ADDENDA

ANSI/ASHRAE Addendum ai to
ANSI/ASHRAE Standard 135-2012

Data Communication Protocol for Building Automation and Control Networks

Approved by ASHRAE on December 30, 2014; and by the American National Standards Institute on December 31, 2014.

This addendum was approved by a Standing Standard Project Committee (SSPC) for which the Standards Committee has established a documented program for regular publication of addenda or revisions, including procedures for timely, documented, consensus action on requests for change to any part of the standard. The change submittal form, instructions, and deadlines may be obtained in electronic form from the ASHRAE website (www.ashrae.org) or in paper form from the Senior Manager of Standards.

The latest edition of an ASHRAE Standard may be purchased on the ASHRAE website (www.ashrae.org) or from ASHRAE Customer Service, 1791 Tullie Circle, NE, Atlanta, GA 30329-2305. E-mail: orders@ashrae.org. Fax: 678-539-2129. Telephone: 404-636-8400 (worldwide), or toll free 1-800-527-4723 (for orders in US and Canada). For reprint permission, go to www.ashrae.org/permissions.

 ISSN 1041-2336

Approved American National Standard
ANSI
www.ansi.org

SPECIAL NOTE

This American National Standard (ANS) is a national voluntary consensus Standard developed under the auspices of ASHRAE. *Consensus* is defined by the American National Standards Institute (ANSI), of which ASHRAE is a member and which has approved this Standard as an ANS, as "substantial agreement reached by directly and materially affected interest categories. This signifies the concurrence of more than a simple majority, but not necessarily unanimity. Consensus requires that all views and objections be considered, and that an effort be made toward their resolution." Compliance with this Standard is voluntary until and unless a legal jurisdiction makes compliance mandatory through legislation.

ASHRAE obtains consensus through participation of its national and international members, associated societies, and public review.

ASHRAE Standards are prepared by a Project Committee appointed specifically for the purpose of writing the Standard. The Project Committee Chair and Vice-Chair must be members of ASHRAE; while other committee members may or may not be ASHRAE members, all must be technically qualified in the subject area of the Standard. Every effort is made to balance the concerned interests on all Project Committees.

The Senior Manager of Standards of ASHRAE should be contacted for

a. interpretation of the contents of this Standard,
b. participation in the next review of the Standard,
c. offering constructive criticism for improving the Standard, or
d. permission to reprint portions of the Standard.

DISCLAIMER

ASHRAE uses its best efforts to promulgate Standards and Guidelines for the benefit of the public in light of available information and accepted industry practices. However, ASHRAE does not guarantee, certify, or assure the safety or performance of any products, components, or systems tested, installed, or operated in accordance with ASHRAE's Standards or Guidelines or that any tests conducted under its Standards or Guidelines will be nonhazardous or free from risk.

ASHRAE INDUSTRIAL ADVERTISING POLICY ON STANDARDS

ASHRAE Standards and Guidelines are established to assist industry and the public by offering a uniform method of testing for rating purposes, by suggesting safe practices in designing and installing equipment, by providing proper definitions of this equipment, and by providing other information that may serve to guide the industry. The creation of ASHRAE Standards and Guidelines is determined by the need for them, and conformance to them is completely voluntary.

In referring to this Standard or Guideline and in marking of equipment and in advertising, no claim shall be made, either stated or implied, that the product has been approved by ASHRAE.

[This foreword and the "rationales" on the following pages are not part of this standard. They are merely informative and do not contain requirements necessary for conformance to the standard.]

FOREWORD

The purpose of this addendum is to present a proposed change for public review. These modifications are the result of change proposals made pursuant to the ASHRAE continuous maintenance procedures and of deliberations within Standing Standard Project Committee 135. The proposed changes are summarized below.

In the following document, language to be added to existing clauses of ANSI/ASHRAE 135-2012 and Addenda is indicated through the use of *italics*, while deletions are indicated by ~~strikethrough~~. Where entirely new subclauses are proposed to be added, plain type is used throughout. Only this new and deleted text is open to comment at this time. All other material in this addendum is provided for context only and is not open for public review comment except as it relates to the proposed changes.

135-2012*ai*-1 Add Network Port Object Type

Rationale

There is currently no BACnet-visible mechanism for viewing and/or configuring a device's network settings. While this exists to some degree via some properties in the Device object, via Network Layer messages or via BVLL messages, there needs to be a way for BACnet client devices to easily and consistently access and manipulate this information.

This new object will be a mandatory object.

[Change **Clause 22.1.5**, p. 715]

22.1.5 Minimum Device Requirements

A device that conforms to the BACnet protocol ~~and contains an application layer~~ shall:

(a) contain exactly one Device object,
(b) execute the ReadProperty service,
(c) execute the Who-Has and Who-Is services (and thus initiate the I-Have and I-Am services) unless the device is an MS/TP slave device,
(d) execute the WriteProperty service if the device executes the WritePropertyMultiple, AddListElement or RemoveListElement services,
(e) allow the WriteProperty service to modify any properties that are modifiable by the AddListElement or RemoveListElement services,
(f) execute the WriteProperty service if the device contains any objects with properties that are required to be writable
~~and~~
(g) have a configurable device instance that can take on any value across the range 0 .. 4194302
and
(h) contain a Network Port object for each configured network port.

[Add new entries to **Clause 3.3**, p. 7, in alphabetical order]

...
DHCP Dynamic Host Configuration Protocol
...
DNS Domain Name Service
...
NAT Network Address Translation or Port Address Translation RFC 2663

[Add new **Clause 12.X**, p. 459]

12.X Network Port Object

The Network Port object provides access to the configuration and properties of network ports of a device. All BACnet devices shall contain one Network Port object per configured port. It is a local matter whether or not Network Port objects exist for non-configured ports. It is a local matter whether or not the Network Port object is used for non-BACnet ports.

Verification and validation of property values within a Network Port object is a local matter.

Property values which are required to maintain proper operation of the network shall be retained across a device reset.

Network Port objects may optionally support intrinsic reporting to facilitate the reporting of fault conditions. Network Port objects that support intrinsic reporting shall apply the NONE event algorithm.

As specified in Table 12-X and the text below, some properties of the Network Port object are required if the object is used to represent a network of a given type. For example, a Network Port object whose Network_Type is MSTP must include the Max_Master property, and a Network Port object whose Network_Type is BACNET_IPV4 must include the BACnet_IP_Subnet_Mask property. Aside from the properties so required, it is a local matter whether a Network Port object contains properties that do not apply to its Network_Type. For example, a Network Port object whose Network_Type is MSTP may include the BACnet_IP_Subnet_Mask property, although the value of this property would not be used by the network. Some vendors may find it convenient to have all of their Network Port objects support the same list of properties regardless of Network_Type. This is permitted, but not required.

Table 12-X. Properties of the Network Port Object Type

Property Identifier	Property Datatype	Conformance Code
Object_Identifier	BACnetObjectIdentifier	R
Object_Name	CharacterString	R
Object_Type	BACnetObjectType	R
Description	CharacterString	O
Status_Flags	BACnetStatusFlags	R
Reliability	BACnetReliability	R
Out_Of_Service	BOOLEAN	R
Network_Type	BACnetNetworkType	R
Network_Number	Unsigned16	R[1]
Network_Number_Quality	BACnetNetworkNumberQuality	R
Changes_Pending	BOOLEAN	R
Command	BACnetNetworkPortCommand	~~R~~[2]O[2]
MAC_Address	OCTET STRING	O[3]
APDU_Length	Unsigned	R
Link_Speed	REAL	R
Link_Speeds	BACnetARRAY[N] of REAL	O[4]
Link_Speed_Autonegotiate	BOOLEAN	O
Network_Interface_Name	CharacterString	O
BACnet_IP_Mode	BACnetIPMode	O[5]
BACnet_IP_Address	OCTET STRING	O[6]
BACnet_IP_UDP_Port	Unsigned16	O[5]
BACnet_IP_Subnet_Mask	OCTET STRING	O[6]
BACnet_IP_Default_Gateway	OCTET STRING	O[6]
BACnet_IP_Multicast_Address	OCTET STRING	O[7]
BACnet_IP_DNS_Server	BACnetARRAY[N] of OCTET STRING	O[6]
BACnet_IP_DHCP_Enable	BOOLEAN	O[8]
BACnet_IP_DHCP_Lease_Time	Unsigned	O
BACnet_IP_DHCP_Lease_Time_Remaining	Unsigned	O
BACnet_IP_DHCP_Server	OCTET STRING	O
BACnet_IP_NAT_Traversal	BOOLEAN	O[9]
BACnet_IP_Global_Address	BACnetHostNPort	O[10]
BBMD_Broadcast_Distribution_Table	BACnetLIST of BACnetBDTEntry	O[11]
BBMD_Accept_FD_Registrations	BOOLEAN	O[11]
BBMD_Foreign_Device_Table	BACnetLIST of BACnetFDTEntry	O[12]
FD_BBMD_Address	BACnetHostNPort	O[13]
FD_Subscription_Lifetime	Unsigned16	O[13]
Max_Master	Unsigned8	O[14]
Max_Info_Frames	Unsigned8	O[14]
Slave_Proxy_Enable	BOOLEAN	O[15]
Manual_Slave_Address_Binding	BACnetLIST of BACnetAddressBinding	O[15]
Auto_Slave_Discovery	BOOLEAN	O[16]
Slave_Address_Binding	BACnetLIST of BACnetAddressBinding	O[17]
Virtual_MAC_Address_Table	BACnetLIST of BACnetVMACEntry	O[18]

Routing_Table	BACnetLIST of BACnetRouterEntry	O
Event_Detection_Enable	BOOLEAN	O[19,20]
Notification_Class	Unsigned	O[19,20]
Event_Enable	BACnetEventTransitionBits	O[19,20]
Acked_Transitions	BACnetEventTransitionBits	O[19,20]
Notify_Type	BACnetNotifyType	O[19,20]
Event_Time_Stamps	BACnetARRAY[3] of BACnetTimeStamp	O[19,20]
Event_Message_Texts	BACnetARRAY[3] of CharacterString	O[20]
Event_Message_Texts_Config	BACnetARRAY[3] of CharacterString	O[20]
Event_State	BACnetEventState	O[19]
Reliability_Evaluation_Inhibit	BOOLEAN	O
Property_List	BACnetARRAY[N] of BACnetPropertyIdentifier	R
Profile_Name	CharacterString	O

[1] Required to be writable in routers, secure devices, and any other device that requires knowledge of the network number for proper operation.
[2] Shall be present if, and only if, the object supports execution of any of the values of the Command property. If present, this property shall be writable.
[3] Required if the port is not a PTP link. Read-only if the port is a BACnet/IP port or if the network represented by this object requires VMAC addressing.
[4] Required if Link_Speed is writable.
[5] Required to be present if the port is a BACnet/IP port.
[6] Required if the port is a BACnet/IP port. If the BACnet_IP_DHCP property is TRUE, and this property is configured by DHCP, this property shall be read-only.
[7] Required to be present if the port is a BACnet/IP port and supports multicast.
[8] Shall be present if, and only if, the port can be configured by DHCP.
[9] Required to be present if the port is a BACnet/IP port and the device is capable of communicating through a NAT router as described in J.7.8.
[10] Required if the port is a BACnet/IP port and the device is configured to communicate through a NAT router as described in J.7.8
[11] Required to be present if the port is a BACnet/IP port and the device is capable of functioning as a BBMD.
[12] Required if the port is a BACnet/IP port and the device is capable of functioning as a BBMD.
[13] Required to be present if the port is a BACnet/IP port and BACnet_IP_Mode is set to FOREIGN.
[14] Required to be present and writable if the port is an MS/TP port, the device is an MS/TP master node, and the device supports the WriteProperty service.
[15] Required to be present and writable if the port is an MS/TP port, and the device is capable of being a Slave-Proxy device.
[16] Required if the port is an MS/TP port, and the device is capable of being a Slave-Proxy device that implements automatic discovery of slaves.
[17] Required if the port is an MS/TP port, and the device is capable of being a Slave-Proxy device.
[18] Required if the network represented by this object requires VMAC addressing.
[19] These properties are required if the object supports intrinsic reporting.
[20] These properties shall be present only if the object supports intrinsic reporting.

For convenience, the following table further illustrates the properties that are required based on the value of the Network_Type property and the various capabilities of the network types. This table is not meant to be an exhaustive list and is shown for informative purposes.

Table 12-Y. Required Properties of the Network Port Object Type Based on Network_Type

If the value of Network_Type is…	… then these are the additional properties required of the corresponding Network Port Object.
ETHERNET	MAC_Address
MSTP	MAC_Address Max_Master Max_Info_Frames
MSTP (capable of Slave Proxy)	MAC_Address Max_Master Max_Info_Frames Slave_Proxy_Enable Manual_Slave_Address_Binding Auto_Slave_Discovery Slave_Address_Binding
BACNET_IPV4 (BACNET_IP_MODE is NORMAL)	MAC_Address BACnet_IP_Mode BACnet_IP_Address BACnet_IP_UDP_Port BACnet_IP_Subnet_Mask BACnet_IP_Default_Gateway BACnet_IP_DNS_Server
BACNET_IPV4 (BACNET_IP_MODE is FOREIGN)	MAC_Address BACnet_IP_Mode BACnet_IP_Address BACnet_IP_UDP_Port BACnet_IP_Subnet_Mask BACnet_IP_Default_Gateway BACnet_IP_DNS_Server FD_BBMD_Address FD_Subscription_Lifetime
BACNET_IPV4 (BACNET_IP_MODE is BBMD)	MAC_Address BACnet_IP_Mode BACnet_IP_Address BACnet_IP_UDP_Port BACnet_IP_Subnet_Mask BACnet_IP_Default_Gateway BACnet_IP_DNS_Server BBMD_Broadcast_Distribution_Table BBMD_Accept_FD_Registrations BBMD_Foreign_Device_Table

12.X.1 Object_Identifier

This 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. The instance number (see Clause 20.2.14) shall correspond to the Port ID of the associated network as described in Clause 6.

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 property, of type BACnetObjectType, indicates membership in a particular object type class. The value of this property shall be NETWORK_PORT.

12.X.4 Description

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

12.X.5 Status_Flags

This property, of type BACnetStatusFlags, represents four Boolean flags that indicate the general "health" of the Network. The four flags are:

{IN_ALARM, FAULT, OVERRIDDEN, OUT_OF_SERVICE}

where:

IN_ALARM	Logical TRUE (1) if the Event_State property is present and does not have a value of NORMAL, otherwise logical FALSE (0).
FAULT	Logical TRUE (1) if the Reliability property does not have a value of NO_FAULT_DETECTED, otherwise logical FALSE (0).
OVERRIDDEN	Always 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.6 Reliability

This property, of type BACnetReliability, provides an indication of whether the Network Port object, the network port, and the network connected to the port are "reliable" as far as the BACnet Device can determine and, if not, why.

12.X.7 Out_Of_Service

The Out_Of_Service property, of type BOOLEAN, is an indication whether (TRUE) or not (FALSE) the network port is out of service.

When a network port is Out_Of_Service, all BACnet communication through that port shall be disabled, and writing any value other than RESTART_PORT, DISCONNECT, and DISCARD_CHANGES to the Command property shall result in an error response with an ‘Error Class’ of PROPERTY and ‘Error Code’ of VALUE_OUT_OF_RANGE.

12.X.8 Network_Type

This property, of type BACnetNetworkType, represents the type of network this Network Port object is representing.

This property shall have one of the following values:

ETHERNET	ISO 8802-3 (“Ethernet”), as defined in Clause 7
ARCNET	ARCNET, as defined in Clause 8
MSTP	MS/TP, as defined in Clause 9
PTP	Point-To-Point, as defined in Clause 10
LONTALK	LonTalk, as defined in Clause 11
BACNET_IPV4	BACnet/IP, as defined in Annex J

ZIGBEE	ZigBee, as defined in Annex O
VIRTUAL	Indicates that this port represents the configuration and properties of a virtual network as described in Annex H.2.
NON_BACNET	Indicates that this port represents a non-BACnet network.
<Proprietary Enum Values>	A vendor may use other proprietary enumeration values to indicate that this port represents the use of message structures, procedures, and medium access control techniques other than those contained in this standard. For proprietary extensions of this enumeration, see Clause 23.1 of this standard.

12.X.9 Network_Number

This property, of type Unsigned16, represents the BACnet network number associated with this network.

The range for this property shall be 0 .. 65534. A Network_Number of 0 indicates that the Network_Number is not known or cannot be determined. Writing 0 to the Network_Number property shall force the value of the Network_Number_Quality property to UNKNOWN and allows the device to attempt to learn the network number, if possible. Writing a value other than 0 shall force the Network_Number_Quality property to CONFIGURED.

If the Network_Type is PTP or NON_BACNET, then this property shall be read-only and contain a value of 0.

This property shall be writable in routers, secure devices, and any other device that requires knowledge of the network number for proper operation. Routers are permitted to refuse a value of 0. In that case, the write request shall result in an error response with ‘Error Class’ of PROPERTY and an ‘Error Code’ of VALUE_OUT_OF_RANGE.

If this property is writable, then a successful write to this property shall set the Changes_Pending property to TRUE. A value written to this property shall become effective when the device receives a ReinitializeDevice service request with a 'Reinitialized State of Device' of ACTIVATE_CHANGES or WARMSTART.

12.X.10 Network_Number_Quality

This read-only property, of type BACnetNetworkNumberQuality, represents the current quality of the Network_Number property. If the Network_Type is PTP, the Network_Number_Quality shall be CONFIGURED.

This property shall have one of the following values:

UNKNOWN	None of the below meanings.
LEARNED	The Network_Number was learned via receipt of a Network-Number-Is message with a flag value of 0 (learned).
LEARNED_CONFIGURED	The Network_Number was learned via receipt of a Network-Number-Is message with a flag value of 1 (configured).
CONFIGURED	The Network_Number is configured for this port.

12.X.11 Changes_Pending

This property, of type BOOLEAN, indicates whether the configuration settings in the Network Port object map to the current configuration settings. A value of FALSE indicates that the configuration settings reflect the current port configuration information. A value of TRUE indicates the configuration settings have been modified but have not been activated on the port.

When a value is written to a property that requires activation, the value of the Changes_Pending property shall automatically be set to TRUE, indicating that the current property values are not the values actively in use.

It is necessary for the client to initiate a ReinitializeDevice service request with a 'Reinitialized State of Device' of ACTIVATE_CHANGES or WARMSTART in order to activate the currently visible configuration settings. This interaction is necessary to support atomic updating of multiple properties when modifying a network port configuration.

It is a local matter as to whether or not resetting the device by means other than a ReinitializeDevice service with a 'Reinitialized State of Device' of ACTIVATE_CHANGES or WARMSTART discards pending changes, activates pending changes, or leaves the changes pending.

It is a local matter whether, or not, a device refuses requests to write to a Network Port object if:

any Network Port object has pending changes,
the write request is from a device other than that which wrote the existing pending changes, and
the write would result in pending changes in any Network Port object.

When refusing such a request, the device shall return a Result(-) with an ‘Error Class’ of DEVICE and an ‘Error Code’ of CONFIGURATION_IN_PROGRESS.

12.X.12 Command

This property, of type BACnetNetworkPortCommand, is used to request that the Network Port object perform various actions.

When this property is written, the sequence of operations shall be as follows:

1. Perform any necessary validation. If Result(-) is returned, this property shall retain the value that it had before the write was attempted and no change shall be made to any other property of the object.
2. If validation succeeds, this property shall be set to the value written and a Result(+) be returned.
3. The device shall begin performing the requested command actions.
4. When the object is able to accept another command, the Command property shall be set to IDLE. This may occur immediately, when the actions have completed, or when the actions have proceeded to a point that allows the implementation to accept another command. The exact timing is a local matter.

When this property has a value other than IDLE, any attempt to write to it shall result in the return of a Result(-) with an 'Error Class' of OBJECT and an 'Error Code' of BUSY.

Writing a value of IDLE to this property shall result in the return of a Result(-) with an 'Error Class' of PROPERTY and an 'Error Code' of OUT_OF_RANGE.

If the value of the Changes_Pending property is TRUE, writing a value other than DISCARD_CHANGES (if supported) to the Command property shall result in the return of a Result(-) with an ‘Error Class’ of PROPERTY and an ‘Error Code’ of INVALID_VALUE_IN_THIS_STATE.

Any of the following commands may be written to this property:

DISCARD_CHANGES	If the device supports this command, this object shall revert to the set of property values that were contained in this object when Changes_Pending was last equal to FALSE. Changes_Pending shall be set to FALSE, and Command shall be set to IDLE. If the device does not support this command, writing this value shall result in the return of a Result(-) with an ‘Error Class’ of PROPERTY and an ‘Error Code’ of OPTIONAL_FUNCTIONALITY_NOT_SUPPORTED. Changes_Pending shall remain TRUE, and Command shall be set to IDLE.
RENEW_FD_REGISTRATION	This port shall attempt to renew its foreign device registration with the BBMD indicated in FD_BBMD_Address. If the value of Network_Type is not BACNET_IPV4, or if the value of BACnet_IP_Mode is not FOREIGN, writing this value shall result in the return of a Result(-) with an ‘Error Class’ of PROPERTY and an ‘Error Code’ of VALUE_OUT_OF_RANGE.

	If the attempt to renew the foreign device registration fails, then the value of the Reliability property shall be set to RENEW_FD_REGISTRATION_FAILURE. It is a local matter whether the value of this property remains at RENEW_FD_REGISTRATION until the registration has completed (whether in success or failure), and then returns to IDLE; or whether the property returns to IDLE once the registration process has been initiated and the object is prepared to accept another command.
RESTART_SLAVE_DISCOVERY	The port shall restart the slave detection algorithm as described in Clauses 12.X.39 through 12.X.41, and 16.10.2. If the value of Network_Type is not MSTP, writing this value shall result in the return of a Result(-) with an 'Error Class' of PROPERTY and an 'Error Code' of VALUE_OUT_OF_RANGE. If the value of Network_Type is MSTP but the device does not support MS/TP Slave Proxy functionality, writing this value shall result in the return of a Result(-) with an 'Error Class' of PROPERTY and an 'Error Code' of OPTIONAL_FUNCTIONALITY_NOT_SUPPORTED. The value of the Command property shall return to IDLE as soon as discovery has been initiated. The discovery process will typically require a significant amount of additional time.
RENEW_DHCP	If DHCP is supported, then this port shall attempt to renew the DHCP lease for this port. If the port cannot be made to renew the DHCP lease, writing this value shall result in the return of a Result(-) with an 'Error Class' of PROPERTY and an 'Error Code' of OPTIONAL_FUNCTIONALITY_NOT_SUPPORTED. If the value of Network_Type is BACNET_IPV4 and the BACnet_IP_DHCP_Enable property is not present, writing this value shall result in the return of a Result(-) with an 'Error Class' of PROPERTY and an 'Error Code' of OPTIONAL_FUNCTIONALITY_NOT_SUPPORTED. If the value of Network_Type is not BACNET_IPV4, writing this value shall result in the return of a Result(-) with an 'Error Class' of PROPERTY and an 'Error Code' of VALUE_OUT_OF_RANGE. If the value of Network_Type is BACNET_IPV4 and the value of BACnet_IP_DHCP_Enable is FALSE, writing this value shall result in the return of a Result(-) with an 'Error Class' of PROPERTY and an 'Error Code' of VALUE_OUT_OF_RANGE. If the attempt to renew the DHCP address fails, then the value of the Reliability property shall be set to RENEW_DHCP_FAILURE. It is a local matter whether the value of this property remains at RENEW_DHCP until the renewal has completed (whether in success or failure), and then returns to IDLE; or whether the property returns to IDLE once the renewal process has been initiated and the object is prepared to accept another command.

RESTART_AUTONEGOTIATION	This port shall restart its link speed auto-negotiation algorithm. If the value of the Link_Speed_Autonegotiate property is FALSE, writing this value shall result in the return of a Result(-) with an 'Error Class' of PROPERTY and an 'Error Code' of VALUE_OUT_OF_RANGE. If this port does not support auto-negotiation when the command is issued, writing this value shall result in the return of a Result(-) with an 'Error Class' of PROPERTY and an 'Error Code' of OPTIONAL_FUNCTIONALITY_NOT_SUPPORTED. If the auto-negotiation algorithm fails, the value of the Reliability property shall be set to RESTART_AUTONEGOTIATION_FAILURE. It is a local matter whether the value of this property remains at RESTART_AUTONEGOTIATION until the auto-negotiation has completed (whether in success or failure), and then returns to IDLE; or whether the property returns to IDLE once the auto-negotiation process has been initiated and the object is prepared to accept another command.
DISCONNECT	This port shall terminate the network connection. If the value of Network_Type is not PTP, writing this value shall result in the return of a Result(-) with an 'Error Class' of PROPERTY and an 'Error Code' of VALUE_OUT_OF_RANGE. It is a local matter whether the value of this property remains at DISCONNECT until the disconnection has completed (whether in success or failure), and then returns to IDLE; or whether the property returns to IDLE once the disconnection process has been initiated and the object is prepared to accept another command.
RESTART_PORT	This port shall attempt to restart and reconnect to the network as if the device were reinitialized. All data that was learned, cached, or otherwise automatically determined for the port's operation shall be cleared. All initialization, negotiation, and registration functions the port is expected to perform upon device initialization shall be performed again. If the restart fails, the value of the Reliability property shall be set to RESTART_FAILURE. It is a local matter whether the value of this property remains at RESTART_PORT until the restart has completed (whether in success or failure), and then returns to IDLE; or whether the property returns to IDLE once the restart process has been initiated and the object is prepared to accept another command. If the device cannot perform the restart of the port without a reinitialization of the entire device, writing this value shall result in the return of a Result(-) with an 'Error Class' of PROPERTY and an 'Error Code' of OPTIONAL_FUNCTIONALITY_NOT_SUPPORTED.
<Proprietary Enum Values>	A vendor may use other proprietary enumeration values to allow command values other than those defined by the standard. For proprietary extensions of this enumeration, see Clause 23.1 of this standard.

A proprietary command failure shall result in the value of the Reliability property being set to PROPRIETARY_COMMAND_FAILURE and the value of this property being set to IDLE.

It is a local matter whether the value of this property remains at the proprietary value until the proprietary action has completed (whether in success or failure), and then returns to IDLE; or whether the property returns to IDLE once the action has been initiated and the object is prepared to accept another command.

This enumerated value is extensible. Writing an unknown value to this property shall result in the return of a Result(-) with an 'Error Class' of PROPERTY and an 'Error Code' of VALUE_OUT_OF_RANGE.

12.X.13 MAC_Address

This property, of type OCTET STRING, contains the BACnet MAC address used on this network. The value of this property shall be conveyed with the most significant octet first. If Network_Type is BACNET_IPV4, then the value of this property shall contain the six octet combination of the BACnet_IP_Address and BACnet_IP_UDP_Port and shall be read-only. If the value of Network_Type is a value that represents a port that requires VMAC addressing, then the value of this property shall be read-only and contain the VMAC address.

If this property is writable, then a successful write to this property shall set the Changes_Pending property to TRUE. The value of this property shall become effective when the device receives a ReinitializeDevice service request with a 'Reinitialized State of Device' of ACTIVATE_CHANGES or WARMSTART.

12.X.14 APDU_Length

This property, of type Unsigned, is the maximum number of octets that may be contained in a single indivisible application protocol data unit sent or received on this port. The value of this property shall be greater than or equal to 50. This property also indicates the maximum number of octets that may be contained in a single individual network service data unit sent or received on this port.

12.X.15 Link_Speed

This property, of type REAL, represents the network communication rate as the number of bits per second. A value of 0 indicates an unknown communication rate.

If the value of the Link_Speed_Autonegotiate property is TRUE, then this property shall be read-only.

If this property is writable, writing a value to this property that is not present in the Link_Speeds property shall result in the return of a Result(-) with an 'Error Class' of PROPERTY and an 'Error Code' of VALUE_OUT_OF_RANGE.

If this property is writable, then a successful write to this property shall set the Changes_Pending property to TRUE. A value written to this property shall become effective when the device receives a ReinitializeDevice service request with a 'Reinitialized State of Device' of ACTIVATE_CHANGES or WARMSTART.

12.X.16 Link_Speeds

This read-only property, of type BACnetARRAY of REAL, is an array of the link speeds supported by this network port.

12.X.17 Link_Speed_Autonegotiate

This property, of type BOOLEAN, represents the auto negotiation setting of the network port.

A value of TRUE indicates that the device automatically determines the speed of this network port. In this case, Link_Speed shall be read-only and indicate the determined speed, if available. A value of FALSE indicates that the link speed is determined by the value of the Link_Speed property.

If this property is writable, then a successful write to this property shall set the Changes_Pending property to TRUE. A value written to this property shall become effective when the device receives a ReinitializeDevice service request with a 'Reinitialized State of Device' of ACTIVATE_CHANGES or WARMSTART.

12.X.18 Network_Interface_Name

This property, of type CharacterString, is used to identify the network interface hardware to which this network port is bound. For example, if Network_Type is BACNET_IPV4, the value of this property identifies the Ethernet hardware interface that this network port is using to communicate.

If this property is writable, then a successful write to this property shall set the Changes_Pending property to TRUE. A value written to this property shall become effective when the device receives a ReinitializeDevice service request with a 'Reinitialized State of Device' of ACTIVATE_CHANGES or WARMSTART.

12.X.19 BACnet_IP_Mode

This property, of type BACnetIPMode, indicates the BACnet/IP mode of this network port.
This property shall have one of the following values:

NORMAL The device is operating as neither a foreign device nor a BBMD over this network port.

FOREIGN The device is operating as a foreign device over this network port.

BBMD The device is operating as a BBMD over this network port.

Writing to this property shall set the Changes_Pending property to TRUE. A value written to this property shall become effective when the device receives a ReinitializeDevice service request with a 'Reinitialized State of Device' of ACTIVATE_CHANGES or WARMSTART.

12.X.20 BACnet_IP_Address

This property, of type OCTET_STRING, indicates the IP address of this network port. This property shall be conveyed most significant octet first. If the BACnet_IP_DHCP_Enable property is TRUE, this property shall be read-only.

If this property is writable, then a successful write to this property shall set the Changes_Pending property to TRUE. A value written to this property shall become effective when the device receives a ReinitializeDevice service request with a 'Reinitialized State of Device' of ACTIVATE_CHANGES or WARMSTART.

12.X.21 BACnet_IP_UDP_Port

This property, of type Unsigned16, indicates the UDP port number of this network port.

If this property is writable, then a successful write to this property shall set the Changes_Pending property to TRUE. A value written to this property shall become effective when the device receives a ReinitializeDevice service request with a 'Reinitialized State of Device' of ACTIVATE_CHANGES or WARMSTART.

12.X.22 BACnet_IP_Subnet_Mask

This property, of type OCTET STRING, indicates the subnet mask for this network. This property shall be conveyed with the most significant octet first. If the BACnet_IP_DHCP_Enable property is TRUE, this property shall be read-only.

If this property is writable, then a successful write to this property shall set the Changes_Pending property to TRUE. A value written to this property shall become effective when the device receives a ReinitializeDevice service request with a 'Reinitialized State of Device' of ACTIVATE_CHANGES or WARMSTART.

12.X.23 BACnet_IP_Default_Gateway

This property, of type OCTET STRING, indicates the IP address of the default gateway for this network. This property shall be conveyed with the most significant octet first. If the BACnet_IP_DHCP_Enable property is TRUE, this property shall be read-only.

If this property is writable, then a successful write to this property shall set the Changes_Pending property to TRUE. A value written to this property shall become effective when the device receives a ReinitializeDevice service request with a 'Reinitialized State of Device' of ACTIVATE_CHANGES or WARMSTART.

12.X.24 BACnet_IP_Multicast_Address

This property, of type OCTET STRING, contains the BACnet/IP multicast group address to be used for the distribution of

BACnet broadcast messages. See Clause J.8. The value of this property shall be conveyed with the most significant octet first.

A value of X'00000000' indicates that BACnet/IP multicast is not used.

If present, this property shall be writable. Writing to this property shall set the Changes_Pending property to TRUE. A value written to this property shall become effective when the device receives a ReinitializeDevice service request with a 'Reinitialized State of Device' of ACTIVATE_CHANGES or WARMSTART.

12.X.25 BACnet_IP_DNS_Server

This property, of type BACnetARRAY[N] of OCTET STRING containing at least one entry, indicates the DNS server used by this network port for Internet host name resolution. The values of this property shall be conveyed with the most significant octet first.

A value of X'00000000' in an array entry indicates that the DNS server address is not available or is not configured.

If the BACnet_IP_DHCP_Enable property is TRUE, and this property value is configured by DHCP, then this property shall be read-only.

If this property is writable, then a successful write to this property shall set the Changes_Pending property to TRUE. A value written to this property shall become effective when the device receives a ReinitializeDevice service request with a 'Reinitialized State of Device' of ACTIVATE_CHANGES or WARMSTART.

12.X.26 BACnet_IP_DHCP_Enable

This property, of type BOOLEAN, indicates whether or not this network is configured via DHCP. A value of TRUE indicates that DHCP configuration is enabled, FALSE indicates it is not.

This property is required if DHCP configuration is supported by this network port.

If this property is writable, then a successful write to this property shall set the Changes_Pending property to TRUE. A value written to this property shall become effective when the device receives a ReinitializeDevice service request with a 'Reinitialized State of Device' of ACTIVATE_CHANGES or WARMSTART.

12.X.27 BACnet_IP_DHCP_Lease_Time

This read-only property, of type Unsigned, indicates the lease time in seconds of the last DHCP lease obtained for the port. If BACnet_IP_DHCP_Enable is FALSE, or no lease has been acquired, or the value is unknown, this property shall be 0.

12.X.28 BACnet_IP_DHCP_Lease_Time_Remaining

This read-only property, of type Unsigned, indicates the lease time in seconds remaining of the last DHCP lease obtained for the port. If BACnet_IP_DHCP_Enable is FALSE, or no lease has been acquired, or the value is unknown, this property shall be 0.

12.X.29 BACnet_IP_DHCP_Server

This read-only property, of type OCTET STRING, indicates the address of the DHCP server from which the last DHCP lease was obtained for the port. If the address of the DHCP server cannot be determined, the value of this property shall be X'00000000'.

12.X.30 BACnet_IP_NAT_Traversal

This property, of type BOOLEAN, indicates whether (TRUE) or not (FALSE) this port is configured to operate in a NAT environment, as described in Annex J.7.8, and the global address is indicated by the value of the BACnet_IP_Global_Address property.

If present, this property shall be writable. Writing to this property shall set the Changes_Pending property to TRUE. A value written to this property shall become effective when the device receives a ReinitializeDevice service request with a 'Reinitialized State of Device' of ACTIVATE_CHANGES or WARMSTART.

12.X.31 BACnet_IP_Global_Address

This property, of type BACnetHostNPort, indicates the global address and UDP port from which the network port can be accessed from the global side of a NAT router. How the public IP address and UDP port are determined is a local matter.

The 'none' choice in the BACnetHostAddress portion and a value of X'0000' in the port portion indicates that the global address cannot be determined or is not yet configured.

If the device does not automatically determine the global address, then this property shall be writable.

If this property is writable, then a successful write to this property shall set the Changes_Pending property to TRUE. A value written to this property shall become effective when the device receives a ReinitializeDevice service request with a 'Reinitialized State of Device' of ACTIVATE_CHANGES or WARMSTART.

12.X.32 BBMD_Broadcast_Distribution_Table

This property, of type BACnetLIST of BACnetBDTEntry, is required to be present and writable if BACnet_IP_Mode is BBMD.
The value of this property maps to the BDT (as specified in Annex J) for this port as follows:

(a) The current value of the BDT may be read at any time with the Read-Broadcast-Distribution-Table BVLL message.
(b) If this property has no pending changes, reading this property shall return the current value of the BDT.
(c) If this property has pending changes, reading this property shall return the last value written to the property, and not the current value of the BDT.
(d) If this property has pending changes, reading the BDT via Read-Broadcast-Table BVLL shall return the current value of the BDT.
(e) If a list entry contains a host name, then the corresponding entry in the Read-Broadcast-Distribution-Table-Ack BVLL message shall contain the IP address of the resolved host name or X'000000000000' to indicate that the host name has not been resolved.

The 'none' choice of the BACnetHostAddress portion shall not be used for list entries.

Writing to this property shall set the Changes_Pending property to TRUE. A value written to this property shall become effective when the device receives a ReinitializeDevice service request with a 'Reinitialized State of Device' of ACTIVATE_CHANGES or WARMSTART.

12.X.33 BBMD_Accept_FD_Registrations

This property, of type BOOLEAN, indicates whether (TRUE) or not (FALSE) this device shall accept foreign device registrations. This property is required to be present and writable if BACnet_IP_Mode is BBMD.

Writing to this property shall set the Changes_Pending property to TRUE. A value written to this property shall become effective when the device receives a ReinitializeDevice service request with a 'Reinitialized State of Device' of ACTIVATE_CHANGES or WARMSTART.

12.X.34 BBMD_Foreign_Device_Table

This read-only property, of type BACnetLIST of BACnetFDTEntry, is required to be present if BBMD_Accept_FD_Registrations is TRUE.

The value of this property reflects the current value of the foreign device table. Each entry shall contain the following information:

bacnetip-address	The 6-octet B/IP address of the registered foreign device.
time-to-live	The time to live for the entry, as provided at the time of registration.
remaining-time-to-live	The remaining time to live for the entry. This includes the grace period added at the time of registration.

12.X.35 FD_BBMD_Address

This property, of type BACnetHostNPort, indicates the BBMD with which the local device is to register as a foreign device when BACnet_IP_Mode is FOREIGN. This property shall be present and writable if BACnet_IP_Mode is FOREIGN.

If this property is writable, then a successful write to this property shall set the Changes_Pending property to TRUE. A value written to this property shall become effective when the device receives a ReinitializeDevice service request with a 'Reinitialized State of Device' of ACTIVATE_CHANGES or WARMSTART.

12.X.36 FD_Subscription_Lifetime

This property, of type Unsigned16, indicates the Time-To-Live value, in seconds, to be used in the Register-Foreign-Device BVLL message. This property shall be present and writable if BACnet_IP_Mode is FOREIGN.

If this property is writable, then a successful write to this property shall set the Changes_Pending property to TRUE. A value written to this property shall become effective when the device receives a ReinitializeDevice service request with a 'Reinitialized State of Device' of ACTIVATE_CHANGES or WARMSTART.

12.X.37 Max_Master

This property, of type Unsigned8, shall be present if the Network_Type is MSTP and the device is a master node on the MS/TP network connected to this port. If the device supports execution of the WriteProperty service, then this property shall be writable and the valid range for the value of this property shall be 0 to 127. Otherwise, its value shall be 127. See Clause 9.5.3.

Writing to this property shall set the Changes_Pending property to TRUE. A value written to this property shall become effective when the device receives a ReinitializeDevice service request with a 'Reinitialized State of Device' of ACTIVATE_CHANGES or WARMSTART.

12.X.38 Max_Info_Frames

This property, of type Unsigned8, shall be present if the Network_Type is MSTP and the device is a master node on the MS/TP network connected to this port. The value of Max_Info_Frames specifies the maximum number of information frames the node may send on this port before it must pass the token. If the device supports execution of the WriteProperty service, then this property shall be writable and the valid range for the value of this property shall be 1 to 255. Otherwise, its value shall be 1. See Clause 9.5.3.

Writing to this property shall set the Changes_Pending property to TRUE. A value written to this property shall become effective when the device receives a ReinitializeDevice service request with a 'Reinitialized State of Device' of ACTIVATE_CHANGES or WARMSTART.

12.X.39 Slave_Proxy_Enable

This property, of type BOOLEAN, is an indication of whether (TRUE) or not (FALSE) the device will perform Slave-Proxy functions for this port. This property shall be present and writable if the device is capable of performing the functions of a Slave-Proxy device on this port.

12.X.40 Manual_Slave_Address_Binding

This property, of type BACnetLIST of BACnetAddressBinding, describes the manually configured set of slave devices for which this device is acting as a Slave Proxy as described in 16.10.2. This property shall be present and writable if the device is capable of performing the functions of a Slave-Proxy device on this port.

This property is used to manually configure a set of slave devices connected to this port for which this device will be a proxy. This property allows a Slave-Proxy that does not support automatic slave discovery to be configured with a set of slaves for which this device will be a proxy. It also allows a Slave-Proxy device to be a proxy for Slave devices that do not support the special object instance of 4194303 as described in Clause 12. When enabled, the Slave-Proxy device shall periodically check each device that is in this list, and not in the Slave_Address_Binding list, by reading the device's Protocol_Services_Supported property from the device's Device object using the ReadProperty service. If the device responds and indicates that it does not execute the Who-Is service, it shall be added to the Slave_Address_Binding property. The period at which the devices are checked is a local matter.

12.X.41 Auto_Slave_Discovery

This property, of type BOOLEAN, is an indication whether (TRUE) or not (FALSE) the device will perform automatic slave detection functions for this port. This property shall be present if the device is capable of performing the functions of a Slave-Proxy device on this port.

Slave detection shall be accomplished by the proxy device using ReadProperty services to read, at a minimum, the Device object's Protocol_Services_Supported property for each MAC address on the network connected to this port. The ReadProperty service shall use the special object instance of 4194303 as described in Clause 12. If the device is found to support execution of the Who-Is service, it is ignored; otherwise, the device shall be added to the Slave_Address_Binding property. The slave detection algorithm shall be repeated periodically. The period at which it is repeated is a local matter.

12.X.42 Slave_Address_Binding

This property, of type BACnetLIST of BACnetAddressBinding, describes the set of slave devices for which this device is acting as a Slave-Proxy on this port as described in 16.10.2. This property shall be present if the device is capable of performing the functions of a Slave-Proxy device on this port.

The set of devices described by the Slave_Address_Binding property consists of those devices described in the Manual_Slave_Address_Binding and those devices that are automatically discovered. When enabled, the Slave-Proxy device shall periodically check each device in this list by reading the device's Protocol_Services_Supported property from the device's Device object using the ReadProperty service. If the device fails to respond or indicates that it executes the Who-Is service, it shall be removed from the list. The period at which the devices are checked is a local matter.

12.X.43 Virtual_MAC_Address_Table

This property, of type BACnetLIST of BACnetVMACEntry, is the list of VMAC entries as described in Annex H.7.

VMAC table entries shall contain the following information:

virtual-mac-address	The virtual MAC address used for the native MAC address indicated in the native-mac-address portion. The maximum size of the Virtual MAC address shall be 6 octets.
native-mac-address	The native MAC address used by the datalink to address the node identified by the virtual MAC address.

If this property is writable, then a successful write to this property will set the Changes_Pending property to TRUE. A value written to this property shall become effective when the device receives a ReinitializeDevice service request with a 'Reinitialized State of Device' of ACTIVATE_CHANGES or WARMSTART.

12.X.44 Routing_Table

This read-only property, of type BACnetLIST of BACnetRouterEntry, contains the table of first hop routers to remote networks reachable through this port.

Router table entries shall contain the following information:

network-number	The network number reachable through the router specified by mac-address.
mac-address	The MAC address of the router on the network connected to this port that leads directly or indirectly to that network number.
status	Conveys whether the associated network is able to receive traffic. The values for this field are: AVAILABLE, BUSY, and DISCONNECTED.
performance-index	This optional field is used to convey the performance index as conveyed in an I-Could-Be-Router-To-Network network layer message. See Clause 6.4.3.

12.X.45 Event_Detection_Enable

This property, of type BOOLEAN, indicates whether (TRUE) or not (FALSE) intrinsic reporting is enabled in the object and controls whether (TRUE) or not (FALSE) the object will be considered by event summarization services.

This property is expected to be set during system configuration and is not expected to change dynamically.

When this property is FALSE, Event_State shall be NORMAL, and the properties Acked_Transitions, Event_Time_Stamps, and Event_Message_Texts shall be equal to their respective initial conditions.

12.X.46 Notification_Class

This property, of type Unsigned, shall specify the instance of the Notification Class object to use for event-notification-distribution.

12.X.47 Event_Enable

This property, of type BACnetEventTransitionBits, shall convey three flags that separately enable and disable the distribution of TO_OFFNORMAL, TO_FAULT, and TO_NORMAL notifications (see Clause 13.2.5). A device is allowed to restrict the set of supported values for this property but shall support (T, T, T) at a minimum.

12.X.48 Acked_Transitions

This read-only property, of type BACnetEventTransitionBits, shall convey three flags that separately indicate the acknowledgment state for TO_OFFNORMAL, TO_FAULT, and TO_NORMAL events (see Clause 13.2.2.1.5). Each flag shall have the value TRUE if no event of that type has ever occurred for the object.

12.X.49 Notify_Type

This property, of type BACnetNotifyType, shall convey whether the notifications generated by the object should be Events or Alarms. The value of the property is used as the value of the ‘Notify Type’ service parameter in event notifications generated by the object.

12.X.50 Event_Time_Stamps

This read-only property, of type BACnetARRAY[3] of BACnetTimeStamp, shall convey the times of the last TO_OFFNORMAL, TO_FAULT, and TO_NORMAL events, (see Clause 13.2.2.1). Time stamps of type Time or Date shall have X'FF' in each octet, and Sequence number time stamps shall have the value 0 if no event of that type has ever occurred for the object.

12.X.51 Event_Message_Texts

This read-only property, of type BACnetARRAY[3] of CharacterString, shall convey the message text values of the last for TO_OFFNORMAL, TO_FAULT, and TO_NORMAL events, respectively (see Clause 13.2.2.1). If a particular type of event has yet to occur, an empty string shall be stored in the respective array element.

12.X.52 Event_Message_Texts_Config

This property, of type BACnetARRAY[3] of CharacterString, contains the character strings which are the basis for the 'Message Text' parameter for the event notifications of TO_OFFNORMAL, TO_FAULT, and TO_NORMAL events, respectively, generated by this object. The character strings may optionally contain proprietary text substitution codes to incorporate dynamic information such as date and time or other information.

12.X.53 Event_State

The Event_State property, of type BACnetEventState, is included in order to provide a way to determine whether this object has an active event state associated with it (see Clause 13.2.2.1). If the object supports event reporting, then the Event_State property shall indicate the event state of the object. If the object does not support event reporting, then the value of this property shall be NORMAL.

12.X.54 Reliability_Evaluation_Inhibit

This property, of type BOOLEAN, indicates whether (TRUE) or not (FALSE) reliability evaluation is disabled in the object. This property is a runtime override that allows temporary disabling of reliability evaluation.

When reliability evaluation is disabled, the Reliability property shall have the value NO_FAULT_DETECTED unless Out_Of_Service is TRUE and an alternate value has been written to the Reliability property.

12.X.55 Property_List

This read-only property is a BACnetARRAY of property identifiers, one property identifier 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.56 Profile_Name

This 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.

[Change **Clause 6.6**, p. 66]

BACnet routers are devices that interconnect two or more BACnet networks to form a BACnet internetwork. ~~A router may, or may not, provide BACnet application layer functionality.~~ *BACnet routers shall, at a minimum, implement the device requirements as specified in Clause 22.1.5. Table 6-1 specifies the maximum NPDU length of the different data link layer types. Routers shall be capable of routing the maximum sized NPDUs between any two of those data link layers supported by the router based on the destination data link maximum NPDU size*. BACnet routers make use of BACnet network layer protocol messages to maintain their routing tables. Routers perform the routing tasks described in *Clause* 6.5. See Figure 6-12 for a flow chart of router operation.

[Change **Clause 9.5.2**, p. 95]

TS "This Station," the MAC address of this node. ~~TS is generally read from a hardware DIP switch, or from nonvolatile memory.~~ *This variable represents the value of the MAC_Address property of the node's Network Port object which represents this MS/TP port.* Valid values for TS are 0 to 254. The value 255 is used to denote broadcast when used as a destination address but is not allowed as a value for TS.

[Change **Clause 9.5.3**, p. 96]

N_{max_master} This parameter represents the value of the Max_Master property of the node's ~~Device object~~ *Network Port object which represents this MS/TP port*. The value of Max_Master specifies the highest allowable address for master nodes. The valid range of this parameter is 0 to 127. ~~The value of Max_Master shall be less than or equal to 127. If Max_Master is not writable in a node, its value shall be 127.~~

$N_{max\_info\_frames}$ This parameter represents the value of the Max_Info_Frames property of the node's ~~Device object~~ *Network Port object which represents this MS/TP port*. The value of Max_Info_Frames specifies the maximum number of information frames the node may send before it must pass the token. Max_Info_Frames may have different values on different nodes *but shall have a minimum value of 1 and a maximum value of 255*. This may be used to allocate more or less of the available link bandwidth to particular nodes. ~~If Max_Info_Frames is not writable in a node, its value shall be 1.~~

[Change **Table 12-13**, p. 199]

Table 12-13. Properties of the Device Object Type

Property Identifier	Property Datatype	Conformance Code
...	...	...
Local_Time	Time	O[3,4,~~15~~*12*]
Local_Date	Date	O[3,4,~~15~~*12*]
…	…	…
Max_Master	Unsigned~~(1..127)~~	O[6]
Max_Info_Frames	Unsigned	O[6]
...	...	...
Backup_Preparation_Time	Unsigned16	O[~~16~~*13*]
Restore_Preparation_Time	Unsigned16	O[~~16~~*13*]
Restore_Completion_Time	Unsigned16	O[~~16~~*13*]
…	…	…
~~Slave_Proxy_Enable~~	~~BACnetARRAY[N] of BOOLEAN~~	~~O[10]~~
~~Manual_Slave_Address_Binding~~	~~BACnetLIST of BACnetAddressBinding~~	~~O[10,12]~~
~~Auto_Slave_Discovery~~	~~BACnetARRAY[N] of BOOLEAN~~	~~O[10,11]~~
~~Slave_Address_Binding~~	~~BACnetLIST of BACnetAddressBinding~~	~~O[10,12]~~
Last_Restart_Reason	BACnetRestartReason	O[~~13~~*10*]
Time_Of_Device_Restart	BACnetTimeStamp	O[~~13~~*10*]
Restart_Notification_Recipients	BACnetLIST of BACnetRecipient	O[~~17~~*14*]
UTC_Time_Synchronization_Recipients	BACnetLIST of BACnetRecipient	O[5]
Time_Synchronization_Interval	Unsigned	O[~~14~~*11*]
Align_Intervals	BOOLEAN	O[~~14~~*11*]
Interval_Offset	Unsigned	O[~~14~~*11*]
…	…	…

…

[~~10~~] ~~This property is required if, and shall be present only if, the device is capable of being a Slave-Proxy device.~~

[~~11~~] ~~This property is required if, and shall be present only if, the device is capable of being a Slave-Proxy device that implements automatic discovery of slaves.~~

[~~12~~] ~~This property shall be writable if the device is directly connected to an MS/TP network.~~

[~~13~~*10*] These properties are required if the device supports the restart procedure as described in Clause 19.3.

[~~14~~*11*] These properties are required if, and shall be present only if, Time_Synchronization_Recipients or UTC_Time_Synchronization_Recipients is present. If present, these properties shall be writable.

[~~15~~*12*] These properties shall be present if the device is capable of tracking date and time.

[~~16~~*13*] These properties are required if, and shall be present only if, the device supports execution of the backup and restore procedures as described in Clause 19.1 and cannot respond to subsequent communications within the minimum value it will accept in its APDU_Timeout property.

[~~17~~*14*] This property is required if, and shall be present only if, the device supports execution of the restart procedure as described in Clause 19.3.

[Change **Clause 12.11.18**, p. 202]

12.11.18 Max_APDU_Length_Accepted

This property, of type Unsigned, is the maximum number of octets that may be contained in a single, indivisible application layer protocol data unit. The value of this property shall be greater than or equal to 50. The value of this property is also constrained by the underlying data link technology *and shall be less than or equal to the largest APDU_Length of the enabled Network Port objects used to represent the underlying data links*. See Clauses 6 through 11.

If the value of this property is not encodable in the 'Max APDU Length Accepted' parameter of a ConfirmedRequest-PDU, then the value encoded shall be the highest encodable value less than the value of this property. In such cases, a responding device may ignore the encoded value in favor of the value of this property, if it is known.

[Replace **Clause 12.11.32**, p. 203]

12.11.32 Max_Master

This property, of type Unsigned, shall be present if the device is a master node on an MS/TP network. The value of this property shall reflect the value of the Max_Master property of the Network Port object with the lowest object instance whose Network_Type is MSTP and whose Out_Of_Service is FALSE. See Clause 12.X.37.

The value of this property is a local matter if there are no MS/TP Network Port objects with Out_Of_Service set to FALSE.

If this property is writable, writing to this property shall cause the new value to take effect immediately, bypassing the activation functionality of the Network Port Object. See Clause 12.X.11 and 12.X.12.

[Replace **Clause 12.11.33**, p.203]

12.11.33 Max_Info_Frames

This property, of type Unsigned, shall be present if the device is a master node on an MS/TP network. The value of this property shall reflect the value of the Max_Info_Frames property of the Network Port object with the lowest object instance whose Network_Type is MSTP and whose Out_Of_Service is FALSE. See Clause 12.X.38.

The value of this property is a local matter if there are no MS/TP Network Port objects with Out_Of_Service set to FALSE.

If this property is writable, writing to this property shall cause the new value to take effect immediately, bypassing the activation functionality of the Network Port Object. See Clauses 12.X.11 and 12.X.12.

[Replace **Clauses 12.11.40 – 12.11.43**, p. 204]

~~**12.11.40 Slave_Proxy_Enable**~~

~~This property, of type BACnetARRAY of BOOLEAN, is an indication whether (TRUE) or not (FALSE) the device will perform Slave-Proxy functions for each of the MS/TP ports represented by each array element. The value of this property shall be retained over a device reset.~~

12.11.40 Deleted Clause

This clause has been removed.

~~**12.11.41 Manual_Slave_Address_Binding**~~

~~This property, of type BACnetLIST of BACnetAddressBinding, describes the manually configured set of slave devices for which this device is acting as a Slave Proxy as described in 16.10.2.~~

~~This property is used to manually configure a set of slave devices for which this device will be a proxy. This property allows a Slave Proxy that does not support automatic slave discovery be configured with a set of slaves for which this device will be a proxy. It also allows a Slave Proxy device to be a proxy for Slave devices that do not support the special object instance of 4194303 as described in Clause 12. The value of this property shall be retained over a device reset. When enabled, the Slave-Proxy device shall periodically check each device that is in this list, and not in the Slave_Address_Binding list, by reading the device's Protocol_Services_Supported property from the device's Device object using the ReadProperty service. If the device responds and indicates that it does not execute the Who-Is service, it shall be added to the Slave_Address_Binding property. The period at which the devices are checked is a local matter.~~

12.11.41 Deleted Clause

This clause has been removed.

~~**12.11.42 Auto_Slave_Discovery**~~

~~This property, of type BACnetARRAY of BOOLEAN, is an indication whether (TRUE) or not (FALSE) the device will perform automatic slave detection functions for each of the MS/TP ports represented by each array element. The value of this property shall be retained over a device reset.~~

~~Slave detection shall be accomplished by the proxy device using ReadProperty services to read, at a minimum, the Device object's Protocol_Services_Supported property for each MAC address on each port where Auto_Slave_Discovery for that port is TRUE. The ReadProperty service shall use the special object instance of 4194303 as described in Clause 12. If the device is found to support execution of the Who-Is service, it is ignored; otherwise, the device shall be added to the Slave_Address_Binding property. The slave detection algorithm shall be repeated periodically. The period at which it is repeated is a local matter.~~

12.11.42 Deleted Clause

This clause has been removed.

~~12.11.43 Slave_Address_Binding~~

~~This property, of type BACnetLIST of BACnetAddressBinding, describes the set of slave devices for which this device is acting as a Slave Proxy as described in Clause 16.10.2.~~

~~The set of devices described by the Slave_Address_Binding property consists of those devices described in the Manual_Slave_Address_Binding and those devices that are automatically discovered. When enabled, the Slave Proxy device shall periodically check each device in this list by reading the device's Protocol_Services_Supported property from the device's Device object using the ReadProperty service. If the device fails to respond, or indicates that it executes Who-Is, it shall be removed from the list. The period at which the devices are checked is a local matter.~~

12.11.43 Deleted Clause

This clause has been removed.

[Change **Clause 12**, p. 146]

...
Several object types defined in this clause have a property called "Reliability." This property is an enumerated datatype that may have different possible enumerations for different object types. The values defined below are a superset of all possible values of the Reliability property for all object types. The range of possible values returned for each specific object is defined in the appropriate object type definition.

NO_FAULT_DETECTED	The present value is reliable; that is, no other fault (enumerated below) has been detected.
…	
TRIPPED	The end device, such as an actuator, is not responding to commands, prevented by a tripped condition or by being mechanically held open.
ACTIVATION_FAILURE	*Activation of changes has failed.*
RENEW_FD_REGISTRATION_FAILURE	*Renewing a foreign device registration with a BBMD has failed.*
RENEW_DHCP_FAILURE	*The attempt to obtain an IP address from a DHCP server has failed.*
RESTART_AUTONEGOTIATION_FAILURE	*The auto-negotiation algorithm has failed.*
RESTART_FAILURE	*The attempt to restart the port has failed.*
PROPRIETARY_COMMAND_FAILURE	*A proprietary command has failed.*

[Change **Clause 15.5.2**, p. 549]

...
When the object-type in the Object Identifier parameter contains the value ~~'Device Object'~~ *DEVICE* and the instance in the 'Object Identifier' parameter contains the value 4194303, the responding BACnet-user shall treat the Object Identifier as if it correctly matched the local Device object. This allows the device instance of a device that does not generate I-Am messages to be determined.
When the object-type in the Object Identifier parameter contains the value NETWORK_PORT and the instance in the 'Object Identifier' parameter contains the value 4194303, the responding BACnet-user shall treat the Object Identifier as if it correctly matched the local Network Port object representing the network port through which the request was received. This allows the network port instance of the network port that was used to receive the request to be determined.

[Change **Clause 18.3**, p. 596]

18.3 Error Class - PROPERTY

...
INVALID_DATATYPE - The datatype of a property value specified in a service parameter does not match the datatype of the property referenced by the specified Property_Identifier.

INVALID_VALUE_IN_THIS_STATE *- The value specified in a service parameter is invalid in the current state of the property.*

LOGGED_VALUE_PURGED - A previously logged value was purged due to a change to the list of logged properties.
...

[Change ASN.1 production in **Clause 21**, pp. 655]

```
Error ::= SEQUENCE {
-- NOTE: The valid combinations of error-class and error-code are defined in Clause 18.
        error-class     ENUMERATED ...
        error-code      ENUMERATED { -- see below for numerical order
                        ...
                        invalid-time-stamp                      (14),
                        invalid-value-in-this-state             (138),
                        key-update-in-progress                  (100),
                        ...
                        -- numerical order reference
                        ...
                        -- see abort-security-error            (136),
                        -- see invalid-value-in-this-state     (138),
                        ...
                        }
                        -- Enumerated values 0-255 are reserved for definition by ASHRAE. Enumerated values
                        -- 256-65535 may be used by others subject to the procedures and constraints described
                        -- in Clause 23.
        }
```

[Add new productions to **Clause 21** in appropriate alphabetical position, starting p. 661]

```
BACnetBDTEntry ::= SEQUENCE {
        bbmd-address    [0] BACnetHostNPort,
        broadcast-mask  [1] OCTET STRING
        }

BACnetFDTEntry ::= SEQUENCE {
```

```
        bacnetip-address        [0] OCTET STRING,       -- the 6-octet B/IP address of the registrant
        time-to-live            [1] Unsigned16,         -- time to live in seconds at the time of registration
        remaining-time-to-live  [2] Unsigned16          -- remaining time to live in seconds, incl. grace period
        }

BACnetHostAddress ::= CHOICE {
        none            [0] NULL,
        ip-address      [1] OCTET STRING,       -- 4 octets
        name            [2] CharacterString     -- Internet host name (see RFC 1123)
        }

BACnetHostNPort ::= SEQUENCE {
        host            [0] BACnetHostAddress,
        port            [1] Unsigned16
        }

BACnetIPMode ::= ENUMERATED {
        normal          (0),
        foreign         (1),
        bbmd            (2)
        }

BACnetNetworkPortCommand :: = ENUMERATED {
        idle                    (0),
        discard-changes         (1),
        renew-fd-registration   (2),
        restart-slave-discovery (3),
        renew-dhcp              (4),
        restart-autonegotiation (5),
        disconnect              (6),
        restart-port            (7),
        …
        }
-- Enumerated values 0-127 are reserved for definition by ASHRAE. Enumerated values
-- 128-255 may be used by others subject to the procedures and constraints described
-- in Clause 23.

BACnetNetworkNumberQuality ::= ENUMERATED {
        unknown             (0),
        learned             (1),
        learned-configured  (2),
        configured          (3)
        }
...

BACnetNetworkType ::= ENUMERATED {
        ethernet        (0),
        arcnet          (1),
        mstp            (2),
        ptp             (3),
        lontalk         (4),
        bacnet-ipv4     (5),
        zigbee          (6),
        virtual         (7),
        non-bacnet      (8),
        ...
```

```
        }
-- Enumerated values 0-63 are reserved for definition by ASHRAE. Enumerated values
-- 64-255 may be used by others subject to the procedures and constraints described
-- in Clause 23.

BACnetRouterEntry ::= SEQUENCE {
        network-number          [0]     Unsigned16,
        mac-address             [1]     OCTET STRING,
        status                  [2]     ENUMERATED {
                                          available     (0),
                                          busy          (1),
                                          disconnected  (2)
                                        },
        performance-index       [3]     Unsigned8 OPTIONAL
        }

BACnetVMACEntry ::= SEQUENCE {
        virtual-mac-address     [0]     OCTET STRING, -- maximum size 6 octets
        native-mac-address      [1]     OCTET STRING
        }
```

[Change **Clause 21**, pp. 690]

```
BACnetObjectType ::= ENUMERATED { -- see below for numerical order
        ...
        multi-state-value       (19),
        network-port            (56),
        network-security        (38),
        ...
        -- see lighting-output  (54),
        -- see network-port     (56),
        ...
        }
-- Enumerated values 0-127 are reserved for definition by ASHRAE. Enumerated values
-- 128-1023 may be used by others subject to the procedures and constraints described
-- in Clause 23.

BACnetObjectTypesSupported ::= BIT STRING {
        ...
        lighting-output         (54),
        network-port            (56)
        }
```

[Change **Clause 21**, p. 694]

```
BACnetPropertyIdentifier ::= ENUMERATED { -- see below for numerical order
        ...
        allow-group-delay-inhibit       (365),
        apdu-length                     (399),
        apdu-segment-timeout            (10),
        ...
        backup-preparation-time         (339),
        bacnet-ip-address               (400),
        bacnet-ip-default-gateway       (401),
        bacnet-ip-dhcp-enable           (402),
        bacnet-ip-dhcp-lease-time       (403),
```

bacnet-ip-dhcp-lease-time-remaining *(404)*,
bacnet-ip-dhcp-server *(405)*,
bacnet-ip-dns-server *(406)*,
bacnet-ip-global-address *(407)*,
bacnet-ip-mode *(408)*,
bacnet-ip-multicast-address *(409)*,
bacnet-ip-nat-traversal *(410)*,
bacnet-ip-subnet-mask *(411)*,
bacnet-ip-udp-port *(412)*,
base-device-security-policy (327),
bbmd-accept-fd-registrations *(413)*,
bbmd-broadcast-distribution-table *(414)*,
bbmd-foreign-device-table *(415)*,
belongs-to (262),
...
change-of-state-time (16),
changes-pending *(416)*,
channel-number (366),
client-cov-increment (127),
command *(417)*,
configuration-files (154),
…
fd-bbmd-address *(418)*,
fd-subscription-lifetime *(419)*,
firmware-revision (44),
...
limit-monitoring-interval (182),
link-speed *(420)*,
link-speeds *(421)*,
link-speed-autonegotiate *(422)*,
list-of-group-members (53),
...
low-limit (59),
mac-address *(423)*,
maintenance-required (158),
...
network-access-security-policies (332),
network-interface-name *(424)*,
network-number *(425)*,
network-number-quality *(426)*,
network-type *(427)*,
node-subtype (207),
...
restore-preparation-time (341),
routing-table *(428)*,
scale (187),
...
verification-time (326),
virtual-mac-address-table *(429)*,
...

-- -numerical order reference

...
-- see egress-active (386),
-- *see apdu-length* *(399)*,
-- *see bacnet-ip-address* *(400)*,
-- *see bacnet-ip-default-gateway* *(401)*,

```
        -- see bacnet-ip-dhcp-enable                (402),
        -- see bacnet-ip-dhcp-lease-time            (403),
        -- see bacnet-ip-dhcp-lease-time-remaining  (404),
        -- see bacnet-ip-dhcp-server                (405),
        -- see bacnet-ip-dns-server                 (406),
        -- see bacnet-ip-global-address             (407),
        -- see bacnet-ip-mode                       (408),
        -- see bacnet-ip-multicast-address          (409),
        -- see bacnet-ip-nat-traversal              (410),
        -- see bacnet-ip-subnet-mask                (411),
        -- see bacnet-ip-udp-port                   (412),
        -- see bbmd-accept-fd-registrations         (413),
        -- see bbmd-broadcast-distribution-table    (414),
        -- see bbmd-foreign-device-table            (415),
        -- see changes-pending                      (416),
        -- see command                              (417),
        -- see fd-bbmd-address                      (418),
        -- see fd-subscription-lifetime             (419),
        -- see link-speed                           (420),
        -- see link-speeds                          (421),
        -- see link-speed-autonegotiate             (422),
        -- see mac-address                          (423),
        -- see network-interface-name               (424),
        -- see network-number                       (425),
        -- see network-number-quality               (426),
        -- see network-type                         (427),
        -- see routing-table                        (428),
        -- see virtual-mac-address-table            (429),
        ...
        }
-- The special property identifiers all, optional, and required are reserved for use in the
...
```

[Change **Clause 21**, p. 709]

```
BACnetReliability ::= ENUMERATED {
        no-fault-detected                (0),
        no-sensor                        (1),
        over-range                       (2),
        under-range                      (3),
        open-loop                        (4),
        shorted-loop                     (5),
        no-output                        (6),
        unreliable-other                 (7),
        process-error                    (8),
        multi-state-fault                (9),
        configuration-error              (10),
        -- enumeration value 11 is reserved for a future addendum
        communication-failure            (12),
        member-fault                     (13),
        monitored-object-fault           (14),
        tripped                          (15),
        activation-failure               (17),
        renew-dhcp-failure               (18),
        renew-fd-registration-failure    (19),
        restart-auto-negotiation-failure (20),
```

```
        restart-failure                 (21),
        proprietary-command-failure     (22),
        ...
        }
-- Enumerated values 0-63 are reserved for definition by ASHRAE. Enumerated values
-- 64-65535 may be used by others subject to the procedures and constraints described
-- in Clause 23.
```

[Change **Clause 21**, p. 707]

```
BACnetPropertyStates ::= CHOICE {
-- This production represents the possible datatypes for properties that
-- have discrete or enumerated values. The choice must be consistent with the
-- datatype of the property referenced in the Event Enrollment Object.

        boolean-value                   [0] BOOLEAN,
        ...
        lighting-transition             [40] BACnetLightingTransition,
        bacnet-ip-mode                  [45] BACnetIPMode,
        network-port-command            [46] BACnetNetworkPortCommand,
        network-type                    [47] BACnetNetworkType,
        network-number-quality          [48] BACnetNetworkNumberQuality,
        ...
        }
-- Tag values 0-63 are reserved for definition by ASHRAE. Tag values of 64-254 may be used by others to
-- accommodate vendor specific properties that have discrete or enumerated values, subject to the constraints described
-- in Clause 23.
```

[Change **Table 23-1**, p. 718]

Table 23-1. Extensible Enumerations

Enumeration Name	Reserved Range	Maximum Value
...	...	...
BACnetLightingTransition	0-63	255
BACnetNetworkPortCommand	*0-127*	*255*
BACnetNetworkType	*0-63*	*255*

[Change **Clause 12.11.44**, p. 205]

12.11.44 Last_Restart_Reason

This property, of type BACnetRestartReason, indicates the reason for the last device restart. See Clause 19.3 for a description of the restart procedure. The possible values for this property are:

UNKNOWN	The device cannot determine the cause of the last reset.
COLDSTART	A ReinitializeDevice request was received with a 'Reinitialized State of Device' of COLDSTART or the device was made to COLDSTART by some other means.
WARMSTART	A ReinitializeDevice request was received with a 'Reinitialized State of Device' of WARMSTART or the device was made to WARMSTART by some other means.
DETECTED_POWER_LOST	The device detected that incoming power was lost.
DETECTED_POWERED_OFF	The device detected that its power switch was turned off.
HARDWARE_WATCHDOG	The hardware watchdog timer reset the device.
SOFTWARE_WATCHDOG	The software watchdog timer reset the device.
SUSPENDED	The device was suspended. How the device was suspended or what it means to be suspended is a local matter.
ACTIVATE_CHANGES	*A ReinitializeDevice request was received with a 'Reinitialized State of Device' of ACTIVATE_CHANGES which caused the device to restart.*

[Change **Clause 16.4**, p. 571]

16.4 ReinitializeDevice Service

The ReinitializeDevice service is used by a client BACnet-user to instruct a remote device to reboot itself (cold start), reset itself to some predefined initial state (warm start), *to activate network port object changes*, or to control the backup or restore procedure. Resetting or rebooting a device is primarily initiated by a human operator for diagnostic purposes. Use of this service during the backup or restore procedure is usually initiated on behalf of the user by the device controlling the backup or restore. Due to the sensitive nature of this service, a password may be required by the responding BACnet-user prior to executing the service.

A BACnet device may support the ReinitializeDevice service by supporting only the restart choices COLDSTART and WARMSTART. Support for the backup and restore features of this service is claimed separately. *If the device supports a Network Port Object using the pending changes functionality, then the restart choice ACTIVATE_CHANGES shall also be supported.*

16.4.1 Structure

The structure of the ReinitializeDevice service primitives is shown in Table 16-4. The terminology and symbology used in this table are explained in Clause 5.6.

Table 16-4. Structure of ReinitializeDevice Service Primitives

Parameter Name	Req	Ind	Rsp	Cnf
Argument	M	M(=)		
Reinitialized State of Device	M	M(=)		
Password	U	U(=)		
Result (+)			S	S(=)

Result (-) Error Type			S M	S(=) M(=)

16.4.1.1 Argument

This parameter shall convey the parameters for the ReinitializeDevice confirmed service request.

16.4.1.1.1 Reinitialized State of Device

This parameter allows the client user to specify the desired state of the device after its reinitialization. The value of the parameter may be one of COLDSTART, WARMSTART, *ACTIVATE_CHANGES*, STARTBACKUP, ENDBACKUP, STARTRESTORE, ENDRESTORE, or ABORTRESTORE. WARMSTART shall mean to reboot the device and start over, retaining all data and programs that would normally be retained during a brief power outage. The precise interpretation of COLDSTART shall be defined by the vendor.

If the value of the parameter is *ACTIVATE_CHANGES or* WARMSTART and the device is not ready due to a configuration procedure in progress, the request shall be considered invalid and the responding BACnet user shall issue a Result(-) response.

If the value of the parameter is one of STARTBACKUP, ENDBACKUP, STARTRESTORE, ENDRESTORE, or ABORTRESTORE and communication has been disabled due to receipt of a DeviceCommunicationControl request with 'Enable/Disable' equal to DISABLE, the request shall be considered invalid and the responding BACnet user shall issue a Result(-) response.

The use of the backup and restore commands are defined in Clause 19.1.

16.4.1.1.2 Password

This optional parameter shall be a CharacterString of up to 20 characters. For those devices that require the password as a protection, the service request shall be denied if the parameter is absent or if the password is incorrect. For those devices that do not require a password, this parameter shall be ignored.

16.4.1.2 Result(+)

This parameter shall indicate that the service request succeeded.

16.4.1.3 Result(-)

This parameter shall indicate that the service request has failed. The reason for the failure shall be specified by the 'Error Type' parameter.

16.4.1.3.1 Error Type

This parameter consists of two component parameters: (1) the 'Error Class' and (2) the 'Error Code'. See Clause 18. The 'Error Class' and 'Error Code' to be returned for specific situations are as follows:

Situation	Error Class	Error Code
The password is invalid or absent when one is required.	SECURITY	PASSWORD_FAILURE
The device is in the process of being configured.	DEVICE	CONFIGURATION_IN_PROGRESS
Communication has been disabled due to receipt of a DeviceCommunicationControl request.	SERVICES	COMMUNICATION_DISABLED
The validation of network port object property values fails.	*PROPERTY*	*INVALID_CONFIGURATION_DATA*

16.4.2 Service Procedure

After verifying the validity of the request, including the 'Reinitialized State of Device' and 'Password' parameters, the responding BACnet-user shall pre-empt all other outstanding requests and respond with a 'Result(+)' primitive. If the request is valid and 'Reinitialized State of Device' is WARMSTART or COLDSTART, then the responding BACnet-user shall immediately proceed to perform any applicable shut-down procedures prior to reinitializing the device as specified by the requesting BACnet-user in the request.

If 'Reinitialized State of Device' is WARMSTART and the device is not ready due to its initial characterization being in progress, a 'Result (-)' response primitive shall be issued.
If 'Reinitialized State of Device' is one of STARTBACKUP, ENDBACKUP, STARTRESTORE, ENDRESTORE, or ABORTRESTORE and communication has been disabled due to receipt of a DeviceCommunicationControl request with 'Enable/Disable' equal to DISABLE, the responding BACnet user shall respond with a Result(-) primitive. Otherwise, the responding BACnet user shall behave as described in Clause 19.1.

If the password is invalid or is absent when one is required, an Error-PDU with 'error class' of SECURITY and 'error code' of PASSWORD_FAILURE shall be issued.

See Clause 12.X for a description of 'Reinitialized State of Device' when its value is ACTIVATE_CHANGES.

[Change **Clause 19.3**, p. 613]

19.3 Device Restart Procedure

When a BACnet device restarts, there are a number of different configuration items that can be lost. For example, a device need not remember which devices have subscribed to receive change-of-value notifications or to which values they have subscribed. For this reason, other devices may be interested in determining when a device has restarted. This section outlines how a device may interoperably indicate that it has restarted.

When a device is powered on, when it restarts due to a ReinitializeDevice service (*ACTIVATE_CHANGES*, COLDSTART, or WARMSTART), or when it restarts for some other reason, the device shall transmit an UnconfirmedCOVNotification request. The 'Subscriber Process Identifier' parameter shall be 0, the 'Monitored Object Identifier' parameter shall reference the Device object, the 'Time Remaining' parameter shall be 0, and the 'List of Values' parameter shall contain three values, the System_Status, the Time_Of_Device_Restart, and the Last_Restart_Reason properties of the Device object. The device shall transmit this message after the complete power-up or restart sequence has been completed so that the system-status value is accurate.

The device shall send the restart notification to each recipient in the Restart_Notification_Recipients property of the Device object.

MS/TP slave devices are not able to support this procedure, although they may support the Time_Of_Device_Restart and Last_Restart_Reason properties.

[Change **Clauses K.5.15 and K.5.16**, p. 899]

K.5.15 BIBB - Device Management-ReinitializeDevice-A (DM-RD-A)

The A device is authorized to reinitialize the B device.

BACnet Service	Initiate	Execute
ReinitializeDevice	x	

Devices claiming conformance to DM-RD-A shall be able to initiate ReinitializeDevice requests containing the Password parameter. Devices claiming conformance to DM-RD-A are only required to support the *ACTIVATE_CHANGES*, WARMSTART, and COLDSTART service choices.

K.5.16 BIBB - Device Management-ReinitializeDevice-B (DM-RD-B)

The B device performs reinitialization requests from the A device. The optional password field shall be supported.

BACnet Service	Initiate	Execute
ReinitializeDevice		x

Devices claiming conformance to DM-RD-B are only required to support the WARMSTART and COLDSTART service choices. *If the device supports a configurable Network Port Object, then it shall also support the restart choice ACTIVATE_CHANGES.*

[Change **Clause 21**, p. 649]

```
ReinitializeDevice-Request ::= SEQUENCE {
        reinitializedStateOfDevice      [0] ENUMERATED {
                                                coldstart           (0),
                                                warmstart           (1),
                                                startbackup         (2),
                                                endbackup           (3),
                                                startrestore        (4),
                                                endrestore          (5),
                                                abortrestore        (6),
                                                activate_changes    (7)
                                                },
        password                        [1] CharacterString (SIZE (1..20)) OPTIONAL
        }
```

[Change **Clause 21**, p. 709]

```
BACnetRestartReason ::= ENUMERATED {
        unknown                 (0),
        coldstart               (1),
        warmstart               (2),
        detected-power-lost     (3),
        detected-powered-off    (4),
        hardware-watchdog       (5),
        software-watchdog       (6),
        suspended               (7),
        activate-changes        (8),
        ...
        }
    -- Enumerated values 0-63 are reserved for definition by ASHRAE. Enumerated values 64-255
    -- may be used by others subject to the procedures and constraints described in Clause 23.
```

135-2012*ai*-2 Changes to Annex J for the Network Port Object

Rationale

There are several amendments required for Annex J with the introduction of the Network Port Object.

- Registration as a foreign device for non-BBMD BACnet/IP devices has been a long standing BTL requirement. This change adds this requirement to the 135 Standard.
- Addition of DNS support in Annex J.
- Clarify the error return from the Write BDT BVLL.

[Add new **Clause J.1.2.1**, p. 859]

J.1.2.2 Domain Names within B/IP Networks

B/IP devices shall be capable of resolving Internet host names to IP addresses via the use of the Domain Name Service (see RFC 1123) in the case where a DNS resolver is available. When a DNS resolver is available, then the IP address obtained via DNS name resolution shall be used in the IP address portion of B/IP addresses. The dotted-decimal form ("#.#.#.#") may also be used in the DNS name. The DNS name format shall be checked for dotted-decimal format prior to attempting name resolution.

B/IP devices shall be capable of using IP addresses in place of DNS host names when a DNS resolver is not available.

For example, a B/IP device might specify bbmd.example.ashrae.org as the BBMD with which to register as a foreign device. In this case, the B/IP device will attempt to resolve bbmd.example.ashrae.org with a DNS resolver.

The frequency that a B/IP device resolves DNS host names is a local matter.

[Change **Clause J.2.2.1**, p. 860]

J.2.2.1 Write-Broadcast-Distribution-Table: Format

The Write-Broadcast-Distribution-Table message consists of four fields:

BVLC Type:	1-octet	X'81'	BVLL for BACnet/IP
BVLC Function:	1-octet	X'01'	Write-Broadcast-Distribution-Table
BVLC Length:	2-octets	L	Length L, in octets, of the BVLL message
List of BDT Entries:	N*10-octets		

N indicates the number of entries in the BDT. Each BDT entry consists of the 6-octet B/IP address of a BBMD followed by a 4-octet field called the broadcast distribution mask that indicates how broadcast messages are to be distributed on the IP subnet served by the BBMD. See J.4.3.2.

Prior to the introduction of the Network Port object in Protocol_Revision 17, this message was the interoperable means of updating BDTs. That function is now performed by writes to the Network Port object.

[Change **Clause J.4.3**, p. 864]

J.4.3 BBMD Concept

Each IP subnet that contains B/IP devices that do not register as a foreign device and that are part of a B/IP network comprised of two or more IP subnets ~~Each IP subnet that is part of a B/IP network comprised of two or more subnets~~ shall have at least one BBMD. Each BBMD shall possess a table called a Broadcast Distribution Table (BDT). If there are two or more BBMDs on a single subnet, their BDTs shall not contain any common entries in order to avoid a broadcast forwarding loop. The BDT determines which remote IP subnets receive forwarded BACnet broadcasts. To reduce BACnet broadcast traffic, it is possible to configure

the BDT to forward broadcasts only to IP subnets where they are required. ~~If the BBMD has also been designated to register foreign devices as described below, it shall also possess a Foreign Device Table (FDT).~~ A BBMD shall be able to be configured to accept Foreign Device registrations, shall support the two-hop broadcast distribution method, ~~and~~ shall support the execution of all ~~BDT and~~ FDT read and write messages defined in Clause J.2, *and shall support execution of the Read-Broadcast-Distribution-Table message as defined in Clause J.2.* Support for the one-hop broadcast distribution method is optional.

A B/IP device which cannot be configured as a BBMD shall be capable of registering as a foreign device with a BBMD.

[Change **Clause J.4.4**, p. 865]

J.4.4 BBMD Configuration

The configuration of the BACnet-related capability of a BBMD shall consist of supplying it with a BDT. The table ~~may be supplied by local means or by means of the BVLL Write-Broadcast-Distribution-Table message~~ *shall be supplied by writing to the BBMD_Broadcast_Distribution_Table property of the Network Port Object which represents this B/IP port.*

[Change **Clause J.4.5**, p. 865]

J.4.5 BBMD Operation - Broadcast Distribution

~~Upon receipt of a BVLL Write-Broadcast-Distribution-Table message, a BBMD shall attempt to create or replace its BDT, depending on whether or not a BDT has previously existed. If the creation or replacement of the BDT is successful, the BBMD shall return a BVLC-Result message to the originating device with a result code of X'0000'. Otherwise, the BBMD shall return a BVLC-Result message to the originating device with a result code of X'0010' indicating that the write-attempt has failed.~~ *Upon receipt of a BVLL Write-Broadcast-Distribution-Table message, a BBMD shall return a BVLC-Result message to the originating device with a result code of X'0010' indicating that the Write-Broadcast-Distribution BVLL message is not supported. Prior to the introduction of the Network Port object in Protocol_Revision 17, this message was the interoperable means of updating BDTs. That function is now performed by writes to the Network Port object.*

Upon receipt of a BVLL Read-Broadcast-Distribution-Table message, a BBMD shall load the contents of its BDT into a BVLL Read-Broadcast-Distribution-Table-Ack message and send it to the originating device. *If a table entry contains a host name, then the corresponding entry in the Read-Broadcast-Distribution-Table-Ack message shall contain the IP address of the resolved host name or X'000000000000' to indicate that the host name has not been resolved.* If the BBMD is unable to perform the read of its BDT, it shall return a BVLC-Result message to the originating device with a result code of X'0020' indicating that the read attempt has failed.

135-2012*ai*-3 Changes to 135-2012*al* for the Network Port Object

Rationale

Addendum 135-2012*al* added BIBBs for BBMD configuration. The A device BIBB is changed to make it clear that the changes to the Write-Broadcast-Distribution-Table service do not change the need for the support of the service in the BIBB due to the need for interoperability with older implementations.

[Change **K.5.X2** in **Addendum 135-2012*al***, p. 8]
[Italics shows text added to Clause K.5.X2 in Addendum 135-2012*al*]

K.5.X2 BIBB - Network Management - BBMD Configuration - A (NM-BBMDC-A)

The A device is able to query and change the configuration of BBMDs.

BACnet Virtual Link Layer Message	Initiate	Execute
Write-Broadcast-Distribution-Table	x	
Read-Broadcast-Distribution-Table	x	
Read-Foreign-Device-Table	x	
Delete-Foreign-Device-Table-Entry	x	

Devices claiming conformance to this BIBB are required to initiate Write-Broadcast-Distribution-Table for interoperability with devices implementing Protocol_Revisions older than 17.

[Add a new entry to **History of Revisions**, p. 1006]

(This History of Revisions is not part of this standard. It is merely informative and does not contain requirements necessary for conformance to the standard.)

HISTORY OF REVISIONS

Protocol		***Summary of Changes to the Standard***
Version	***Revision***	
…	…	**…**
1	17	**Addendum *ai* to ANSI/ASHRAE 135-2012** Approved by the ASHRAE Standards Committee Xxxx xx, 2015; by the ASHRAE Board of Directors Xxxx xx, 2015; and by the American National Standards Institute Xxxx xx, 2015. 1. Add Network Port Object Type 2. Changes to Annex J for the Network Port Object 3. Changes to 135-2012*al* for the Network Port Object

POLICY STATEMENT DEFINING ASHRAE'S CONCERN FOR THE ENVIRONMENTAL IMPACT OF ITS ACTIVITIES

ASHRAE is concerned with the impact of its members' activities on both the indoor and outdoor environment. ASHRAE's members will strive to minimize any possible deleterious effect on the indoor and outdoor environment of the systems and components in their responsibility while maximizing the beneficial effects these systems provide, consistent with accepted Standards and the practical state of the art.

ASHRAE's short-range goal is to ensure that the systems and components within its scope do not impact the indoor and outdoor environment to a greater extent than specified by the Standards and Guidelines as established by itself and other responsible bodies.

As an ongoing goal, ASHRAE will, through its Standards Committee and extensive Technical Committee structure, continue to generate up-to-date Standards and Guidelines where appropriate and adopt, recommend, and promote those new and revised Standards developed by other responsible organizations.

Through its *Handbook*, appropriate chapters will contain up-to-date Standards and design considerations as the material is systematically revised.

ASHRAE will take the lead with respect to dissemination of environmental information of its primary interest and will seek out and disseminate information from other responsible organizations that is pertinent, as guides to updating Standards and Guidelines.

The effects of the design and selection of equipment and systems will be considered within the scope of the system's intended use and expected misuse. The disposal of hazardous materials, if any, will also be considered.

ASHRAE's primary concern for environmental impact will be at the site where equipment within ASHRAE's scope operates. However, energy source selection and the possible environmental impact due to the energy source and energy transportation will be considered where possible. Recommendations concerning energy source selection should be made by its members.

ASHRAE · 1791 Tullie Circle NE · Atlanta, GA 30329 · www.ashrae.org

About ASHRAE

ASHRAE, founded in 1894, is a global society advancing human well-being through sustainable technology for the built environment. The Society and its members focus on building systems, energy efficiency, indoor air quality, refrigeration, and sustainability. Through research, Standards writing, publishing, certification and continuing education, ASHRAE shapes tomorrow's built environment today.

For more information or to become a member of ASHRAE, visit www.ashrae.org.

To stay current with this and other ASHRAE Standards and Guidelines, visit www.ashrae.org/standards.

Visit the ASHRAE Bookstore

ASHRAE offers its Standards and Guidelines in print, as immediately downloadable PDFs, on CD-ROM, and via ASHRAE Digital Collections, which provides online access with automatic updates as well as historical versions of publications. Selected Standards and Guidelines are also offered in redline versions that indicate the changes made between the active Standard or Guideline and its previous version. For more information, visit the Standards and Guidelines section of the ASHRAE Bookstore at www.ashrae.org/bookstore.

IMPORTANT NOTICES ABOUT THIS STANDARD

To ensure that you have all of the approved addenda, errata, and interpretations for this Standard, visit www.ashrae.org/standards to download them free of charge.

Addenda, errata, and interpretations for ASHRAE Standards and Guidelines are no longer distributed with copies of the Standards and Guidelines. ASHRAE provides these addenda, errata, and interpretations only in electronic form to promote more sustainable use of resources.

1/15