



REAL TIME AUTOMATION



BACnet

Building Connectivity



BACnet

Building Connectivity

Introduction

In June of 1987, a group of HVAC and Building Automation professionals met in Nashville to develop a standard, open protocol specifically for the building automation industry. That effort produced BACnet, an alternative to the proprietary and closed protocols offered by large vendors in the industry.

With its open architecture and ability to control and monitor any building automation process, BACnet meets the needs of users, integrators, and equipment vendors. Its popularity has steadily grown over the years such that it is now one of the leading, if not the leading, networking technology in Building Automation.

BACnet's success is the result of a series of critical design decisions:

- ◆ **Use an Object-oriented approach to standardize the representation of processes and data within a device.**
- ◆ **Provide standard services to access the data within a device.**
- ◆ **Provide more than one physical interface to accommodate small, medium, large, and huge systems.**

These design decisions created a system where devices from various manufacturers could interoperate with little or no additional integration. A BACnet system could offer not only significant cost savings over proprietary systems, but also tremendous savings in resources, installation costs, maintenance costs, and energy savings.

These benefits can best be illustrated using a typical HVAC application. Present are lighting, heating/cooling, and some form of security/safety applications. Each system includes an automation component. The lighting system includes motion sensors to save energy when no one is present and a timer to make sure lights are off after hours. The heating and cooling system contains a programmable thermostat. Finally, the security/safety system has a timer, an access pad, a thermo sensor, and a smoke detector. Each component functions well and all show considerable improvements in efficiencies over traditional dumb switches and systems. So what's lacking and what do we have to gain by integrating and completely automating with BACnet?

The first and obvious benefit is the savings in physical resources. With the price of copper and other conductive metals on the rise, the advantage of having three separate systems wired into one is a significant savings. In a large office building, the savings can amount to millions of dollars in savings. Sharing sensors also offers a considerable savings in physical material and increased efficiency of the different systems.

Consider the idiosyncrasies of office lighting being turned off while you sit quietly as your desk or lights that never turn off because the ventilation system keeps the leaves of an office plant rustling just enough to trick the motion sensor. These small annoyances are endured because there are still significant savings, but why can't the system be improved?

With BACnet it can. In these applications, motion detectors can be used to more efficiently measure human presence and activity. Infrared body sensors would be far more effective, but that cost is usually not justified in a lighting application unless it can be shared with security, heating, cooling and other systems over BACnet. Not only are unnecessary sensors eliminated, but the system itself is far more effective and more user-friendly.

There are countless other similar examples across different building systems. As proof, look no further than how we experience the world each day. We do it with nine senses: Sight, Hearing, Taste, Smell, Touch, Balance and acceleration, Temperature, Kinesthetic sense, and Pain. Every decision we make is directed by one or more of the senses. With BACnet we give our applications similar abilities by sharing sensors and data that is used to make better decisions.



Overview

BACnet (Building Automation and Control Network) is a communications protocol that defines the services used to communicate between building automation end-devices and building control systems. The BACnet protocol specification, like all protocol specifications, defines both how data is represented on the network and the services that are used to move data from one BACnet node to another. It also includes messages that identify data and network nodes such as Who-Is, I-Am, Who-Has, and I-Have.

The BACnet protocol is a completely non-proprietary open communication software standard. There are no proprietary chip sets or special electronics required to implement it. The standard itself is now controlled by the ANSI (American National Standards Institute) and ASHRAE (American Society of Heating Refrigeration and Air-Conditioning Engineers) Standard 135-1995. The BACnet Committee developed the standard with the intention of creating a technically sound and truly non-proprietary standard. They also kept in mind ease of use and did their best to eliminate many barriers to implementation. In June of 1995, after years of industry input, ASHRAE adopted BACnet as a standard for the Building Automation industry.

In BACnet's early days, Ethernet was not used in building and industrial applications like it is today. At that time, there were any number of data link/physical layers used in these applications. To meet the needs of the widest possible set of applications, BACnet was designed to support several different physical and link layers including:

- ◆ **PTP (point-to-point)**

PTP is unique to BACnet and provides for internetworked communications over modems and voice grade phone lines. PTP accommodates modern modem protocols (V.32bis and V.42) and also supports direct cable connections using the EIA-232 signaling standard. Speed is limited to from 9.6 kbit/s to 56.0 kbit/s.

- ◆ **MS/TP (master slave/token passing)**

MS/TP is also unique to BACnet and is implemented using the EIA-485 signaling standard. This is a shielded twisted-pair (STP) LAN operating at speeds from 9.6 kbit/s to 76.0 kbit/s. This LAN type is low cost and particularly suitable for unitary controller communications.

- ◆ **ARCNET**

(ANSI/ATA 878.1) ARCNET® is a token bus standard, and devices typically support it using single-source chips that handle network communications. ARCNET can run on a variety of media at different speeds from 150 kbit/s on EIA-485 (STP) up to 7.5 Mbit/s over coaxial cable, STP, or fiber optics. Typically, ARCNET runs at 2.5 Mbit/s over twisted pair.

- ◆ **LONtalk**

LONtalk is a proprietary technology developed by the Echelon Corporation and is the only LAN type that requires special development tools and a proprietary chip set to implement.

Today, the majority of users prefer to use Ethernet and the number of applications for many of these other physical layers is decreasing.

Like the Control and Information Protocol (CIP) used in industrial application, BACnet uses objects to represent data on a network. These objects are defined by the BACnet specification and have both required and optional data. Objects defined by the specification include Analog Input, Analog Output, Analog Value, Binary Input, Binary Output, Binary Value, Multi-State Input, Multi-State Output, Calendar, Event-Enrollment, File, Notification-Class, Group, Loop, Program, Schedule, Command, and Device.

BACnet is an unconnected, peer network. Any device can send service requests to any other device. Unlike connected protocols where devices have ongoing data transfers, communication in BACnet is unscheduled without any time critical operations.

BACnet is a certifiable standard, though certification is not required. The BACnet Testing Laboratory (BTL) provides services to validate the operation of devices against the specification. Prices vary with the device complexity, but it costs several thousand dollars to certify a BACnet device.



Object-Oriented Technology

In defining the format for BACnet communications, the Standards Committee chose a flexible, object-oriented approach. All data in a BACnet system is represented in terms of "objects," "properties" and "services." This standard method of representing data and actions is what enables BACnet devices from different manufacturers to interoperate. Understanding this object-oriented approach and its terms is essential to understanding BACnet.

Objects

All information in a BACnet system is represented in terms of objects. An object might represent information about a physical input or output, or it may represent a logical grouping of points that perform some function, such as a setpoint. Every object has an identifier (such as AI-1) that allows the BACnet system to identify it. In this regard, an object is much like what is now commonly known as a "data point" in the HVAC community. Where an object differs from a data point is that a data point would typically have a single value associated with it, whereas an object consists of a number of prescribed properties, only one of which is the present value. It is only through its properties that an object is monitored and controlled.

To help clarify this difference, compare the room temperature as a data point to an analog input (AI) object that reports room temperature in a BACnet system. Both are associated with the temperature read from a physical input. When you reference the data point, the only thing it indicates is the room temperature, 72 degrees Fahrenheit. The AI object also reports the room temperature as 72. The key difference is that 72 is the Present-value property of the AI-1 object. Other properties of the object convey more information: the Units property tells the system that the value is in Fahrenheit, the Device-type property that the sensor is a 10k ohm thermistor, and the Description property that it is the conference room temperature. An AI object is much more robust and comprehensive than a simple data point.

Properties

As indicated in the discussion of objects, objects are monitored and controlled only through their properties. BACnet specifies 123 properties of objects. Three properties, Object-identifier, Object-name, and Object-type, must be present in every object. BACnet also may require that certain objects support specific additional properties. The type of object and the type of device in which that object resides determine which properties are present.

BACnet objects have two classes of properties, required properties that must exist in the device and optional properties that exist at the discretion of the device manufacturer. In the temperature example, the present value of the AI object is a required object. The Description property is an optional property that the device manufacturer may not support.

BACnet, like most other protocols, limits access to some properties. The BACnet specification limits define some properties as Read-only and others as Read-write. The device manufacturer can also limit the access to some properties by classifying them as Read-only.

Services

A service is the mechanism that a building automation system uses to access a property or request an action from a BACnet Object. Services are how one BACnet device gets information from another device, commands a device to perform certain actions (through its objects and properties, of course), or communicates events to other objects. The only required service that must be supported by all devices is the Read-property service. There are a total of 32 standard services.

As a system developer or user, you don't need to be concerned with the execution or processing of service requests, which will be transparent and automatic. As a system design engineer for a building automation system you will need to know what objects and services are supported by which devices. This information is found in the device's protocol implementation conformance statement (PICS).



Conformance Classes and the Device PICS

Because not all devices need to have the same level of functionality, BACnet defines conformance classes that categorize the capabilities and functionality of devices. All devices of a certain conformance class will have a minimum set of required features, all in the form of objects and services. Other features can be optional. BACnet insists that this information is made public in a protocol implementation conformance statement (PICS), a list of features that the device supports. The PICS lists what objects are present in the device and whether the device initiates a service request (asks or commands) or executes the request (responds or acts). The PICS also provides you with the conformance class of the device. By comparing a device's PICS with project requirements or with another vendor's PICS, you can determine how well a BACnet product "fits" a given application.