BSR/ASHRAE Addendum *cz* to
ANSI/ASHRAE Standard 135-2024

First Public Review Draft

Proposed Addendum *cz* to Standard 135-2024, BACnet® - A Data Communication Protocol for Building Automation and Control Networks

First Publication Public Review (November 2026)
(Draft shows Proposed Changes to Current Standard)

This draft has been recommended for public review by the responsible project committee. To submit a comment on this proposed standard, go to the ASHRAE website at www.ashrae.org/standards-research--technology/public-review-drafts and access the online comment database. The draft is subject to modification until it is approved for publication by the Board of Directors and ANSI. Until this time, the current edition of the standard (as modified by any published addenda on the ASHRAE website) remains in effect. The current edition of any standard may be purchased from the ASHRAE Online Store at www.ashrae.org/bookstore or by calling 404-636-8400 or 1-800-727-4723 (for orders in the U.S. or Canada).

This standard is under continuous maintenance. To propose a change to the current standard, use the change submittal form available on the ASHRAE website, www.ashrae.org.

The appearance of any technical data or editorial material in this public review document does not constitute endorsement, warranty, or guaranty by ASHARE of any product, service, process, procedure, or design, and ASHRAE expressly disclaims such.

© 2026 ASHRAE. This draft is covered under ASHRAE copyright. Permission to reproduce or redistribute all or any part of this document must be obtained from the ASHRAE Manager of Standards, 180 Technology Parkway NW, Peachtree Corners, GA 30092. Phone: 404-636-8400, Ext. 1125. Fax: 404-321-5478. E-mail: standards.section@ashrae.org.

ASHRAE, 180 Technology Parkway NW, Peachtree Corners, GA 30092

(This foreword is not part of this standard. It is merely informative and does not contain requirements necessary for conformance to the standard. It has not been processed according to the ANSI requirements for a standard and may contain material that has not been subject to public review or a consensus process. Unresolved objectors on informative material are not offered the right to appeal at ASHRAE or ANSI.)

FOREWORD
This addendum provides a solution in the form of two new objects called Network Read and Network Write Objects.

Note: This addendum makes proposed changes to the current standard. These changes are indicated in the text by underlining (for additions) and strikethrough (for deletions) except where the reviewer instructions specifically describe some other means of showing the changes. Only these changes to the current standard are open for review and comment at this time. Additional material is provided for context only and is not open for comment except as it relates to the proposed changes.

Addendum cz to Standard 135-2024

Rationale

There is no standardized mechanism to expose and facilitate the configuration of device-object-property references and describe behaviors that are needed by BACnet client devices that wish to read or write to the referenced value. The lack of a standard mechanism assures that proprietary means most likely must be used to configure BACnet client devices.

Background:

BACnet devices, such as programmable controllers with custom logic capabilities or application-specific controllers that can act as BACnet clients, must have some mechanism to configure or set up the device when the logic needs to reference values in object properties in other devices. As an example, a controller may desire to get its setpoint from a local object property, or perhaps "pull" that setpoint value from an object property in another device. Control loop logic that produces an output, may wish to poke that output to say an AO or AV Present_Value in another device. There are many examples we could imagine when one device might want to be able to read or write to properties of objects in other devices.

The issue is that BACnet does not presently provide a standard way:

- to define such references (other than in certain standard object types e.g. LOOP),
- to examine a given device and find out what its references currently reference,
- to configure those references or show how they have been configured,
- how the reading/writing of those references should behave,
- even when the referenced object to be read or written exists in the same device as the client, there is no mechanism in BACnet to organize all such references used by the device's client processes for easy configuration or examination.

Since these are things that are common in control systems, many devices use proprietary means to accomplish them. In some cases, there may be properties of non-standard objects that have BACnetDeviceObjectPropertyReference datatypes that can be written using WriteProperty, or there may be other means such as proprietary datatypes, services, files etc.

Even if a client knows where particular device-object-property references are, there is no standard mechanism for describing how a given client within a device is managing the fetching/reading of values from a referenced object property, or poking/writing values to a referenced property.

©2025 ASHRAE Inc., All rights reserved

Our proposition is that such *network integration* is desirable for BACnet if it can be standardized with respect to how the configuration and behavior can be determined through examination, and how it can be altered.

Solution Summary:

This proposal provides a solution in the form of two new objects called Network Read and Network Write Objects. Devices that act as clients may expose bindings through one or more Network Read/Write objects that each define the target device-object-property for a named value. A single Network Read/Write object describes a single association between the object and its referenced value. A Network Read/Write object centralizes the reference and the most recent value so that multiple consumers of these values may reference the Network Read/Write object, rather than fetching/poking the reference themselves.

Note version 21 changes highlighted in yellow.

Solution Details:

Network Read/Write objects are modeled as a tuple consisting of:

- a name,
- a BACnetDeviceObjectPropertyReference,
- a binding type that also implies a dataflow direction,
- and several other components that vary according to binding type.

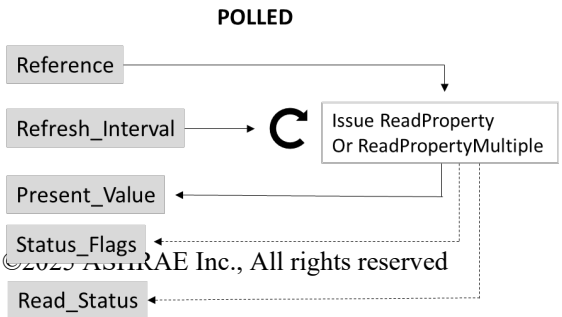
Devices would likely use multiple Network Read/Write objects, one for each configurable parameter or configurable output. Configuration tools can easily discover these bindings by examining the Object_List property of the Device object, looking for Object_Identifiers of type Network Read or Network Write. In the Network Read/Write object, individual properties expose each element of the binding tuple. These bindings may or may not be configurable through these objects. A new BIBB defines the ability of a device to have configurable bindings.

12.X Network Read Object Type

The Network Read object type defines a standardized object whose properties represent the externally visible characteristics of a named value, referenced by objects or functions within a BACnet device for reading. A BACnet device can use a group of Network Read objects to make the specific values visible to other BACnet devices. This can optionally facilitate the configuration of these bindings by other BACnet devices. Network Read objects centralize the location of references to (possibly non-local) properties whose values are needed by the device's applications and describes the expected behavior of reading the referenced value.

Some Network Read objects are not configurable through BACnet and might only be configurable through a mechanism local to the device or the configuration might be fixed and not changeable. Configurable Network Read objects allow the configuration to be changed by writing to the properties of the Network Read object. The mechanisms by which the named values have been established, and the internal control loops or programs that depend on these bindings, might not be visible to BACnet clients who read the Network Read object's properties. The correspondence between a given Network Read and the internal program(s) that depend upon it shall be a local matter.

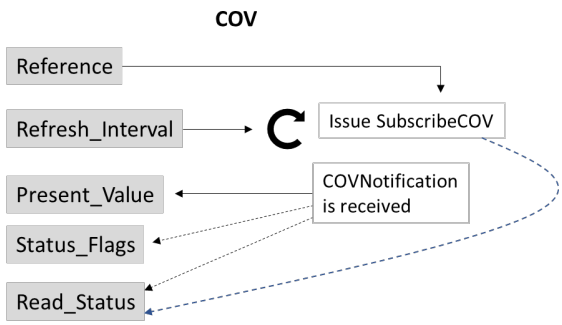
There are three kinds of behavior that can be assigned to a Network Read. These behaviors are called *binding types*. There are three types of read bindings corresponding to Binding_Type values POLLED, COV, and COVFALLBACK.



When Binding_Type is POLLED, the Network Read object polls the value as often as needed using ReadProperty or ReadPropertyMultiple services. The frequency of polling shall be specified in the Refresh_Interval property.

©2023 ASHRAE Inc., All rights reserved

When Binding_Type is COV, the Network Read object attempts to establish a COV subscription with the device referenced by the Reference property. In the case of a failed subscription attempt, the Network Read object shall retry the subscription at a frequency specified by the Refresh_Interval property. If the subscription succeeds, then whenever a COV notification for this value is received the new value shall be reflected in the Present_Value property. While a COV subscription is in effect, the Network Read object shall attempt to resubscribe every Refresh_Interval seconds.



When Binding_Type is COVFALLBACK, the behavior is the same as COV, except that in the case of a failed subscription attempt, the Network Read object shall fall back to using POLLED. While polling, every Refresh_Interval seconds the Network Read object shall retry the COV subscription. If the subscription succeeds, then polling will stop and the object shall revert to COV behavior. Subsequent resubscriptions that fail shall continue to fall back to POLLED operation.

©2025 ASHRAE Inc., All rights reserved

Table 12-X1. Properties of the Network Read Object

Property Identifier	Property Datatype	Conformance Code
Object_Identifier	BACnetObjectIdentifier	R
Object_Name	CharacterString	R
Object_Type	BACnetObjectType	R
Description	CharacterString	O
Used_By	BACnetARRAY[N] of BACnetObjectIdentifier	O
Present_Value	ABSTRACT-SYNTAX.&Type	R[1]
Status_Flags	BACnetStatusFlags	R
Reliability	BACnetReliability	O
Out_Of_Service	Boolean	R
Reference	BACnetDeviceObjectPropertyReference	R
Referenced_Status_Flags	BACnetStatusFlags	O
Read_Status	BACnetNetworkResult	R
Binding_Type	BACnetReadBindingType	R
Refresh_Interval	Unsigned	R
COV_Resubscription_Interval	Unsigned	O[2]
Client_COV_Increment	BACnetClientCOV	O
Property_List	BACnetARRAY[N] of BACnetPropertyIdentifier	R
Tags	BACnetARRAY[N] of BACnetNameValue	O
Profile_Location	CharacterString	O
Profile_Name	CharacterString	O

[1] the device may limit its support to only primitive application datatypes or only certain ones as a local matter

[2] if BINDING_TYPE can have the type COV or COVFALLBACK then this property shall be required

12.X.1 Object_Identifier

This read-only property, of type BACnetObjectIdentifier, is a numeric code that is used to identify the object. It shall be unique within the BACnet Device that maintains it.

12.X.2 Object_Name

This property, of type CharacterString, shall represent a name for the object that is unique within the BACnet Device that maintains it. The minimum length of the string shall be one character. The set of characters used in the Object_Name shall be restricted to printable characters.

12.X.3 Object_Type

This read-only property, of type BACnetObjectType, indicates membership in a particular object type class. The value of this property shall be NETWORK_READ.

12.X.4 Description

This property, of type CharacterString, is a string of printable characters whose content is not restricted.

12.X.5 Used_By

This read-only property, of type BACnetARRAY of BACnetObjectIdentifier, is a list of objects that make use of the value defined by this Network Read object.

©2025 ASHRAE Inc., All rights reserved

12.X.6 Present_Value

This property, of type ABSTRACT-SYNTAX.&Type, reflects the value of the property that is referenced by the Reference property. The Present_Value property shall be writable when Out_Of_Service is TRUE. The datatype(s) supported by this property is a local matter.

12.X.7 Status_Flags

This read-only property, of type BACnetStatusFlags, represents four Boolean flags that indicate the general "health" of a Network Read object. Two of the flags are associated with the values of other properties of this object. A more detailed status could be determined by reading the properties that are linked to these flags. The relationship between individual flags is not defined by the protocol. The four flags are

{IN_ALARM, FAULT, OVERRIDDEN, OUT_OF_SERVICE}

where:

IN_ALARM	Logical FALSE (0).
FAULT	Logical TRUE (1) if the Reliability property is present and does not have a value of NO_FAULT_DETECTED, otherwise logical FALSE (0).
OVERRIDDEN	Logical FALSE (0).
OUT_OF_SERVICE	Logical TRUE (1) if the Out_Of_Service property has a value of TRUE, otherwise logical FALSE (0).

12.X.8 Reliability

This property, of type BACnetReliability, provides an indication of whether the Present_Value of the object is "reliable" as far as the BACnet device can determine and, if not, why. The Reliability property, if present, shall be writable when Out_Of_Service is TRUE.

12.X.9 Out_Of_Service

This property, of type Boolean, is an indication whether (TRUE) or not (FALSE) the Network Read that the object represents is not in service.

When Out_Of_Service is TRUE:

(a) the Present_Value property is decoupled from the property that is referenced by the Reference property and will not track changes to the referenced property's value;
(b) the Reliability property, if present, and the corresponding state of the FAULT flag of the Status_Flags property shall be decoupled from the referenced property;
(c) the Present_Value property and the Reliability property, if present and capable of taking on values other than NO_FAULT_DETECTED, shall be writable to allow simulating specific conditions or for testing purposes;
(d) other functions that depend on the state of the Present_Value or Reliability properties shall respond to changes made to these properties as if those changes had occurred in the referenced property.

When Out_Of_Service changes from TRUE to FALSE the Present_Value shall be refreshed by re-reading the property referenced by the Reference property or re-subscribing for COV.

12.X.10 Reference

This property, of type BACnetDeviceObjectPropertyReference, describes the source of the value that the Network Read object represents. The device's client processes that make use of a given binding shall use it for reading the referenced property according to the Binding_Type.

Network Read object instances that are not initialized, or that are intentionally disabled, shall contain the value 4194303 as the instance number of the object-identifier field of the Reference property value.

©2025 ASHRAE Inc., All rights reserved

12.X.11 Referenced_Status_Flags

This read-only property, of type BACnetStatusFlags, reflects the value of the Status_Flags property of the object that is referenced by the Reference property. If the Reference is disabled, or the referenced object does not contain a Status_Flags property, then this property shall return the value {FALSE,FALSE,FALSE,FALSE} (Normal, No Fault Detected, Not Overridden, In Service).

12.X.12 Read_Status

This read-only property, of type BACnetNetworkResult, conveys to those applications that use the reference the most recent attempt to read from the reference. If the read was successful then Read_Status shall have a value error-class of 'OBJECT' and error-code of 'SUCCESS'. If the read failed then the resulting error class and error code shall be used. Initially this value shall have timestamp of unspecified and result of SUCCESS.

12.X.13 Binding_Type

This property of type BACnetBindingType indicates what type of Network Read this object represents. When Binding_Type is:

POLLED	the Network Read object shall poll the value as often as needed using ReadProperty or ReadPropertyMultiple services.
COV	the Network Read object shall attempt to establish a COV subscription with the device referenced by the Reference property.
COVFALLBACK	the behavior shall be the same as COV, except that in the case of a failed subscription attempt, the Network Read object shall fall back to using POLLED.

12.X.14 Refresh_Interval

This property of type Unsigned specifies the interval in seconds between periodic reading of the referenced object property, when Binding_Type is POLLED. If this value is 0, the frequency of periodic reading is a local matter.

12.X.15 COV_Resubscription_Interval

If the Network Read object is acquiring data from a remote device by COV subscription, this property, of type Unsigned, specifies the number of seconds between COV resubscriptions, provided that COV subscription is in effect. COV subscription requests shall specify twice this lifetime for the subscription and shall specify the issuance of confirmed notifications.

12.X.16 Client_COV_Increment

This property, of type BACnetClientCOV, specifies the increment to be used in determining that a change of value has occurred when the Network Read Object is acquiring COV data.. If the referenced object property supports COV reporting according to Clause 13.1, this property may have the value Null; in this case change of value is determined by the criteria of Clause 13.1.

12.X.17 Property_List

This read-only property, of type BACnetARRAY of property identifiers, contains one BACnetPropertyIdentifier for each property that exists within the object. The Object_Name, Object_Type, Object_Identifier, and Property_List properties are not included in the list.

12.X.18 Tags

This property, of type BACnetARRAY of BACnetNameValue, is a collection of tags for the object. See Clause Y.1.4 for restrictions on the string values used for the names of these tag and for a description of tagging and the mechanism by which tags are defined.

Each entry in the array is a BACnetNameValue construct which consists of the tag name and an optional value. If the tag is defined to be a "semantic tag" then it has no value, and the "value" field of the BACnetNameValue shall be absent.

©2025 ASHRAE Inc., All rights reserved

While some tags may be known in advance when a device is manufactured, it is recommended that implementations consider that this kind of information might not be known until a device is deployed and to provide a means of configuration or writability of this property.

12.X.19 Profile_Location

This property, of type CharacterString, is the URI of the location of an xdd file (See Clause X.2) containing the definition of the CSML type specified by the Profile_Name property and possible other information (See Annex X). The URI is restricted to using only the "http", "https", and "bacnet" URI schemes. See Clause Q.8 for the definition of the "bacnet" URI scheme.

If a Profile_Location value is not provided for a particular object, then the client shall use the Profile_Location of the Device object, if provided, to find the definition of the Profile_Name.

12.X.20 Profile_Name

This optional property, of type CharacterString, is the name of an object profile to which this object conforms. To ensure uniqueness, a profile name must begin with a vendor identifier code (see Clause 23) in base-10 integer format, followed by a dash. All subsequent characters are administered by the organization registered with that vendor identifier code. The vendor identifier code that prefixes the profile name shall indicate the organization that publishes and maintains the profile document named by the remainder of the profile name. This vendor identifier need not have any relationship to the vendor identifier of the device within which the object resides.

A profile defines a set of additional properties, behavior, and/or requirements for this object beyond those specified here. This standard defines only the format of the names of profiles. The definition of the profiles themselves is outside the scope of this standard.

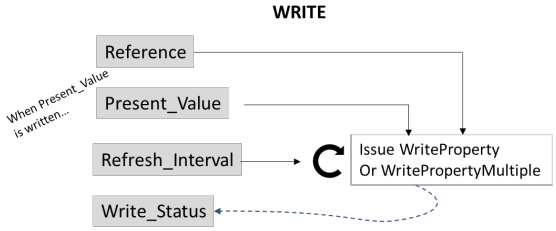
©2025 ASHRAE Inc., All rights reserved

12.Y Network Write Object Type

The Network Write object type defines a standardized object whose properties represent the externally visible characteristics of a named value, referenced by objects or functions within a BACnet device that when written, propagates that value to the referenced object. A BACnet device can use a group of Network Write objects to make the specific values visible to other BACnet devices, and to optionally facilitate the configuration of these bindings by other BACnet devices. Network Write objects centralize the location of references to (possibly non-local) device-object-properties whose values need to be changed/written by the device's applications and describes the expected behavior of writing to the referenced value.

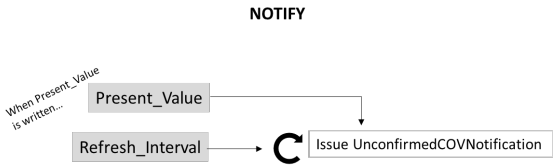
Some Network Write objects are not configurable through BACnet and may only be configurable through a mechanism local to the device or the configuration may be fixed and not changeable. *Configurable* Network Write objects allow the configuration to be changed by writing to the properties of the Network Write object. The mechanisms by which the named values have been established, and the internal control loops or programs that depend on these bindings, might not be visible to BACnet clients who read the Network Write object's properties. The correspondence between a given Network Write and the internal program(s) that depend upon it shall be a local matter.

There are three kinds of behavior that can be assigned to a Network Write. These behaviors are called *binding types*. There are three types of write bindings corresponding to Binding_Type values WRITE, NOTIFY and GROUP.

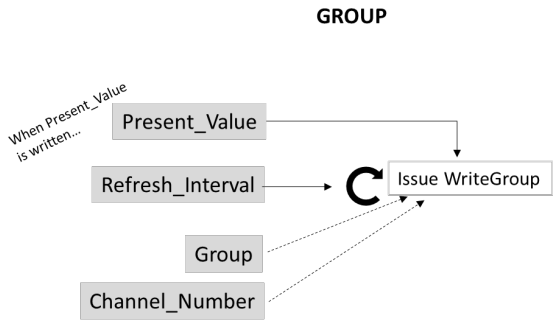


When Binding_Type is WRITE, and Present_Value of the Network Write object is written-to, the Network Write uses WriteProperty or WritePropertyMultiple to pass this value on to the Reference object. The Priority_For_Writing property shall indicate the priority to be used when writing.

When Binding_Type is NOTIFY, and Present_Value is written, the Network Write object uses UnconfirmedCOVNotification that shall be broadcast only on the local network. For this binding type, the Reference object is not used.



If COV_Increment is zero or not provided, then the Network Write object only sends a notification when the value changes. Otherwise COV_Increment is the amount that the value must change by since the previous write, before a new UnconfirmedCOVNotification with 'Subscriber Process Identifier' value of 0 is initiated.



When Binding_Type is GROUP, the Network Write object uses WriteGroup to write to the group and channel specified by the Group and Channel_Number properties.

In all of these write cases, writing shall be triggered whenever the Used_By object(s) or process(es) that change Present_Value perform the write to Present_Value. Whether or not the Network Write object supports writes to Present_Value using BACnet services such as WriteProperty is a local matter. The Refresh_Interval property shall indicate the time in seconds between rewrites of the value to the referenced property after Present_Value has changed. If Refresh_Interval is zero, then the client shall only update the

©2025 ASHRAE Inc., All rights reserved

referenced property once each time Present_Value changes. Otherwise, every Refresh_Interval seconds the value shall be rewritten by the Network Write object.

Table 12-Y1. Properties of the Network Write Object

Property Identifier	Property Datatype	Conformance Code
Object_Identifier	BACnetObjectIdentifier	R
Object_Name	CharacterString	R
Object_Type	BACnetObjectType	R
Description	CharacterString	O
Used_By	BACnetARRAY[N] of BACnetObjectIdentifier	O
Present_Value	ABSTRACT-SYNTAX.&Type	R[1]
Status_Flags	BACnetStatusFlags	R
Reliability	BACnetReliability	O
Out_Of_Service	Boolean	R
Reference	BACnetDeviceObjectPropertyReference	R
Write_Status	BACnetNetworkResult	R
Binding_Type	BACnetWriteBindingType	R
Priority_For_Writing	Unsigned(0..16)	O
Refresh_Interval	Unsigned	R
Last_Write	ABSTRACT-SYNTAX.&Type	O
Write_Threshold	REAL	O
Minimum_Time_Between_Writes	Unsigned	O
Group	Unsigned	O[2]
Channel_Number	Unsigned16	O[1]
Property_List	BACnetARRAY[N] of BACnetPropertyIdentifier	R
Tags	BACnetARRAY[N] of BACnetNameValue	O
Profile_Location	CharacterString	O
Profile_Name	CharacterString	O

[1] the device may limit its support to only primitive application datatypes or only certain ones as a local matter

[2] If either of these properties is present then both shall be present

12.Y.1 Object_Identifier

This read-only property, of type BACnetObjectIdentifier, is a numeric code that is used to identify the object. It shall be unique within the BACnet Device that maintains it.

12.Y.2 Object_Name

This property, of type CharacterString, shall represent a name for the object that is unique within the BACnet Device that maintains it. The minimum length of the string shall be one character. The set of characters used in the Object_Name shall be restricted to printable characters.

12.Y.3 Object_Type

This read-only property, of type BACnetObjectType, indicates membership in a particular object type class. The value of this property shall be NETWORK_WRITE.

©2025 ASHRAE Inc., All rights reserved

12.Y.4 Description

This property, of type CharacterString, is a string of printable characters whose content is not restricted.

12.Y.5 Used_By

This read-only property, of type BACnetARRAY of BACnetObjectIdentifier, is a list of objects that make use of and write to the value defined by this Network Write object.

12.Y.6 Present_Value

This property, of type ABSTRACT-SYNTAX.&Type, reflects the value to be written to the property that is referenced by the Reference property. Whether Present_Value can normally be written with BACnet services or only written by internal processes is a local matter. The Present_Value property shall be writable when Out_Of_Service is TRUE. The datatype(s) supported by this property is a local matter.

12.Y.7 Status_Flags

This read-only property, of type BACnetStatusFlags, represents four Boolean flags that indicate the general "health" of a Network Write object. Three of the flags are associated with the values of other properties of this object. A more detailed status could be determined by reading the properties that are linked to these flags. The relationship between individual flags is not defined by the protocol. The four flags are

{IN_ALARM, FAULT, OVERRIDDEN, OUT_OF_SERVICE}

where:

IN_ALARM	Logical FALSE (0).
FAULT	Logical TRUE (1) if the Reliability property is present and does not have a value of NO_FAULT_DETECTED, otherwise logical FALSE (0).
OVERRIDDEN	Logical FALSE (0).
OUT_OF_SERVICE	Logical TRUE (1) if the Out_Of_Service property has a value of TRUE, otherwise logical FALSE (0).

12.Y.8 Reliability

This read-only property, of type BACnetReliability, provides an indication of whether the Present_Value of the object is "reliable" as far as the BACnet device can determine and, if not, why.

12.Y.9 Out_Of_Service

This property, of type Boolean, is an indication whether (TRUE) or not (FALSE) the Network Write that the object represents is not in service.

When Out_Of_Service is TRUE:

(a) the Present_Value property is decoupled from the property that is referenced by the Reference property and will not attempt any writes to the referenced property;
(b) the Reliability property, if present, and the corresponding state of the FAULT flag of the Status_Flags property shall be decoupled from the referenced property;
(c) the Present_Value property and the Reliability property, if present and capable of taking on values other than NO_FAULT_DETECTED, shall be writable to allow simulating specific conditions or for testing purposes;
(d) other functions that depend on the state of the Present_Value or Reliability properties shall respond to changes made to these properties as if those changes had occurred as a result of attempting to write to the referenced property.

When Out_Of_Service changes from TRUE to FALSE the Present_Value shall be refreshed by re-writing the property referenced by the Reference property or reissuing an UnconfirmedCOVNotification or WriteGroup.

©2025 ASHRAE Inc., All rights reserved

12.Y.10 Reference

This property, of type BACnetDeviceObjectPropertyReference, describes the location of the value that the Network Write object represents. The device's client processes that make use of a given binding shall use it for writing to the referenced property according to the Binding_Type.

Network Write object instances that are not initialized, or that are intentionally disabled, shall contain the value 4194303 as the instance number of the object identifier field of the Reference property value.

12.Y.11 Write_Status

This read-only property, of type BACnetNetworkResult, conveys to those applications that use the reference the most recent attempt to write to the reference. If the write was successful then the result field of Write_Status shall have a value 'Error Class' of OBJECT and 'Error Code' of SUCCESS. If the write failed then the resulting error class and error code shall be used. . Initially this value shall have a timestamp of unspecified and result of SUCCESS.

12.Y.12 Binding_Type

This property of type BACnetBindingType indicates what type of Network Write this object represents.

When Binding_Type is:

WRITE and Present_Value of the Network Write object is written-to by BACnet services or an internal process, the Network Write object shall use WriteProperty or WritePropertyMultiple services to pass this value on to the Reference object.

NOTIFY and Present_Value is written-to by BACnet services or an internal process, the Network Write object shall use UnconfirmedCOVNotification service that shall be broadcast only on the local network. For this binding type, the Reference object is not used.

GROUP the Network Write object shall use WriteGroup service to write to the group and channel specified by the Group and Channel_Number properties.

12.Y.13 Priority_For_Writing

This property, of type Unsigned, defines the priority at which the referenced object property is commanded. It corresponds to the 'Priority' parameter of the WriteProperty service. It is an unsigned integer in the range 0-16, with 1 being considered the highest priority and 16 the lowest. See Clause 19. A value of 0 shall mean that no priority is sent with WriteProperty or WritePropertyMultiple services.

12.Y.14 Refresh_Interval

This property of type Unsigned specifies the interval in seconds between periodic writing of the referenced object property. If this value is 0, periodic writing shall not occur.

12.Y.15 Last_Write

This read-only property, of type ABSTRACT-SYNTAX.&Type, represents the last value that was written to the referenced property.

12.Y.16 Write_Threshold

This property, of type Real, represents the amount by which Present_Value and Last_Write must differ before a WRITE, NOTIFY or GROUP is performed when Out_Of_Service is FALSE. If Write_Threshold is not provided, or the datatype of Present_Value changes or is not a numeric value that can be compared with a Real, then writes occur immediately. If Write_Threshold is 0 then the value shall be rewritten every Refresh_Interval.

©2025 ASHRAE Inc., All rights reserved

12.Y.17 Minimum_Time_Between_Writes

This property of type Unsigned shall indicate the minimum time in seconds between writes to the device object property described by the Reference property. When this value is greater than 0, writes to Present_Value from BACnet or internal sources shall not initiate writes to the referenced property more often than Minimum_Time_Between_Writes seconds. During that delay period, any additional writes to Present_Value from BACnet or internal sources shall overwrite the Present_Value with the new value but the write to the referenced property shall remain pending. Writes to the referenced property caused by the Refresh_Interval are independent of this limitation.

12.Y.18 Group

This property of type Unsigned shall indicate which group number shall be sent in WriteGroup services issued when Binding_Type is GROUP.

12.Y.19 Channel_Number

This property of type Unsigned shall indicate which channel number shall be sent in WriteGroup services issued when Binding_Type is GROUP.

12.Y.20 Property_List

This read-only property, of type BACnetARRAY of BACnetPropertyIdentifier, contains one BACnetPropertyIdentifier for each property that exists within the object. The Object_Name, Object_Type, Object_Identifier, and Property_List properties are not included in the list.

12.Y.21 Tags

This property, of type BACnetARRAY of BACnetNameValue, is a collection of tags for the object. See Clause Y.1.4 for restrictions on the string values used for the names of these tag and for a description of tagging and the mechanism by which tags are defined.

Each entry in the array is a BACnetNameValue construct which consists of the tag name and an optional value. If the tag is defined as a "semantic tag" then it has no value, and the "value" field of the BACnetNameValue shall be absent.

While some tags may be known in advance when a device is manufactured, it is recommended that implementations consider that this kind of information might not be known until a device is deployed and provide a means of configuration or writability of this property.

12.Y.22 Profile_Location

This property, of type CharacterString, is the URI of the location of an xdd file (See Clause X.2) containing the definition of the CSML type specified by the Profile_Name property and possible other information (See Annex X). The URI is restricted to using only the "http", "https", and "bacnet" URI schemes. See Clause Q.8 for the definition of the "bacnet" URI scheme.

If a Profile_Location value is not provided for a particular object, then the client shall use the Profile_Location of the Device object, if provided, to find the definition of the Profile_Name.

12.Y.23 Profile_Name

This optional property, of type CharacterString, is the name of an object profile to which this object conforms. To ensure uniqueness, a profile name must begin with a vendor identifier code (see Clause 23) in base-10 integer format, followed by a dash. All subsequent characters are administered by the organization registered with that vendor identifier code. The vendor identifier code that prefixes the profile name shall indicate the organization that publishes and maintains the profile document named by the remainder of the profile name. This vendor identifier need not have any relationship to the vendor identifier of the device within which the object resides.

A profile defines a set of additional properties, behavior, and/or requirements for this object beyond those specified here. This standard defines only the format of the names of profiles. The definition of the profiles themselves is outside the scope of this standard.

©2025 ASHRAE Inc., All rights reserved

[Add to **21**, BACnetObjectType]

```
        Network Read                    (n),
        Network Write                   (n+1),
```

[Add to **21**, BACnetObjectTypesSupported]

```
        Network Read                    (n),
        Network Write                   (n+1),
```

[Add to **21**, BACnetPropertyIdentifier]

```
        -- see interface-name                       (n),
        -- see reference                            (n+1),
        -- see used-by                              (n+2),
        -- see binding-type                         (n+3),
        -- see group                                (n+4),
        -- see last-sent                            (n+5),
        -- see read-status                          (n+6),
        -- see minimum-time-between-writes          (n+7),
        -- see referenced-status-flags              (n+8),
        -- see refresh-interval                     (n+9)
        …

BACnetReadBindingType ::= ENUMERATED {
        polled                          (0),
        cov                             (1),
        covfallback                     (2)
}

BACnetWriteBindingType ::= ENUMERATED {
        write                           (0),
        notify                          (1),
        group                           (2)
}

BACnetNetworkResult ::= SEQUENCE {
timestamp                               [0] BACnetDateTime,
result                                  [1] Error
}
```

©2025 ASHRAE Inc., All rights reserved