should call for submittal of these proprietary items. This will allow another manufacturer to implement the proprietary services and objects in its system for complete system integration.

Gateways to Non-BACnet Systems and Components

Gateways are devices that translate information from a non-BACnet protocol to one that is BACnet-compatible. Each gateway must interface with a BACnet LAN and also with the LAN used by the non-BACnet system. A gateway converts the information from the non-BACnet system into properties of BACnet objects. This conversion process must be programmed for the application; typically, each object is defined on a project by project basis. The gateway, when properly connected and configured, integrates non-BACnet systems or components into a BACnet-based system.

Specifications must address a number of issues when dealing with gateways. One such issue is the throughput of the gateway. The translation process takes some time to perform and will vary depending on the number of objects moving through the gateway. The LAN types attached to each side of the gateway may also influence throughput. If the non-BACnet system is passing large amounts of data, a high speed BACnet LAN—like Ethernet or ARCNET—should be used on the BACnet side. Slower LANs will not have the capacity to pass large amounts of data effectively.

It is critical to completely define the objects and properties that are to be passed to the BACnet system. Gateways are typically configured to fit a particular project. If a piece of information is not specified as a BACnet object with its associated properties, it will likely not be present on the BACnet side.

BACnet Evaluation Tool

As discussed above, specifying correctly ensures that the proper product and system are defined for a given project. The second part of ensuring that system installation goes smoothly is the review of product submittal data to verify that each product meets the specification. Commissioning the system is the final step in ensuring that a complete, high-performance system is installed.

Any product submittal for a BACnet based system should include the BACnet-defined Protocol Implementation Conformance Statement (PICS). The PICS defines each product's capability. It begins with a product description, and details the LAN type (Data Link Layer Option) and conformance class of the device. PICS' also include the services supported, object types, and, if required in the specification, descriptions of any proprietary services or objects.

The PICS also includes a section on the functional groups supported. Functional groups are groupings of specific services needed to perform certain functions and can be used to specify general requirements for a device. Finally, the PICS includes character sets supported, special functionality, router capabilities, and property range restrictions. These items may or may not be important for a specific project.

A device's PICS defines for the specifier how that device implements BACnet. It is as important as standard data specification sheets when reviewing a submittal for BACnet-based components or systems.

A sample Protocol Implementation Conformance Statement:

Unitary Controller

BACnet Protocol Implementation Conformance Statement

Vendor Name:

Product Name: **Unitary Controller**

Product Model Number: **BACnet-UC**

1. Product Description

The BACnet-UC is a versatile BACnet Unitary Controller designed for heat pump, air conditioning and fan coil applications. It operates either as a stand-alone controller or as part of a building-wide integrated system. Binary and analog outputs are available to control heating and cooling stages, economizers, modulating valves, and other devices. The control logic is fully programmable, thus the controller can be easily adapted to a wide variety of applications.

-

2. BACnet Conformance Class Supported

Class 1	<input type="checkbox"/>	Class 4	<input type="checkbox"/>
	<input checked="" type="checkbox"/>		<input type="checkbox"/>

Class 2	<input type="checkbox"/>	Class 5	<input type="checkbox"/>
Class 3	<input type="checkbox"/>	Class 6	<input type="checkbox"/>

3. BACnet Functional Groups Supported

Clock	<input type="checkbox"/>	Files	<input checked="" type="checkbox"/>
HHWS	<input type="checkbox"/>	Reinitialize	<input type="checkbox"/>
PCWS	<input type="checkbox"/>	Virtual Operator Interface	<input type="checkbox"/>
Event Initiation	<input type="checkbox"/>	Virtual Terminal	<input type="checkbox"/>
Event Response	<input type="checkbox"/>	Device Communications	<input type="checkbox"/>
COV Event Initiation	<input type="checkbox"/>	Time Master	<input type="checkbox"/>
COV Event Response	<input type="checkbox"/>		

4. BACnet Standard Application Services Supported

Application Service	Initiates Requests	Executes Requests
Acknowledge Alarm	<input type="checkbox"/>	<input type="checkbox"/>
ConfirmedCOVNotification	<input type="checkbox"/>	<input type="checkbox"/>
ConfirmedEventNotification	<input type="checkbox"/>	<input type="checkbox"/>
GetAlarmSummary	<input type="checkbox"/>	<input type="checkbox"/>
GetEnrollmentSummary	<input type="checkbox"/>	<input type="checkbox"/>
SubscribeCOV	<input type="checkbox"/>	<input type="checkbox"/>
UnconfirmedCOVNotification	<input type="checkbox"/>	<input type="checkbox"/>
UnconfirmedEventNotification	<input type="checkbox"/>	<input type="checkbox"/>
AtomicReadFile	<input type="checkbox"/>	<input checked="" type="checkbox"/>
AtomicWriteFile	<input type="checkbox"/>	<input checked="" type="checkbox"/>
AddListElement	<input type="checkbox"/>	<input checked="" type="checkbox"/>
RemoveListElement	<input type="checkbox"/>	<input checked="" type="checkbox"/>
CreateObject	<input type="checkbox"/>	<input type="checkbox"/>
DeleteObject	<input type="checkbox"/>	<input type="checkbox"/>
	<input type="checkbox"/>	<input checked="" type="checkbox"/>

ReadProperty	-	-
ReadPropertyConditional	<input type="checkbox"/>	<input type="checkbox"/>
ReadPropertyMultiple	<input type="checkbox"/>	<input type="checkbox"/>
WriteProperty	<input type="checkbox"/>	<input checked="" type="checkbox"/>
WritePropertyMultiple	<input type="checkbox"/>	<input type="checkbox"/>
DeviceCommunicationControl	<input type="checkbox"/>	<input type="checkbox"/>
ConfirmedPrivateTransfer	<input type="checkbox"/>	<input type="checkbox"/>
UnconfirmedPrivateTransfer	<input type="checkbox"/>	<input type="checkbox"/>
ReinitializeDevice	<input type="checkbox"/>	<input type="checkbox"/>
ConfirmedTextMessage	<input type="checkbox"/>	<input type="checkbox"/>
UnconfirmedTextMessage	<input type="checkbox"/>	<input type="checkbox"/>
TimeSynchronization	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Who-Has	<input type="checkbox"/>	<input type="checkbox"/>
I-Have	<input type="checkbox"/>	<input type="checkbox"/>
Who-Is	<input type="checkbox"/>	<input type="checkbox"/>
I-Am	<input type="checkbox"/>	<input type="checkbox"/>
VT-Open	<input type="checkbox"/>	<input type="checkbox"/>
VT-Close	<input type="checkbox"/>	<input type="checkbox"/>
VT-Data	<input type="checkbox"/>	<input type="checkbox"/>
Authenticate	<input type="checkbox"/>	<input type="checkbox"/>
Request Key	<input type="checkbox"/>	<input type="checkbox"/>

5. Standard Object Types Supported

Object-Type	Supported	Dynamically Creatable	Dynamically Deletable	Optional Properties Supported	Writable Properties
--------------------	------------------	------------------------------	------------------------------	--------------------------------------	----------------------------

Analog Input	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Description	
Analog Output	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Description	
Analog Value	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Description	Present_Value
Binary Input	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Description	
Binary Output	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Description	
Binary Value	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Description	Present_Value
Calendar	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
Command	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
Device	Yes	N/A	N/A	Description,	
				Location,	Location,
				Local_Time,	Local_Time,
				Local_Date,	Local_Date
				Daylight_Savings_Status	
Event Enrollment	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
File	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Description	Archive
Group	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
Loop	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
Multi-state Input	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
Multi-state Output	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
Notification Class	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
Program	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Description	Program_Change
Schedule	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Description,	Description,
				Weekly_Schedule	Weekly_Schedule,
					Effective_Period

6. Data Link Layer Option

- | | | |
|--|--|--|
| <input type="checkbox"/> ISO 8802-3, 10BASE5 | <input type="checkbox"/> ARCNET, coax star | <input type="checkbox"/> MS/TP master, baud rate(s): |
| <input type="checkbox"/> ISO 8802-3, 10BASE2 | <input type="checkbox"/> ARCNET, coax bus | <input checked="" type="checkbox"/> MS/TP slave: 9600, 19.2k, 38.4k, 76.8k bps |
| <input type="checkbox"/> ISO 8802-3, 10BASET | <input type="checkbox"/> ARCNET, twisted pair star | <input type="checkbox"/> Point-To-Point, EIA 232, baud rate(s): |
| <input type="checkbox"/> ISO 8802-3, Fiber | <input type="checkbox"/> ARCNET, twisted pair bus | <input type="checkbox"/> Point-To-Point, modem, baud rate(s): |
| | <input type="checkbox"/> ARCNET, fiber star | <input type="checkbox"/> LonTalk, medium: |
| <input type="checkbox"/> Other | | |

7. Character Sets Supported

Indicating support for multiple character sets does not imply that they can all be supported simultaneously.

ANSI X3.4
 IBM/Microsoft DBCS
 JIS C 6226
 ISO 10646 (ICS-4)
 ISO 10646 (UCS2)
 ISO 8859-1

-

8. Special Functionality

Segmented Requests Supported Yes No Window Size:
Segmented Responses Supported Yes No Window Size:

-

9. Router

Describe the supported routing capabilities: n/a

-

10. Property Range Restrictions:

Description and Location: 20 octets, including initial character set octet

Conclusion

BACnet is a communication protocol for building automation systems that is an industry standard developed by ASHRAE and adopted by ANSI. It addresses all aspects of the various systems that are applied in buildings today and looks forward into the future. This includes the operator workstations all the way down to the unitary controllers and smart sensors. BACnet provides flexibility in system design and seamless integration and interfacing to building components from multiple manufacturers. Finally, it gives a user options when procuring systems and system components now and into the future.

This flexibility requires that a system is carefully and thoroughly specified to ensure a properly operating system.

Specification Development Step:	Content of specification section based on:
1. Write sequence of operation for system.	Mechanical and electrical equipment to be monitored and controlled.
2. List physical connections to system.	Sequence of operation and other information to be monitored.
3. Optional step: Specify properties of all BACnet object types.	Need to use properties beyond those properties required for each object type.
4. Optional step: Specify LAN types to be used.	Decision to use computer network for LAN backbone or existing BACnet LANs installed.
5. Specify general system approach for project.	Options include:

	<p>One vendor. One vendor with some interfaces. Multiple vendors for components. Integration of all building systems. New BACnet system with gateway to existing system.</p>
6. Systems look and feel.	Features available from one or various vendors of BACnet equipment.
7. Optional step: Specify conformance class or specific services for each product.	If multiple vendors used, conformance class should be specified for compatibility.
8. Optional step: Specify objects that are to be passed between devices.	If interfaces between different vendors are used or if gateways are specified.
9. Performance and execution clauses.	Standard clauses, but add definition of responsibilities if interfaces, multiple vendors or gateways are used.
10. Submittal Data	Require standard performance data, but add Protocol Implementation Conformance Statement (PICS).

Table: Steps in specifying a BACnet based control system