

This article was published in Foundations, A BACnet International Publication, June 2014. © BACnet International 2014 – further editorial use of articles in Foundations is encouraged. Please send a copy to the BACnet International office at info@bacnetinternational.org.

BACnet Developers Q & A

By Steve Karg, Member ASHRAE

Over the years while being involved in the BACnet committee and developing BACnet products, I have fielded questions about BACnet product development. Some of those questions are answered by the official BACnet Testing Laboratories "Implementation Guidelines". However, some questions are beyond the general scope of that document.

Question: Is there some sort of rule or polling technique for MSTP that I am missing? I have a device that will let me poll a couple of objects at a time without much problem, but when I poll more than a few, I start getting errors from the device. I have contacted the vendor and sent them the same Wireshark captures, but wanted to check with you also, just to make sure I'm not breaking any MSTP rules.

Answer: There aren't any polling rules for MS/TP, but there are some facts that may help with polling techniques:

- 1) each MS/TP segment is only capable of one request at a time (it is a shared bus).
- 2) most MS/TP devices can only handle one request at a time (only have a single incoming or outgoing buffer).
- 3) the Token passing nature of the bus, along with the bus speed (baud rate) makes this datalink a lot slower and bandwidth constrained compared to BACnet/IP. The MS/TP baud can be as slow as 9600 bps, and there can be as many as 128 devices passing Tokens in each segment, so it can take awhile to be able to reply to multiple sequential requests.
- 4) some routers make it easier to poll MS/TP by allowing a large Max_Info_Frames values (number of frames that can be sent when the router has the Token). A router with Max_Info_Frames of 100, for example, can take 100 confirmed requests and queue them and execute them before having to pass the Token.

As for the specific device you reference and after reviewing your Wireshark capture, it seems that #2 (number of requests that a device can handle) is hitting its limit (but is certainly more than one!), and you could then back off on the number of outstanding (simultaneous) requests to that device, or wait until the resources (i.e. Complex-ACK is returned) free up.

Each MS/TP device is different since a vendor can choose how many requests each product can support at one time, so it is unknown how many request any device can support, and there isn't a property in the device that can be read to tell a client what that number could be. MS/TP tends to be used in smaller, lower resource devices since it only needs a microcontroller large

enough to encode and decode BACnet data, and an RS-485 transceiver. Having more than one incoming buffer for the PDU (well, I suppose it could really be how many transmit buffers it uses to hold the responses), is the exception, not the norm.

A single outstanding request per device is a good baseline for MS/TP devices, which means waiting for the response before requesting again from the same device. At the MS/TP datalink level, the request is encoded into a Data-Expecting-Reply frame, and the device has up to 250ms to respond, or send a Reply-Postponed frame to the router (or peer MS/TP device). The Data-Expecting-Reply frame will keep all other devices on the bus from communicating during the time it is waiting for a response, so it is pointless to ask for other devices on that MS/TP segment (except if the router has an incoming request queue and a big enough Max_Info_Frames setting).

If the MS/TP device supports ReadPropertyMultiple (read the Protocol_Services_Supported property in the Device object to verify), then the client can request multiple properties from multiple objects in a single request, up to the limit of the APDU size in the device and the datalink layer. The Max_APDU_Length_Accepted property can guide the client, but the response to the request will likely be larger than the request. The maximum APDU size for MS/TP is currently 480 bytes, but could be as small as 50 bytes. Some devices could support larger APDU sizes by using segmented APDUs, as indicated by their Segmentation_Supported and Max_Segments_Accepted properties.

Question: There is a need for me to have an object within another object. I know that the device object already does it, using the object-list property to list all objects. Are there any BACnet properties that do hierarchy-like object representation?

Answer: A Structured View object, which was added to the BACnet standard in 2006, could present objects in a hierarchy. "The Structured View object type defines a standardized object that provides a container to hold references to subordinate objects, which may include other Structured View objects, thereby allowing multilevel hierarchies to be created. The hierarchies are intended to convey a structure or organization such as a geographical distribution or application organization. Subordinate objects may reside in the same device as the Structured View object or in other devices on the network." Note that the Structured View object is currently being extended to support Application Interfaces. See the Structured View object addenda at:

<http://www.bacnet.org/Addenda/Add-2004-135d.pdf>

Question: If I have a single BACnet client, then I obviously have 255 invoke IDs I can use. Does that mean I have to stay within that range (255) no matter what, even if I have more than 200 devices on network, or can I have 255 invoke IDs per device to which I am sending a request?

Answer: Your client can have up to 255 invoke IDs per device it is sending the request to. It is a matter of being able to match up response with request.

About the Author

Steve Karg is a Senior Engineer at WattStopper, in Birmingham, Alabama. He has been an active member of ASHRAE SSPC 135 (BACnet) since 2001, and convenes their Lighting Applications working group. He wrote an open source BACnet Protocol Stack hosted on SourceForge.net, and continues to help maintain the BACnet decoder in Wireshark.