

ANSI/ASHRAE Addendum *b* to
ANSI/ASHRAE Standard 135-2001



ASHRAE[®] STANDARD

BACnet[®]—A Data Communication Protocol for Building Automation and Control Networks

Approved by the ASHRAE Standards Committee on January 25, 2003; by the ASHRAE Board of Directors on January 30, 2003; and by the American National Standards Institute on April 3, 2003.

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 are given at the back of this document and 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. Fax: 404-321-5478. Telephone: 404-636-8400 (worldwide), or toll free 1-800-527-4723 (for orders in U.S. and Canada).

©Copyright 2003 American Society of Heating, Refrigerating and Air-Conditioning Engineers, Inc.

ISSN 1041-2336



**AMERICAN SOCIETY OF HEATING,
REFRIGERATING AND
AIR-CONDITIONING ENGINEERS, INC.**

1791 Tullie Circle, NE • Atlanta, GA 30329

ASHRAE Standard Project Committee 135
Cognizant TC: TC 1.4, Control Theory and Applications
SPLS Liaison: Steven T. Taylor

Steven T. Bushby, *Chair**
William O. Swan, III, *Vice-Chair*
Carl Neilson, *Secretary**
Barry B. Bridges*
A.J. Capowski*
Keith A. Corbett*

Jeffery Cosiol*
Thomas S. Ertsgaard*
Daniel P. Giorgis*
Stephen T. Karg*
J.D. Ljungquist*
Jerald P. Martocci*

Mark A. Railsback*
David Robin*
Daniel A. Traill*
Grant N. Wichenko*
David J. Branson

**Denotes members of voting status when the document was approved for publication*

ASHRAE STANDARDS COMMITTEE 2002-2003

Thomas E. Watson, *Chair*
Van D. Baxter, *Vice-Chair*
Charles G. Arnold
Dean S. Borges
Paul W. Cabot
Charles W. Coward, Jr.
Brian P. Dougherty
Hakim Elmahdy
Arthur D. Hallstrom
Matt R. Hargan
Richard D. Hermans
Stephen D. Kennedy

David E. Knebel
Frederick H. Kohloss
William J. Landman
Merle F. McBride
Ross D. Montgomery
Cyrus H. Nasser
Davor Novosel
Dennis A. Stanke
Michael H. Tavares
Steven T. Taylor
David R. Tree
Terry E. Townsend, *CO*
Maureen Grasso, *ExO*

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 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 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.

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 is not part of the standard but is provided for information only.)

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 the continuous maintenance procedures contained in the *Manual for Processing ASHRAE Standards* and *PC Guidance* and of deliberations within Standing Standard Project Committee 135. The proposed changes are summarized below.

135b-1. Remove UTC timestamps from Trend Logs and guarantee Trend Log record ordering, p. 1.

In the following document, language to be added to existing clauses of ANSI/ASHRAE 135-2001 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.

135b-1. Remove UTC timestamps from Trend Logs and guarantee Trend Log record ordering.

Addendum 135b-1

[Change 12.23, p.227]

12.23 Trend Log Object Type

A Trend Log object monitors a property of a referenced object and, when predefined conditions are met, saves ("logs") the value of the property and a timestamp in an internal buffer for subsequent retrieval. The data may be logged periodically or upon a change of value. Errors that prevent the acquisition of the data, as well as changes in the status or operation of the logging process itself, are also recorded. Each timestamped buffer entry is called a trend log "record."

The referenced object may reside in the same device as the Trend Log object or in an external device. The referenced property's value may be recorded upon COV subscription or periodic poll. Where status flags are available (such as when the COVNotification or ReadPropertyMultiple services are used), they are also acquired and saved with the data.

Each Trend Log object maintains an internal, optionally fixed-size, buffer. This buffer fills or grows as log records are added. If the buffer becomes full, the least recent record is overwritten when a new record is added, or collection may be set to stop. Trend Log records are transferred as BACnetLogRecords using the ReadRange service. The buffer may be cleared by writing a zero to the Record_Count property. *Each record in the buffer has an implied SequenceNumber which is equal to the value the Total_Record_Count property has immediately after the record is added. If the Total_Record_Count is incremented past $2^{32}-1$, then it shall reset to 1.*

Several datatypes are defined for storage in the log records. The ability to store ANY datatypes is optional. Data stored in the log buffer may be optionally restricted in size to 32 bits, as in the case of bit strings, to facilitate implementation in devices with strict storage requirements.

Logging may be enabled and disabled through the Log_Enable property and at dates and times specified by the Start_Time and Stop_Time properties. Trend Log enabling and disabling is recorded in the log buffer.

Event reporting (notification) may be provided to facilitate automatic fetching of log records by processes on other devices such as file servers. Support is provided for algorithmic reporting; optionally, intrinsic reporting may be provided.

In intrinsic reporting, when the number of records specified by the Notification_Threshold property have been collected since the previous notification (or startup), a new notification is sent to all subscribed devices. BUFFER_READY algorithmic reporting is described in Clause 13.3.7.

~~In response to a notification, subscribers fetch the new records with a ReadRange service request using the Current_Notify_Time of the preceding notification as the 'Beginning Date' parameter and the Current_Notify_Time of the current notification as the 'Ending Date' parameter. In response to a notification, subscribers may fetch all of the new records. If a subscriber needs to fetch all of the new records, it should use the 'By Sequence Number' form of the ReadRange service request.~~

A missed notification may be detected by a subscriber if the ~~Current_Notify_Time~~ *Current_Notify_Record* it received in its previous notification is ~~older than~~ *different than* the ~~Previous_Notify_Time~~ *Previous_Notify_Record* parameter of the current notification. If the ReadRange-ACK response to the ReadRange request issued under these conditions has its 'firstitem' flag set to TRUE, Trend Log records have probably been missed by this subscriber.

The acquisition of log records by remote devices has no effect upon the state of the Trend Log object itself. This allows completely independent, but properly sequential, access to its log records by all remote devices. Any remote device can independently update its records at any time.

[Change **Table 12-27**, p.228]

Table 12-27. Properties of the Trend Log Object Type

Property Identifier	Property Datatype	Conformance Code
...
Records_Since_Notification	Unsigned32	O ³
Previous_Notify_Time	BACnetDateTime	Θ ³
Current_Notify_Time-Last_Notify_Record	BACnetDateTime-Unsigned32	O ³
Event_State	BACnetEventState	R
...

¹ These properties are required to be present if the monitored property is a BACnet property.

² If present, these properties are required to be writable.

³ These properties are required to be present if the object supports intrinsic reporting.

[Change **12.23.14**, p.230]

12.23.14 Log_Buffer

This property is a list of up to Buffer_Size timestamped records of datatype BACnetLogRecord, each of which conveys a recorded data value, an error related to data-collection, or status changes in the Trend Log object. Each record has data fields as follows:

Timestamp The *local* date and ~~time~~^{time, in UTC}, *time* when the record was collected.

...

Also associated with each record is an implied record number, the value of which is equal to Total_Record_Count at the point where the record has been added into the Log Buffer and Total_Record_Count has been adjusted accordingly. All clients must be able to correctly handle the case where the Trend Log is reset such that its Total_Record_Count is returned to zero and also the case where Total_Record_Count has wrapped back to 1.

The buffer is not network accessible except through the use of the ReadRange service with implicit reference to the Trend Log.

[Delete clause **12.23.19** (p. 231) and renumber clauses **12.23.20** through **12.23.27** to **12.23.19** through **12.23.26**]

[Change **12.23.19** (new number), p.231]

12.23.19 ~~Current_Notify_Time-Last_Notify_Record~~

This property, of type ~~BACnetDateTime-Unsigned32~~, represents the ~~timestamp~~^{SequenceNumber} associated with the most recently collected record whose collection triggered a notification. If no notification has occurred since logging occurred ~~the value of this property shall contain all wildcard values.~~ *be zero.* This property is required if intrinsic reporting is supported by this object.

[Change Table 13-3, p.238]

Table 13-3. Standard Object Property Values Returned in Notifications

Object	Event Type	Notification Parameters	Referenced Object's Properties
...			
Trend Log	BUFFER_READY	Buffer_Device Buffer_Object Buffer_Property Previous_Notification Current_Notification	Object_Identifier Object_Identifier BACnetDeviceObjectPropertyReference ¹ Previous_Notify_Time Last_Notify_Record Current_Notify_Time Total_Record_Count
...			

¹ This parameter conveys a reference to the Log_Buffer property of the Trend Log object.

[Change Table 13-4, p.238]

Table 13-4. Notification Parameters for Standard Event Types

Event Type	Notification Parameters	Description
...		
BUFFER_READY	Buffer_Device Buffer_Object Buffer_Property Previous_Notification Current_Notification	Object_Identifier of the device containing the buffer. Object_Identifier of the object containing the buffer Reference to the buffer property. Current_Notify_Time Current_Notification parameter of the previous notification sent or 0 if no previous notification has been sent. Current_Notify_Time of the current notification sent. Sequence Number of the record that triggered this notification.
...		

[Change Table 15-13, p.295]

Table 15-13. Structure of ReadRange Service Primitives

Parameter Name	Req	Ind	Rsp	Cnf
Argument	M	M(=)		
Object Identifier	M	M(=)		
Property Identifier	M	M(=)		
Property Array Index	C	C(=)		
Range	U	U(=)		
Result(+)			S	S(=)
Object Identifier			M	M(=)
Property Identifier			M	M(=)
Property Array Index			C	C(=)
Result Flags			M	M(=)
Item Count			M	M(=)
Item Data			M	M(=)
First Sequence Number			C	C(=)
Result(-)			S	S(=)
Error Type			M	M(=)

[Change Table 15-14, p.296]

Table 15-14. Structure of the 'Range' Parameter

Parameter Name	Req	Ind	Datatype
By Position	S	S(=)	
Reference Index	M	M(=)	Unsigned
Count	M	M(=)	INTEGER
By Time	S	S(=)	
Reference Time	M	M(=)	BACnetDateTime
Count	M	M(=)	INTEGER
Time Range	S	S(=)	
Beginning Time	M	M(=)	BACnetDateTime
Ending Time	M	M(=)	BACnetDateTime
By Sequence Number	S	S(=)	
Reference Sequence Number	M	M(=)	Unsigned32
Count	M	M(=)	INTEGER
By Time	S	S(=)	
Reference Time	M	M(=)	BACnetDateTime
Count	M	M(=)	INTEGER

[Change 15.8.1.1.4.2, p.296]

15.8.1.1.4.2 ~~By Time~~ By Sequence Number

The 'By Time' parameter shall indicate that the particular items to be read are referenced by timestamp. The 'By Sequence Number' parameter shall indicate that the particular items to be read are referenced by a sequence number and that the response shall include the sequence number of the first returned item. This differs semantically from the 'By Position' parameter choice. The Reference Number provided in the 'By Position' choice references an item by its position in the list. In contrast, the Reference Number provided in the 'By Sequence Number' choice references an item by its sequence number, which it is given when the item is added to the list. Not all lists implement the concept of a sequence number. An example of a list that does implement the concept of a sequence number is the Log_Buffer property of the Trend Log object.

[Change 15.8.1.1.4.2.1, p.296]

15.8.1.1.4.2.1 ~~Reference Time~~ Reference Sequence Number

The first (if 'Count' is positive) or last (if 'Count' is negative) item to be read shall be the first item with a timestamp newer than the time specified by the 'Reference Time' parameter. The 'Reference Sequence Number' parameter specifies the sequence number of the first (if 'Count' is positive) or last (if 'Count' is negative) item to be read.

[Change 15.8.1.1.4.2.2, p.296]

15.8.1.1.4.2.2 Count

The absolute value of the 'Count' parameter specifies the number of records to be read. If 'Count' is positive, the record specified by '~~Reference Time~~' 'Reference Sequence Number' shall be the first and oldest record read and returned; if 'Count' is negative, the record specified by '~~Reference Index~~' 'Reference Sequence Number' shall be the last and newest record. record read and returned. 'Count' may shall not be zero.

[Change 15.8.1.1.4.3, p.296]

15.8.1.1.4.3 Time Range By Time

The 'Time Range' 'By Time' parameter shall indicate that the particular ~~items to be read have timestamps within a specified range.~~ *item to be read is referenced by timestamp and that the Sequence Number of the item shall be returned in the response. This form of the service is expected to be used when searching lists that are loosely indexed by time.*

[Change 15.8.1.1.4.3.1, p.296]

15.8.1.1.4.3.1 Beginning Time Reference Time

~~The first item to be read shall be the first item with a timestamp newer than the time specified by the 'Beginning Time' parameter.~~ *If 'Count' is positive, the first record to be read shall be the first record with a timestamp newer than the time specified by the 'Reference Time' parameter. If 'Count' is negative, the last record to be read shall be the newest record with a timestamp older than the time specified by the 'Reference Time' parameter.*

[Change 15.8.1.1.4.3.2, p.296]

15.8.1.1.4.3.2 Ending Time Count

~~The last item to be read shall be the last item with a timestamp older than or equal to the time specified by the 'Ending Time' parameter.~~ *The absolute value of the 'Count' parameter specifies the number of records to be read. If 'Count' is positive, the first record with a timestamp newer than the time specified by 'Reference Time' shall be the first and oldest record read and returned; if 'Count' is negative, the newest record with a timestamp older than the time specified by 'Reference Time' shall be the last and newest record. 'Count' shall not be zero.*

[Add new clause 15.8.1.2.7, p.297]

15.8.1.2.7 First Sequence Number

This parameter, of type Unsigned32, specifies the sequence number of the first item returned. This parameter is only included if the 'Range' parameter of the request was of the type 'By Sequence Number' or 'By Time' and 'Item Count' is greater than 0.

[Change 15.8.2, p.297]

15.8.2 Service Procedure

The responding BACnet-user shall first verify the validity of the 'Object Identifier', 'Property Identifier' and 'Property Array Index' parameters and return a 'Result(-)' response with the appropriate error class and code if the object or property is unknown, if the referenced data is not a list or array, or if it is currently inaccessible for another reason.

If the 'Range' parameter is not present, then the responding BACnet-user shall read and attempt to return all of the available items in the list or array.

If the 'Range' parameter is present and specifies the 'By Position' parameters, then the responding BACnet-user shall read and attempt to return all of the items specified. The items specified include the item at the index specified by 'Reference Index' plus up to 'Count'-1 items following if 'Count' is positive, or up to -1-'Count' items preceding if 'Count' is negative. Array index 0 shall not be returned by this service; lists shall begin with index 1.

~~If the 'Range' parameter is present and specifies the 'By Time' parameters, then the responding BACnet-user shall read and attempt to return all of the items specified. If 'By Time' parameters are specified and the property values are not timestamped an error shall be returned. The items specified include the first item with a timestamp newer than 'Reference Time' plus up to 'Count' -1 items following if 'Count' is positive, or up to -1-'Count' items preceding if 'Count' is negative. Array index 0 shall not be returned by this service; lists shall begin with index 1.~~

If the 'Range' parameter is present and specifies the 'Time Range' parameters, then the responding BACnet user shall read and attempt to return all of the items specified. If 'Time Range' parameters are specified and the property values are not timestamped an error shall be returned. The items specified include all items with a timestamp newer than 'Beginning Time' and less than or equal to 'Ending Time'. If any field of the 'Beginning Time' parameter contains a wildcard value, then all times shall be considered newer than 'Beginning Time'. If any fields of the 'Ending Time' parameter contains a wildcard value then all times shall be considered older than 'Ending Time'.

If the 'Range' parameter is present and specifies the 'By Time' parameter, then the responding BACnet-user shall read and attempt to return all of the items specified. If 'By Time' parameters are specified and the property values are not timestamped, an error shall be returned. The items specified include the first item with a timestamp newer than 'Reference Time' plus up to 'Count'-1 items following if 'Count' is positive, or up to -1-'Count' items preceding if 'Count' is negative. Array index 0 shall not be returned by this service; lists shall begin with index 1. The sequence number of the first item returned shall be included in the response.

If the 'Range' parameter is present and specifies the 'By Sequence Number' parameters, then the responding BACnet-user shall read and attempt to return all of the items specified. The items specified are all items with a sequence number in the range 'Reference Sequence Number' to 'Reference Sequence Number' plus 'Count'-1 if 'Count' is positive, or in the range 'Reference Sequence Number' plus 'Count'+1 to 'Reference Sequence' if 'Count' is negative. Array index 0 shall not be returned by this service; lists shall begin with index 1.

To avoid missing items when using chained time-based reads, ~~items with a timestamp equal to the current time of the device shall not be included in the response. Items with the same timestamp shall be returned atomically.~~ *the first item in the desired set should be found using the 'By Time' form of the 'Range' parameter. Subsequent requests to retrieve the remaining items in the desired set should use the 'By Sequence Number' form of the 'Range' parameter. The reason for this is that lists that include a timestamp but are ordered by time of arrival may have entries with out-of-order timestamps due to negative time changes in the local device's clock.* If items are read that match the request parameters but cannot be returned in the response, the 'Result Flags' parameter shall contain the MOREITEMS flag set to TRUE; otherwise it shall be FALSE. Remaining items may be obtained with subsequent requests specifying appropriately chosen parameters.

The returned response shall convey the number of items read and returned using the 'Item Count' parameter. The actual items shall be returned in the 'Item Data' parameter. If the returned response includes the first positional index and a 'By Position' request had been made, ~~or the oldest timestamped item and a 'By Time' or 'Time Range' request had been made,~~ *or the oldest sequence number and a 'By Sequence Number' or 'By Time' request had been made,* then the 'Result Flags' parameter shall contain the FIRSTITEM flag set to TRUE; otherwise it shall be FALSE.

If the returned response includes the last positional index and a 'By Position' request had been made, ~~or the newest timestamped item and a 'By Time' or 'Time Range' request had been made,~~ *or the newest sequence number and a 'By Sequence Number' or 'By Time' request had been made,* then the 'Result Flags' shall contain the LASTITEM flag set to TRUE; otherwise it shall be FALSE.

If there are no items in the list that match the 'Range' parameter criteria, then a Result(+) shall be returned with an 'Item Count' of 0 and no 'First Sequence Number' parameter.

[Change ReadRange-Request production, 21, p.378]

```

ReadRange-Request ::= SEQUENCE {
    objectIdentifier      [0] BACnetObjectIdentifier,
    propertyIdentifier     [1] BACnetPropertyIdentifier,
    propertyArrayIndex    [2] Unsigned OPTIONAL    -- used only with array datatype
    range                 CHOICE {
        byPosition        [3] SEQUENCE {
            referenceIndex Unsigned,
            count          INTEGER
        },

```

```

byTime [4] SEQUENCE {
    referenceTime BACnetDateTime,
    count INTEGER
},
-- context tag 4 is deprecated
timeRange [5] SEQUENCE {
    beginningTime BACnetDateTime,
    endingTime BACnetDateTime
},
-- context tag 5 is deprecated
bySequenceNumber [6] SEQUENCE {
    referenceIndex Unsigned,
    count INTEGER
},
byTime [7] SEQUENCE {
    referenceTime BACnetDateTime,
    count INTEGER
}
}

```

[Change ReadRange-ACK production, 21, p.378]

```

ReadRange-ACK ::= SEQUENCE {
    objectIdentifier [0] BACnetObjectIdentifier,
    propertyIdentifier [1] BACnetPropertyIdentifier,
    propertyArrayIndex [2] Unsigned OPTIONAL, -- used only with array datatype
    resultFlags [3] BACnetResultFlags,
    itemCount [4] Unsigned,
    itemData [5] SEQUENCE OF ABSTRACT-SYNTAX.&TYPE,
    firstSequenceNumber [6] Unsigned32 OPTIONAL -- used only if 'Item Count' > 0 and the request was either of type
    -- 'By Sequence Number' or 'By Time'
}

```

[Change BACnetEventType production, 21, p.394]

```

BACnetEventType ::= ENUMERATED {
    change-of-bitstring (0),
    change-of-state (1),
    change-of-value (2),
    command-failure (3),
    floating-limit (4),
    out-of-range (5),
    -- complex-event-type (6), -- see comment below
    buffer ready (7),
    -- context tag 7 is deprecated
    change-of-life-safety (8),
    -- enumeration 9 is used in a proposed addition documented in Addendum a to ANSI/ASHRAE 135-2001(135a-2)
    buffer-ready (10),
    ...
}

```

-- 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. It is expected that these enumerated values will correspond to the use of the
-- complex-event-type CHOICE [6] of the BACnetNotificationParameters production.
-- The last enumeration used in this version is ~~8~~ 10.

[Change BACnetEventParameter production, 21, p.393]

BACnetEventParameter ::= CHOICE {

```
...
buffer ready [7] SEQUENCE {
notification threshold [0] Unsigned,
previous notification count [1] Unsigned32
}
-- context tag 7 is deprecated
change-of-life-safety [8] SEQUENCE {
    time-delay [0] Unsigned,
    list-of-life-safety-alarm-values [1] SEQUENCE OF BACnetLifeSafetyState,
    list-of-alarm-values [2] SEQUENCE OF BACnetLifeSafetyState,
    mode-property-reference [3] BACnetDeviceObjectPropertyReference
}
-- context tag 9 is used in a proposed addition documented in Addendum a to ANSI/ASHRAE 135-2001(135a-2)
buffer-ready [10] SEQUENCE {
    notification-threshold [0] Unsigned,
    previous-notification-count [1] Unsigned32
}
}
```

-- CHOICE [6] has been intentionally omitted. It parallels the complex-event-type CHOICE [6] of the
-- BACnetNotificationParameters production which was introduced to allow the addition of proprietary event
-- algorithms whose event parameters are not necessarily network-visible.

[Change BACnetNotificationParameters production, 21, p.397]

BACnetNotificationParameters ::= CHOICE {

```
...
complex-event-type [6] SEQUENCE OF BACnetPropertyValue,
buffer ready [7] SEQUENCE {
buffer device [0] BACnetObjectIdentifier,
buffer object [1] BACnetObjectIdentifier,
previous notification [2] BACnetDateTime,
current notification [3] BACnetDateTime
};
-- context tag 7 is deprecated
change-of-life-safety [8] SEQUENCE {
    new-state [0] BACnetLifeSafetyState,
    new-mode [1] BACnetLifeSafetyMode,
    status-flags [2] BACnetStatusFlags,
    operation-expected [3] BACnetLifeSafetyOperation
};
-- context tag 9 is used in a proposed addition documented in Addendum a to ANSI/ASHRAE 135-2001(135a-2)
buffer-ready [10] SEQUENCE {
    buffer-property [0] BACnetDeviceObjectPropertyReference,
    previous-notification [1] Unsigned32,
    current-notification [2] Unsigned32
}
}
```

[Change BACnetPropertyIdentifier enumeration, 21, p.400 (note: this production was also extended in 135a-2)]

```
BACnetPropertyIdentifier ::= ENUMERATED {  
  ...  
  current_notify_time (129),  
  ...  
  last_notify_record (173),  
  ...  
  previous_notify_time (138),  
  ...  
  see_current_notify_time_unused (129),  
  ...  
  see_previous_notify_time_unused (138),  
  ...  
  see_last_notify_record (173),  
}
```

-- The special property identifiers all, optional, and required are reserved for use in the ReadPropertyConditional and ReadPropertyMultiple services or services not defined in this standard.
-- 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 468. 173.

[Change Annex C, p.438]

```
TREND-LOG ::= SEQUENCE {  
  object-identifier [75] BACnetObjectIdentifier,  
  ...  
  records-since-notification [140] Unsigned32 OPTIONAL,  
  previous_notify_time [138] BACnetDateTime OPTIONAL,  
  current_notify_time [129] BACnetDateTime OPTIONAL,  
  last_notify_record [173] Unsigned32 OPTIONAL,  
  event-state [36] BACnetEventState,  
  ...  
  profile-name [168] CharacterString OPTIONAL  
}
```

[Change Annex D.23, p.456]

D.23 Example of a Trend Log Object

The following is an example of a Trend Log object that periodically logs data from an object in a remote device and which performs buffer-ready notification via intrinsic reporting.

```
Property: Object_Identifier = (Trend Log, Instance 1)  
Property: Object_Name = "Room 3Log"  
Property: Object_Type = TREND_LOG  
Property: Description = "Room 3 Temperature"  
Property: Log_Enable = TRUE  
Property: Log_DeviceObjectProperty = ((Device, Instance 100), Analog Input, Instance 3, Present_Value)  
Property: Log_Interval = 6,000  
Property: Stop_When_Full = FALSE  
Property: Buffer_Size = 250  
Property: Log_Buffer = (((23-MAR-1998,12:32:33.0), 72.0,(FALSE,FALSE,FALSE,FALSE)),  
  ((23-MAR-1998,12:34:32.0),72.1,(FALSE,FALSE,FALSE,FALSE)),...)  
Property: Record_Count = 250
```

Property: Total_Record_Count = 131040
 Property: Notification_Threshold = 83
 Property: Records_Since_Notification = 30
~~Property: Previous_Notify_Time = (23-MAR-1998, 17:06:32.0)~~
~~Property: Current_Notify_Time = (23-MAR-1998, 19:52:34.0)~~
 Property: Last_Notify_Record = 131010
 Property: Event_State = NORMAL
 Property: Notification_Class = 1
 ...

[Add a new entry to **History of Revisions**, p.557]

(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 (Informative)

<i>Protocol</i>		<i>Summary of Changes to the Standard</i>
<i>Version</i>	<i>Revision</i>	
...
1	3	<p>Addendum b to ANSI/ASHRAE 135-2001 Approved by the ASHRAE Standards Committee January 25, 2003; and by the ASHRAE Board of Directors January 30, 2003.</p> <p>1. Remove UTC timestamps from Trend Logs and guarantee Trend Log record ordering.</p>

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.