



# ASHRAE<sup>®</sup> STANDARD

## BACnet<sup>™</sup> - A Data Communication Protocol for Building Automation and Control Networks

©2001 American Society of Heating, Refrigerating and Air-Conditioning Engineers, Inc.

Approved by the ASHRAE Standards Committee June 23, 2001; by the ASHRAE Board of Directors June 28, 2001; and by the American National Standards Institute September 7, 2001.

This standard is under continuous maintenance 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 ASHRAE's Internet Home Page, <http://www.ashrae.org>, or in paper form from the Manager of Standards. The latest edition of an ASHRAE Standard and printed copies of a public review draft may be purchased from ASHRAE Customer Service, 1791 Tullie Circle, NE, Atlanta, GA 30329-2305. E-mail: [orders@ashrae.org](mailto:orders@ashrae.org). Fax: 404-321-5478. Telephone: 404-636-8400 (worldwide), or toll free 1-800-527-4723 (for orders in U.S. and Canada).

**AMERICAN SOCIETY OF HEATING,  
REFRIGERATING AND  
AIR-CONDITIONING ENGINEERS, INC.**  
1791 Tullie Circle, NE · Atlanta GA 30329-2305

**ASHRAE STANDING STANDARD PROJECT COMMITTEE 135**  
**Cognizant TC: TC 1.4, Control Theory and Applications**  
**SPLS Liaison: Ronald E. Jarnagin**

\*Steven T. Bushby, *Chairman*  
William O. Swan III, *Vice-Chairman*  
\*Keith A. Corbett, *Secretary*  
\*Barry B. Bridges  
\*A. J. Capowski  
\*Jeffery Cosiol  
\*Thomas S. Ertsgaard

\*Daniel P. Giorgis  
\*Stephen Karg  
\*J. D. Ljungquist  
\*Jerald P. Martocci  
\*Carl Neilson  
\*Mark A. Railsback  
\*David Robin

\*Daniel A. Traill  
\*Grant N. Wichenco  
Dana R. Epperson  
Winston I. Hetherington  
David J. Branson  
Kevin G. Sweeney

*\*Denotes members of voting status this standard was approved for publication.*

---

**ASHRAE STANDARDS COMMITTEE 2000-2001**

Martha J. Hewett, *Chairman*  
Nance C. Lovvorn, *Vice-Chairman*  
Van D. Baxter  
Dean S. Borges  
Waller S. Clements  
Piotr A. Domanski  
Richard A. Evans  
John F. Hogan  
Ronald E. Jarnagin

David E. Knebel  
Frederick H. Kohloss  
William J. Landman  
Rodney H. Lewis  
Ross D. Montgomery  
Davor Novosel  
Joseph A. Pietsch  
James A. Ranfone  
Michael Tavares

Steven T. Taylor  
James K. Vallort  
Thomas E. Watson  
Bruce A. Wilcox  
J. Richard Wright  
Gerald C. Groff, BOD ExO  
William J. Buck, CO

Claire B. Ramspeck, Manager of Standards

---

**SPECIAL NOTE**

This American National Standard (ANS) is a national voluntary consensus standard developed under the auspices of the American Society of Heating, Refrigerating and Air-Conditioning Engineers (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 members may or may not be members of ASHRAE, 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 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,
- d. permission to reprint portions of the Standard.

**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 which may serve to guide the industry. The creation of ASHRAE Standards is determined by the need for them, and conformance to them is completely voluntary.

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

**DISCLAIMER**

ASHRAE uses its best efforts to promulgate standards 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, designed, installed, or operated in accordance with ASHRAE's Standards or Guidelines or that any tests conducted under its standards will be nonhazardous or free from risk.

## Foreword

The purpose of this addendum is to add a number of independent substantive changes to the BACnet standard. These modifications are the result of change proposals made pursuant to ASHRAE continuous maintenance procedures and of deliberations within Standing Standard Project Committee 135. The changes and their rationale are summarized below.

- 135e-1. Define the PTP connection status when the half-router can and cannot re-establish the connection, p. 1.

The disconnection procedure in the prior PTP state machine does not provide a way to distinguish between circumstances where the router is able to re-establish the connection and circumstances where it cannot. The latter case can occur, for example, when the PTP connection uses a telephone line and one of the half-routers is not configured to call the other. This change provides a way for a half-router to indicate that a remote network is "permanently" unreachable because the half-router has no means with which to re-establish the connection.

- 135e-2. Add Object Profiles and Extensions, p. 2.

BACnet provides mechanisms for extending the capabilities of the standard by adding new properties to standard objects, creating new objects, and creating new services using the ConfirmedPrivateTransfer and UnconfirmedPrivateTransfer services. The mechanisms for making these extensions are defined in Clause 23. Registered vendor identifiers are used to avoid conflicts between extensions made by different organizations.

There is a need to provide a way to make information about the semantics of extensions to standard object types and new object type definitions readily available. There is also a need to be able to verify claims of conformance to "well-known" extensions. A new concept called an "object profile" is defined to achieve these aims. Object Profiles also provide a mechanism that can be used in the future to make information about extensions to the standard machine-readable so that BACnet devices can discover and make use of them in an automated fashion.

An "Object Profile" is defined as a group of properties and/or behaviors either as an extension to a standard object, or as a stand-alone object definition.

To support these profiles, a new standard optional property, "Profile\_Name" is added to every BACnet object type. This property indicates the object profile to which the object claims to conform. To ensure uniqueness, a profile name begins with the vendor identifier belonging to the defining organization. (Organizations can be ASHRAE, other standards bodies, industry or academic consortiums, or individual vendors.)

Initially, the definition of the profiles would be published in a human language form in the style of Clause 12. In the future, it would be highly desirable to have machine-readable definitions available as well. The definition of that machine-readable form is beyond the scope of this addendum and will be addressed separately.

- 135e-3. Add the capability for devices to advertise the maximum number of segments of a segmented APDU that they can receive, p. 4.

BACnet provides a segmentation mechanism for use when conveying messages too long to fit into a single transmission. The BACnet-Confirmed-Request-PDU has a field in the header that indicates whether or not a segmented response can be accepted. However, the responding device has no way to determine how many segments can be accepted. If a segmented response is required, it begins transmitting segments until either the message is complete or until the requester aborts.

This addition to the standard provides a solution for this problem by using a reserved field in the BACnet-Confirmed-Request-PDU header to indicate how many segments can be accepted. This solution allows a requesting device to dynamically allocate buffers for each request and convey that dynamic information to the responding device. It also allows an efficient response to be conveyed without any prior contact with the requesting device.

A similar problem exists when making a segmented request. The requesting device has no way to know how many segments the responding device is likely to accept. The addition of a Max\_Segments\_Accepted property to the Device object solves this problem.

In the following document, language to be added to existing clauses of ANSI/ASHRAE 135-1995 and Addenda is indicated through the use of *italics* while deletions are indicated by ~~strike through~~. Where entirely new subclauses are proposed to be added, plain type is used throughout.

## **135e-1. Define the PTP connection status when the half-router can and cannot re-establish the connection.**

[Change 6.7.2.1.2, p. 70]

### **6.7.2.1.2 Initiating/Answering Half-Router Procedures**

Upon receipt of a Disconnect-Connection-To-Network message, a half-router shall disconnect the PTP connection as specified in Clause 10. When the connection is terminated, the half-router shall adjust its routing table to indicate that any DNETs accessible from the previously connected half-router have a "reachability status" that is "~~temporarily unreachable.~~ unreachable" if the half-router is able to re-establish the connection or "permanently unreachable" if the half-router is unable to re-establish the connection.

[Change 6.7.2.2, p. 70]

### **6.7.2.2 Timed Disconnection of a PTP Connection**

If the activity timer ( $T_{\text{active}}$ ) expires, a half-router shall disconnect the PTP connection as specified in Clause 10. When the connection is terminated, the half-router shall adjust its routing table to indicate that any DNETs accessible from the previously connected half-router have a "reachability status" that is "~~temporarily unreachable.~~ unreachable" if the half-router is able to re-establish the connection or "permanently unreachable" if the half-router is unable to re-establish the connection.

## 135e-2. Add Object Profiles and Extensions.

[Add a new definition in 3, p. 4, and renumber subsequent definitions]

**3.2.36 object profile:** an object profile is a means of defining objects beyond those defined in Clause 12. A profile defines the set of properties, behavior, and/or requirements for a proprietary object, or for proprietary extensions to a standard object.

[Add a new Profile\_Name property to each table that summarizes the properties of standard object types. The tables affected are:

<b>Table 12-1</b> , p. 138,	<b>Table 12-11</b> , p. 176,	<b>Table 12-21</b> , p. 208,
<b>Table 12-2</b> , p. 143,	<b>Table 12-12</b> , p. 181,	<b>Table 12-22</b> , p. 213,
<b>Table 12-3</b> , p. 148,	<b>Table 12-14</b> , p. 186,	<b>Table 12-X</b> (135b-7), p. 14,
<b>Table 12-4</b> , p. 153,	<b>Table 12-15</b> , p. 188,	<b>Table 12-X</b> (135b-8), p. 21,
<b>Table 12-6</b> , p. 159,	<b>Table 12-16</b> , p. 190,	<b>Table 12-23</b> (135b-15), p. 47,
<b>Table 12-8</b> , p. 165,	<b>Table 12-17</b> , p. 197,	<b>Table 12-X</b> (proposed 135c-1), p. 2,
<b>Table 12-9</b> , p. 170,	<b>Table 12-18</b> , p. 201,	<b>Table 12-X</b> (proposed 135c-2), p. 13.]
<b>Table 12-10</b> , p. 173,	<b>Table 12-19</b> , p. 205,	

Property Identifier	Property Datatype	Conformance Code
...	...	...
<i>Profile_Name</i>	<i>CharacterString</i>	<i>O</i>

[Add a new section to the end of each standard object type description that describes the Profile\_Name property. The new clauses are as follows ("x" indicates that the exact clause number will be determined at the time of final publication and could be affected by the outcome of other addenda currently in public review.):

<b>12.1.27</b> (12.1.26 added by 135b-3), p. 142,	<b>12.13.37</b> (12.13.36 added by 135b-3), p. 196,
<b>12.2.28</b> (12.2.27 added by 135b-3), p. 147,	<b>12.14.21</b> (12.14.20 added by 135b-3), p. 200,
<b>12.3.24</b> (12.3.23 added by 135b-3), p. 152,	<b>12.15.22</b> (12.15.21 added by 135b-3), p. 204,
<b>12.4.26</b> (12.4.24 added by 135b-3), p. 158,	<b>12.16.9</b> , p. 207,
<b>12.5.30</b> (12.5.29 added by 135b-3), p. 164,	<b>12.17.14</b> , p. 212,
<b>12.6.28</b> (12.6.27 added by 135b-3), p. 169,	<b>12.18.11</b> , p. 215,
<b>12.7.7</b> , p. 170,	<b>12.X.22</b> (135b-7), p. 18,
<b>12.8.10</b> , p. 175,	<b>12.X.16</b> (135b-8), p. 24,
<b>12.9.x</b> , p. 180,	<b>12.X.27</b> (135b-15), p. 51,
<b>12.10.18</b> (12.10.17 added by 135b-3), p. 185,	<b>12.X.x</b> (proposed 135c-1), p. 7, and
<b>12.11.12</b> (12.11.11 added by 135b-12), p. 187,	<b>12.X.x</b> , (proposed 135c-2), p. 17.]
<b>12.12.7</b> , p. 189,	

### 12.xx.yy 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 the following production in **21**, p. 375-378]

```
BACnetPropertyIdentifier ::= SEQUENCE {  
    ...  
    process-identifier          (89),  
    profile-name                (168),  
    program-change             (90),  
    ...  
    -- see profile-name        (168)  
}
```

[Add to **23.3**, p. 392]

### **23.3 Adding Proprietary Properties to a Standardized Object**

BACnet defines a set of standard objects, each with a set of standard properties that can be accessed and manipulated with BACnet services. BACnet allows a vendor to add proprietary properties to extend the capabilities of a standard object. Proprietary properties receive the same support from BACnet services as standard properties and therefore can be accessed and manipulated in a manner identical to standard properties.

*Objects may indicate conformance to an object profile by use of the Profile\_Name property.*

If a proprietary property is to be a commandable property...

[Add to **23.4**, p. 392]

### **23.4 Adding Proprietary Object Types to BACnet**

To accommodate building applications where the defined set of standardized objects is not adequate, BACnet allows a vendor to add proprietary object types. Standard object types shall be used when possible. To enhance extensibility, BACnet provides the same support for proprietary objects as for standard objects.

*Objects may indicate conformance to an object profile by use of the Profile\_Name property.*

[Append the following to each object type structure in Annex C, p. 411-418, and to the new descriptions added to Annex C in the following Addenda:

135b-7, p. 22

135b-8, p. 27

135b-15, p. 58

135c-1, p. 11, and

135c-2, p. 19]

```
...,  
profile-name    [168] CharacterString OPTIONAL  
}
```

### 135e-3. Add the capability for devices to advertise the maximum number of segments of an APDU that they can receive.

[Add a new transition to 5.4.4.1, preceding SendConfirmedSegmented, p. 27]

#### *CannotSend*

*If CONF\_SERV.request is received from the local application program and the length of the APDU is greater than maximum-transmittable-length as determined according to 5.2.1 and the Max\_Segments\_Accepted property of the destination's Device Object is known and the total APDU cannot be transmitted without exceeding the maximum number of segments accepted,*

*then send an ABORT.indication to the local application program and enter the IDLE state.*

[Change 5.4.4.1, SendConfirmedSegmented, p. 27]

#### SendConfirmedSegmented

If CONF\_SERV.request is received from the local application program and the length of the APDU is greater than maximum-transmittable-length as determined according to 5.2.1, *and the Max\_Segments\_Accepted property of the destination's Device Object is not known, or Max\_Segments\_Accepted is known and the total APDU can be transmitted without exceeding the maximum number of segments accepted,*

then assign an 'invokeID' to this transaction; set SentAllSegments to FALSE; set RetryCount to zero; set SegmentRetryCount to zero; set InitialSequenceNumber to zero; set ProposedWindowSize to whatever value is desired; set ActualWindowSize to 1; start SegmentTimer; issue an N-UNITDATA.request with 'data\_expecting\_reply' = TRUE to transmit a BACnet-Confirmed-Request-PDU containing the first segment of the message, with 'segmented-message' = TRUE, 'more-follows' = TRUE, 'sequence-number' = zero, and 'proposed-window-size' = ProposedWindowSize; and enter the SEGMENTED\_REQUEST state to await an acknowledgment. (The method used to determine ProposedWindowSize is a local matter, except that the value shall be in the range 1 to 127, inclusive.)

[Change 5.4.5.3, CannotSendSegmentedComplexACK transition, p. 35]

#### CannotSendSegmentedComplexACK

If a CONF\_SERV.response(+) is received from the local application program, which is to be conveyed via a BACnet-ComplexACK-PDU, and the length of the APDU is greater than maximum-transmittable-length as determined according to 5.2.1, and either

- (a) this device does not support the transmission of segmented messages or
- (b) the client will not accept a segmented response (the 'segmented-response-accepted' parameter in the BACnet-ConfirmedRequest-PDU is FALSE), *or*
- (c) *the client's max-segments-accepted parameter in the BACnet-ConfirmedRequest-PDU is fewer than required to transmit the total APDU,*

then, *for case (a) and (b),* issue an N-UNITDATA.request with 'data\_expecting\_reply' = FALSE to transmit a BACnet-Abort-PDU with 'server' = TRUE and 'abort-reason' = SEGMENTATION\_NOT\_SUPPORTED *or for case (c) issue BUFFER\_OVERFLOW,* and enter the IDLE state.



[Append entry to Table 12-11, p. 176]

Property Identifier	Property Datatype	Conformance Code
...	...	...
<i>Max_Segments_Accepted</i>	<i>Unsigned</i>	<i>O<sup>2</sup></i>

[Add 12.9.x, a new section at the end of the Device object Description that describes the Max\_Segments\_Accepted property]

### 12.9.x Max\_Segments\_Accepted

The Max\_Segments\_Accepted property, of type Unsigned, shall indicate the maximum number of segments of an APDU that this device will accept.

[Change 20.1.2, p. 323]

### 20.1.2 BACnet-Confirmed-Request-PDU

The BACnet-Confirmed-Request-PDU is used to convey the information contained in confirmed service request primitives.

```

BACnet-Confirmed-Request-PDU ::= SEQUENCE {
    pdu-type                [0] Unsigned (0..15), -- 0 for this PDU type
    segmented-message       [1] BOOLEAN,
    more-follows            [2] BOOLEAN,
    segmented-response-accepted [3] BOOLEAN,
    reserved                [4] Unsigned (0..31), -- must be set to zero
    max-segments-accepted   [5] Unsigned (0..7), -- as per 20.1.2.4
    max-APDU-length-accepted [6] Unsigned (0..15), -- as per 20.1.2.4 20.1.2.5
    invokeID                [7] Unsigned (0..255),
    sequence-number         [8] Unsigned (0..255) OPTIONAL, -- only if segmented msg
    proposed-window-size    [9] Unsigned (1..127) OPTIONAL, -- only if segmented msg
    service-choice          [10] BACnetConfirmedServiceChoice,
    service-request         [11] BACnet-Confirmed-Service-Request
    -- Context-specific tags 0..11 are NOT used in header encoding
}

```

[Renumber clauses 20.1.2.4 through 20.1.2.10 to be 20.1.2.5 through 20.1.2.11, then add new 20.1.2.4, p. 323]

#### 20.1.2.4 max-segments-accepted

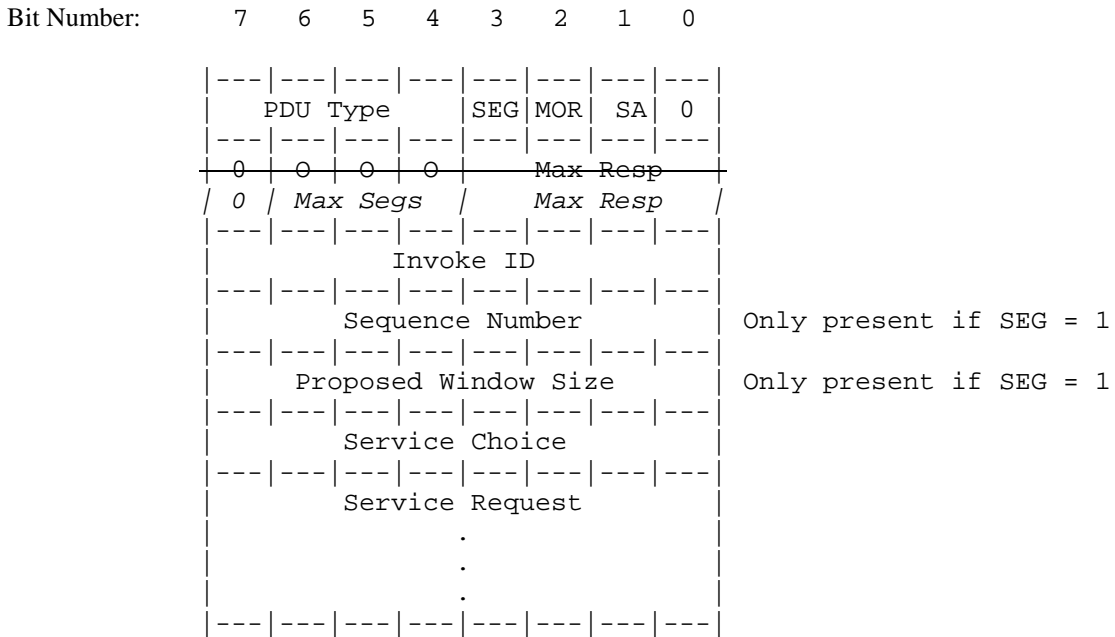
This optional parameter specifies the maximum number of segments that the device will accept. This parameter is included in the confirmed request so that the responding device may determine how to convey its response. The parameter shall be encoded as follows:

B'000'	Unspecified number of segments accepted.
B'001'	2 segments accepted.
B'010'	4 segments accepted.
B'011'	8 segments accepted.
B'100'	16 segments accepted.
B'101'	32 segments accepted.
B'110'	64 segments accepted.
B'111'	Greater than 64 segments accepted.

[Change 20.1.2.11, p. 325]

### 20.1.2.11 Format of the BACnet-Confirmed-Request-PDU

The format of the BACnet-Confirmed-Request-PDU is:



The PDU fields have the following values:

- PDU Type = 0 (BACnet-Confirmed-Service-Request-PDU)
- SEG = 0 (Unsegmented Request)
- = 1 (Segmented Request)
- MOR = 0 (No More Segments Follow)
- = 1 (More Segments Follow)
- SA = 0 (Segmented Response not accepted)
- = 1 (Segmented Response accepted)
- Max Segs = (0..7) (Number of response segments accepted per 20.1.2.4)
- Max Resp = (0..15) (Size of Maximum APDU accepted per 20.1.2.4 20.1.2.5)
- Invoke ID = (0..255)
- Sequence Number = (0..255) Only present if SEG = 1
- Proposed Window Size = (1..127) Only present if SEG = 1
- Service Choice = BACnetConfirmedServiceChoice
- Service Request = Variable Encoding per 20.2.

Bits shown in the diagram as '0' shall be set to zero. These bits are currently unused and are reserved by ASHRAE.

[Change the following production in **21**, p. 348]

```

BACnet-Confirmed-Request-PDU ::= SEQUENCE {
    pdu-type                [0] Unsigned (0..15), -- 0 for this PDU type
    segmented-message       [1] BOOLEAN,
    more-follows            [2] BOOLEAN,
    segmented-response-accepted [3] BOOLEAN,
    reserved                [4] Unsigned (0..31), -- must be set to zero
    max-segments-accepted   [5] Unsigned (0..7), -- as per 20.1.2.4
    max-APDU-length-accepted [6] Unsigned (0..15), -- as per 20.1.2.4 20.1.2.5
    invokeID                [67] Unsigned (0..255),
    sequence-number         [78] Unsigned (0..255) OPTIONAL, -- only if segmented msg
    proposed-window-size    [89] Unsigned (1..127) OPTIONAL, -- only if segmented msg
    service-choice          [910] BACnetConfirmedServiceChoice,
    service-request         [1011] BACnet-Confirmed-Service-Request
    -- Context-specific tags 0..1011 are NOT used in header encoding
}

```

[Change the following production in **21**, p. 375-378 – note that 135b-8 changed the highest enumeration from 123 to 151 and that 135c-1, currently under consideration, proposes to change 151 to 166; 168 is assigned in 135e-2]

```

BACnetPropertyIdentifier ::= ENUMERATED {
    ...
    max-pres-value          (65),
    max-segments-accepted   (167),
    minimum-off-time        (66),
    ...
    weekly-schedule         (123),
    ...
    -- max-segments-accepted (167),
}
...
-- Enumerated values 0-511 are reserved for definition by ASHRAE. Enumerated values 512-4194303 may be used by
-- others subject to the procedures and constraints described in Clause 23. The highest enumeration used in this version
-- is 464 168.

```

[Change the following production in **Annex C**, p. 414]

```

DEVICE ::= SEQUENCE {
    ...
    device-address-binding [30] SEQUENCE OF BACnetAddressBinding_BACnetAddressBinding_
    max-segments-accepted [167] Unsigned
}

```

[Change the following example in **Annex D.9**, p. 425]

### D.9 Examples of a Device Object

Example 1: A “sophisticated” BACnet device.

```

...
Property: Segmentation_Supported = SEGMENTED_BOTH
Property: Max_Segments_Accepted = 16
Property: VT_Classes_Supported (DEFAULT-TERMINAL, VT100)
...

```

[Add 5.2.1.3, p. 19]

### **5.2.1.3 Maximum Segments Accepted**

The maximum number of segments transmitted in a Confirmed-Request or ComplexACK message shall be the smallest of:

- (a) the maximum number of segments transmittable by a device, which may be restricted by local limitations and is a local matter;
- (b) the maximum number of segments accepted by the remote peer device.

If the sending device is the requesting BACnet-user, i.e., the message to be sent is a Confirmed-Request, then the maximum number of segments accepted by the remote peer device is specified in the Max\_Segments\_Accepted property of the remote peer's Device object.

If the sending device is not the requesting BACnet-user, i.e., the message to be sent is a Complex-ACK, then the maximum number of segments accepted by the remote peer device is specified in the 'Max Segments Accepted' parameter of the BACnet-Confirmed-Request-PDU for which this is a response.

## HISTORY OF REVISIONS

<i>Protocol</i>		<i>Summary of Changes to the Standard</i>
<i>Version</i>	<i>Revision</i>	
<b>1</b>	<b>NA</b>	ANSI/ASHRAE 135-1995
<b>1</b>	<b>NA</b>	Addendum <i>a</i> to ANSI/ASHRAE 135-1995 1. Add Annex J - BACnet/IP and supporting definitions
<b>1</b>	<b>1</b>	Addendum <i>b</i> to ANSI/ASHRAE 135-1995 1. Inconsistencies are eliminated in the definitions of the Analog and Binary Value object types 2. Any device that receives and executes UnconfirmedEventNotification service requests must support programmable process identifiers 3. Modify each event-generating object type to contain the last timestamp for each acknowledgeable transition 4. Modify the Notification Class object by requiring that the 'Notification Class' property be equivalent to the instance number of the Notification Class object 5. Modify the Event Notification services to make the 'To State' parameter mandatory for notifications of type ACK_NOTIFICATION 6. A new BACnetDeviceObjectPropertyReference production is added and its use in the Event Enrollment and Schedule object types is specified 7. Add a Multi-state Value object type 8. Add an Averaging object type 9. Change all 'Process Identifier' properties and parameters to Unsigned32 10. Change the Multi-state Input object type to correct flaws related to fault detection and reporting and achieve consistency with the proposed Multi-state Value object type 11. Add a Protocol_Revision property to the Device object type 12. The File object type is changed to allow truncation and partial deletion operations 13. A new ReadRange service is added to permit reading a range of data items from a property whose datatype is a list or array of lists 14. A new UTCTimeSynchronization service is introduced and related changes are made to properties in the Device object type 15. Add a Trend Log object type 16. The UnconfirmedCOVNotification service is extended to allow notifications without prior subscription as a means of distributing globally important data to a potentially large number of recipients 17. Add eight new BACnet engineering units.
<b>1</b>	<b>2</b>	Addendum <i>c</i> to ANSI/ASHRAE 135-1995 1. Add a new Life Safety Point object type that represents the characteristics of initiating and indicating devices in the fire, life safety, and security applications 2. Add a new Life Safety Zone object type that represents the

		<p>characteristics associated with an arbitrary group of BACnet Life Safety Point and Life Safety Zone objects</p> <ol style="list-style-type: none"> <li>3. Add functionality to the existing BACnet alarm and event features needed to support the Life Safety Point and Life Safety Zone object types</li> <li>4. Add a new LifeSafetyOperation service that provides silence and reset capabilities needed for life safety systems</li> <li>5. Add a new subclause to 16 to describe the use of existing BACnet services to provide backup and restore capability</li> <li>6. Define a new service, SubscribeCOVProperty, to allow COV notifications for arbitrary properties of an object with subscriber-specified COV increments</li> <li>7. Add Vendor ID to proprietary MS/TP frames</li> <li>8. Add a new service, GetEventInformation, that provides enough information to acknowledge alarms</li> </ol>
<b>1</b>	<b>2</b>	<p>Addendum <i>d</i> to ANSI/ASHRAE 135-1995</p> <ol style="list-style-type: none"> <li>1. Replace Clause 22 with a new clause entitled "Conformance and Specification".</li> <li>2. Update Annex A, "Protocol Implementation Conformance Statement".</li> <li>3. Add a new Annex K entitled "BACnet Interoperability Building Blocks (BIBBs)".</li> <li>4. Add a new Annex L entitled "Descriptions and Profiles of Standardized BACnet Devices".</li> </ol>
<b>1</b>	<b>2</b>	<p>Addendum <i>e</i> to ANSI/ASHRAE 135-1995</p> <ol style="list-style-type: none"> <li>1. Define the PTP connection status when the half-router can and cannot re-establish the connection.</li> <li>2. Add Object Profiles and Extensions.</li> <li>3. Add the capability for devices to advertise the maximum number of segments of a segmented APDU that they can receive.</li> </ol>

NA = Not Applicable because the Protocol\_Revision property was first defined in Addendum *b*