ADDENDA

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

BACnet™ Data Communication Protocol for Building Automation and Control Networks

Approved by ASHRAE on February 29, 2016, and by the American National Standards Institute on March 1, 2016.

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.

© 2016 ASHRAE ISSN 1041-2336

Approved American National Standard
ANSI
www.ansi.org

ASHRAE Standing Standard Project Committee 135
Cognizant TC: 1.4, Control Theory and Application
SPLS Liaison: Mark P. Modera

Carl Neilson*, *Chair*
Bernhard Isler, *Vice-Chair*
Michael Osborne*, *Secretary*
Coleman L. Brumley, Jr.*
Clifford H. Copass*
Stuart G. Donaldson*
Michael P. Graham*
David G. Holmberg*
Daniel Kollodge*
Jake Kopocis*
Thomas Kurowski*
H. Michael Newman*
Duffy O'Craven*
Gregory M. Spiro*
Grant N. Wichenko*

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

ASHRAE STANDARDS COMMITTEE 2015–2016

Douglass T. Reindl, *Chair*
Rita M. Harrold, *Vice-Chair*
James D. Aswegan
Niels Bidstrup
Donald M. Brundage
John A. Clark
Waller S. Clements
John F. Dunlap
James W. Earley, Jr.
Keith I. Emerson
Steven J. Emmerich
Julie M. Ferguson
Walter T. Grondzik
Roger L. Hedrick
Srinivas Katipamula
Rick A. Larson
Lawrence C. Markel
Arsen K. Melikov
Mark P. Modera
Cyrus H. Nasseri
Heather L. Platt
David Robin
Peter Simmonds
Dennis A. Stanke
Wayne H. Stoppelmoor, Jr.
Jack H. Zarour
Julia A. Keen, *BOD ExO*
James K. Vallort, *CO*

Stephanie C. Reiniche, *Senior Manager of Standards*

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.

ASHRAE is a registered trademark of the American Society of Heating, Refrigerating and Air-Conditioning Engineers, Inc.
ANSI is a registered trademark of the American National Standards Institute.

© ASHRAE (www.ashrae.org). For personal use only. Additional reproduction, distribution,
or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

[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.

135-2012*aq*-1 Add Elevator Object Types, p. 2
135-2012*aq*-2 Add COV Property Multiple Services, p. 30
135-2012*aq*-3 Add a New Fault Algorithm FAULT_LISTED, p. 49

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.

© ASHRAE (www.ashrae.org). For personal use only. Additional reproduction, distribution,
or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

135-2012*aq*-1 Add Elevator Object Types

Rationale

"Elevators," defined as lifts (vertical or near-vertical transport) and escalators/passenger conveyors (horizontal or near-horizontal transport), are standard provisions for many buildings, including all high rise buildings.

There is need for standard objects to represent the status of lifts and escalators, and standardized services to convey this status, so that standardized remote condition-based monitoring and maintenance becomes possible using BACnet.

Some systems utilizing the objects and services presented herein will be quite large, connected by IP networks. For this reason, data is not considered to be conveyed in a timely manner, yet a central monitoring system needs to be able to know which of the items of data it has is the latest. This led to the concept here of data timestamped at the point of origin.

Also, because of the potential for large numbers of COV (change-of-value) subscriptions, and run-time changes in those subscriptions, the ability to subscribe or unsubscribe in a single request for COV notifications on a number of properties of a number of objects is provided, as well as to convey multiple COVs in a single notification. These new services are constructed very closely along the lines of the existing COV Property services.

Certain elevator industry concepts and practices have also been codified in these objects, such as the Energy_Meter properties wrapping at a certain point to an unspecified value which may differ from manufacturer to manufacturer. As an alternative to the intrinsic energy consumption indication, other objects may be referenced that represent energy consumption respectively.

Some elements of the elevator monitoring requirements, in particular representing Buildings, Rooms, and structural hierarchies, were generalized with an eye towards future extension for other building automation applications.

This addendum was developed by Bill Swan in cooperation with Prof. Dr. Albert So and Wendy Fan of the City University of Hong Kong, Department of Building & Construction.

[Add to **Clause 3.2**, in alphabetical position, p.6]

universal floor number: the number of a floor, beginning with 1 for the absolute lowest floor of the building and incrementing by one for each subsequent floor up.

© ASHRAE (www.ashrae.org). For personal use only. Additional reproduction, distribution,
or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

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

...

Several object types defined in this clause have a property called "Reliability" that indicates the existence of fault conditions for the object. Reliability-evaluation is the process of determining the value of this property. The first stage of reliability-evaluation is internal to the object and is completely defined by the device's vendor. The second stage, which is only found in certain object types, is the application of a fault algorithm. See Clause 13.4 for fault algorithm definitions and see the object type definitions to determine the fault algorithm supported by any particular object type. The different values that the Reliability property can take on are described below. Note that not all values are applicable to all object types.

...	...
TRIPPED	The end device, such as an actuator, is not responding to commands, prevented by a tripped condition or by being mechanically held open.
FAULTS_LISTED	*At least one fault indication is present in a BACnetLIST property. For details of the respective FAULT_LISTED fault algorithm see Clause 13.4.X.*

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

12.X Elevator Group Object Type

The Elevator Group object type defines a standardized object whose properties represent the externally visible characteristics of a group of lifts or escalators (a group being defined as those lifts or escalators controlled by a single supervisory controller).

The following figure illustrates an example of a structure of objects that represent lifts and escalators of a building.

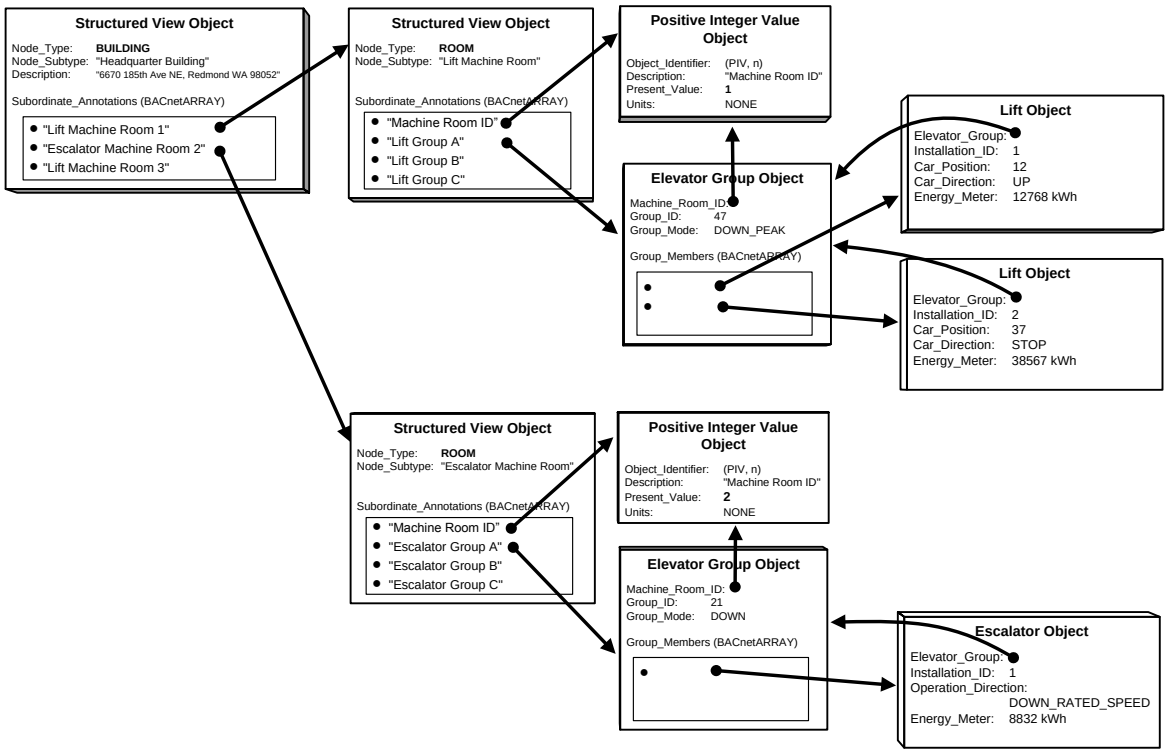

Figure 12-X1. Elevator Object Structure Example

The Elevator Group object type and its properties are summarized in Table 12-X and described in detail in this subclause.

© ASHRAE (www.ashrae.org). For personal use only. Additional reproduction, distribution,
or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

Table 12-X. Properties of the Elevator Group Object Type

Property Identifier	Property Datatype	Conformance Code
Object_Identifier	BACnetObjectIdentifier	R
Object_Name	CharacterString	R
Object_Type	BACnetObjectType	R
Description	CharacterString	O
Machine_Room_ID	BACnetObjectIdentifier	R
Group_ID	Unsigned8	R
Group_Members	BACnetARRAY[N] of BACnetObjectIdentifier	R
Group_Mode	BACnetLiftGroupMode	O[1]
Landing_Calls	BACnetLIST of BACnetLandingCallStatus	O[1]
Landing_Call_Control	BACnetLandingCallStatus	O[1]
Property_List	BACnetARRAY[N] of BACnetPropertyIdentifier	R
Profile_Name	CharacterString	O

[1] These properties shall be present only if this object represents a group of lifts.

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**.**

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 ELEVATOR_GROUP.

12.X.4 Description
This property, of type CharacterString, is a string of printable characters whose content is not restricted.

12.X.5 Machine_Room_ID
This property, of type BACnetObjectIdentifier, shall reference the Positive Integer Value Object whose Present_Value property contains the identification number for the machine room that contains the group of lifts or escalators represented by this object.

If there is no such identification number, this property shall contain an object instance number of 4194303.

12.X.6 Group_ID
This property, of type Unsigned8, shall represent the identification number for the group of lifts or escalators represented by this object. This identification number shall be unique for the groups in this machine room, but might not be otherwise unique in the building.

12.X.7 Group_Members
This property, of type BACnetARRAY of BACnetObjectIdentifier, references the member objects of the group. Each element shall contain the object identifier of a Lift or Escalator object representing lifts or escalators contained within the group represented by this object.

For inclusion of a multi-deck lift, represented by a chain of Lift objects, only one Lift object out of the chain is referenced. The selection of which Lift object is referenced is a local matter.

12.X.8 Group_Mode
This property, of type BACnetLiftGroupMode, shall convey the operating mode of the group of lifts. This is used to represent some special traffic modes of control of the supervisory controller of a group of lifts. Supervisory controllers are not required to support all modes. Under a special traffic mode, the car dispatching algorithm may be different.

UNKNOWN The current operating mode of the lift group is unknown.

© ASHRAE (www.ashrae.org). For personal use only. Additional reproduction, distribution,
or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

NORMAL	The lift group is in normal operating mode, and no special operating mode is in place.
DOWN_PEAK	Most passengers want to leave the building. This usually happens before lunch break or at the close of business.
TWO_WAY	Many passengers want to get to, or leave, a particular floor. This usually happens when there is a special function, ceremony, meeting or conference on a particular floor.
FOUR_WAY	Many passengers want to move between two particular floors. This can happen, for example, at a school when a group of students wants to travel from classroom to classroom on two different floors.
EMERGENCY_POWER	The whole lift group is operating under an emergency power supply. In such a condition, only limited services are provided and most lifts are in a homing mode. This situation can also occur during a fire alarm.
UP_PEAK	Most passengers gather at the main terminal, usually the ground floor, to get to different floors of the building. This situation usually happens in the morning right before office hours or right after lunch.

12.X.9 Landing_Calls
This property, of type BACnetLIST of BACnetLandingCallStatus, may be present if the Elevator Group object represents a group of lifts. Each element of this list shall represent a currently active call for the group of lifts.

12.X.10 Landing_Call_Control
This property, of type BACnetLandingCallStatus, may be present if the Elevator Group object represents a group of lifts. If it is present, it shall be writable. A write to this property is equivalent to a passenger pressing a call button at a landing, indicating either desired direction of travel or destination floor.

12.X.11 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.12 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.

© ASHRAE (www.ashrae.org). For personal use only. Additional reproduction, distribution,
or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

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

12.Y Lift Object Type

The lift object type defines a standardized object whose properties represent the externally visible characteristics of a lift. The object and its properties are summarized in Table 12-Y and described in detail in this subclause.

As there could be multiple car doors on a lift car, there could also be up to the same number of landing doors for the lift car at each floor. Normally, a landing door is driven by the car door as landing doors are not powered. So the status of a lift car door also reflects the status of the corresponding landing door at a particular floor. The following figure illustrates a lift car that has two doors. On the floor shown, there is no landing door for car door [1].

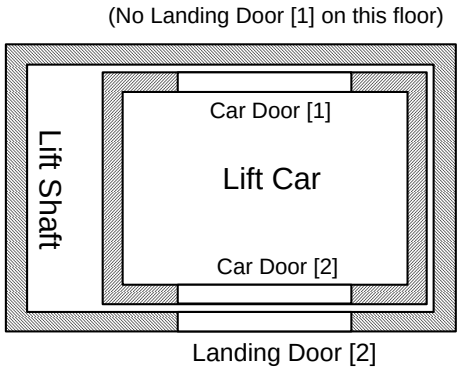

Figure 12-X2. Example Two Door Lift Car

Properties that are related to the doors of a car are of type BACnetARRAY. In each of these arrays, the element with the same array index is related to the same car door or corresponding landing door. The array properties Car_Door_Text, Assigned_Landing_Calls, Making_Car_Call, Registered_Car_Call, Car_Door_Status, Car_Door_Command, and Landing_Door_Status, if present, shall be of the same size. The number of array elements in these properties shall be equal to the number of car doors present in the lift car. The assignment of a car door to an array index is a local matter.

Lift objects may represent multi-deck lift cars. In this configuration, each deck is represented by its own Lift object. A Lift object representing a deck shall reference, in the properties Lower_Deck and Higher_Deck, the Lift objects representing its adjacent lower and higher decks, respectively. In the Lift object for the lowest deck, the Lower_Deck property may be absent. If the property is present, the object instance shall be 4194303. In the Lift object for the highest deck, the Higher_Deck property may be absent. If the property is present, the object instance shall be 4194303. Any synchronization or sharing of property values among all Lift objects of a multi-deck lift is a local matter. Write requests to properties that are synchronized or shared shall be equally accepted by any of the objects whose respective properties are synchronized or shared. The following figure illustrates an example of Lift objects for an elevator group of one double deck lift car and one triple deck lift car.

© ASHRAE (www.ashrae.org). For personal use only. Additional reproduction, distribution,
or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

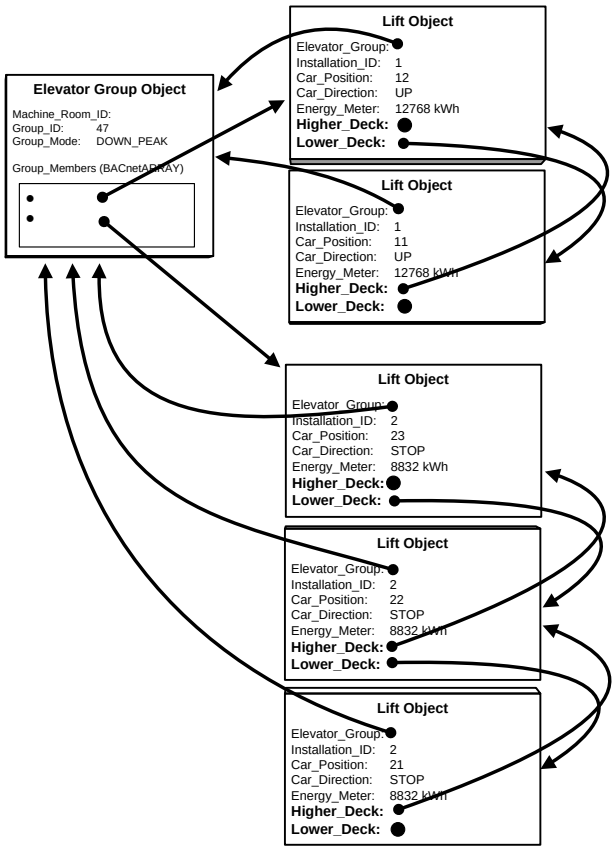

Figure 12-X3. Group of Multi-Deck Lift Cars Example

Lift objects that support intrinsic reporting shall apply the CHANGE_OF_STATE event algorithm on the Passenger_Alarm property. The pAlarmValues parameter shall contain the value TRUE.

For reliability-evaluation, the FAULT_LISTED fault algorithm can be applied.

© ASHRAE (www.ashrae.org). For personal use only. Additional reproduction, distribution,
or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

The Lift object type and its properties are summarized in Table 12-Y and described in detail in this subclause.

Table 12-Y. Properties of the Lift 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
Elevator_Group	BACnetObjectIdentifier	R
Group_ID	Unsigned8	R
Installation_ID	Unsigned8	R
Floor_Text	BACnetARRAY[N] of CharacterString	O
Car_Door_Text	BACnetARRAY[N] of CharacterString	O
Assigned_Landing_Calls	BACnetARRAY[N] of BACnetAssignedLandingCalls	O
Making_Car_Call	BACnetARRAY[N] of Unsigned8	O
Registered_Car_Call	BACnetARRAY[N] of BACnetLiftCarCallList	O
Car_Position	Unsigned8	R
Car_Moving_Direction	BACnetLiftCarDirection	R
Car_Assigned_Direction	BACnetLiftCarDirection	O
Car_Door_Status	BACnetARRAY[N] of BACnetDoorStatus	R
Car_Door_Command	BACnetARRAY[N] of BACnetLiftCarDoorCommand	O
Car_Door_Zone	BOOLEAN	O
Car_Mode	BACnetLiftCarMode	O
Car_Load	REAL	O
Car_Load_Units	BACnetEngineeringUnits	O[1]
Next_Stopping_Floor	Unsigned8	O
Passenger_Alarm	BOOLEAN	R
Time_Delay	Unsigned	O[2,3]
Time_Delay_Normal	Unsigned	O[3]
Energy_Meter	REAL	O
Energy_Meter_Ref	BACnetDeviceObjectReference	O
Reliability	BACnetReliability	O
Out_Of_Service	BOOLEAN	R
Car_Drive_Status	BACnetLiftCarDriveStatus	O
Fault_Signals	BACnetLIST of BACnetLiftFault	R
Landing_Door_Status	BACnetARRAY[N] of BACnetLandingDoorStatus	O
Higher_Deck	BACnetObjectIdentifier	O
Lower_Deck	BACnetObjectIdentifier	O
Event_Detection_Enable	BOOLEAN	O[2,3]
Notification_Class	Unsigned	O[2,3]
Event_Enable	BACnetEventTransitionBits	O[2,3]
Event_State	BACnetEventState	O[2,3]
Acked_Transitions	BACnetEventTransitionBits	O[2,3]
Notify_Type	BACnetNotifyType	O[2,3]
Event_Time_Stamps	BACnetARRAY[3] of BACnetTimeStamp	O[2,3]
Event_Message_Texts	BACnetARRAY[3] of CharacterString	O[3]
Event_Message_Texts_Config	BACnetARRAY[3] of CharacterString	O[3]
Event_Algorithm_Inhibit_Ref	BACnetObjectPropertyReference	O[3]
Event_Algorithm_Inhibit	BOOLEAN	O[3,4]
Reliability_Evaluation_Inhibit	BOOLEAN	O[5]
Property_List	BACnetARRAY[N] of BACnetPropertyIdentifier	R

© ASHRAE (www.ashrae.org). For personal use only. Additional reproduction, distribution,
or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

Profile_Name	CharacterString	O

[1] Car_Load_Units shall be present if, and only if, Car_Load is present
[2] These properties are required if the object supports intrinsic reporting.
[3] These properties shall be present only if the object supports intrinsic reporting.
[4] Event_Algorithm_Inhibit shall be present if Event_Algorithm_Inhibit_Ref is present.
[5] If this property is present, then the Reliability property shall be present.

12.Y.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**.**

12.Y.2 Object_Name
This property, of type CharacterString, shall represent a name for the object that is unique within the BACnet Device that maintains it. The minimum length of the string shall be one character. The set of characters used in the Object_Name shall be restricted to printable characters**.**

12.Y.3 Object_Type
This property, of type BACnetObjectType, indicates membership in a particular object type class. The value of this property shall be LIFT.

12.Y.4 Description
This property, of type CharacterString, is a string of printable characters whose content is not restricted.

12.Y.5 Status_Flags
This property, of type BACnetStatusFlags, represents four Boolean flags that indicate the general "health" of the lift. Three of the flags are associated with the values of other properties of this object. A more detailed status could be determined by reading the properties that are linked to these flags. The relationship between individual flags is not defined by the protocol. The four flags are

{IN_ALARM, FAULT, OVERRIDDEN, OUT_OF_SERVICE}

where:

IN_ALARM — Logical FALSE (0) if the Event_State property has a value of NORMAL, otherwise logical TRUE (1).

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

OVERRIDDEN — Logical TRUE (1) if the properties Making_Car_Call, Car_Door_Command, or Car_Mode have been overridden by some mechanism local to the lift this object represents. In this context, "overridden" is taken to mean that the lift operation is no longer tracking changes to these properties, and these properties are no longer indicating the respective values applied by the lift. Other properties indicating status of the lift are still tracking the current status of the lift. Otherwise logical FALSE (0).

OUT_OF_SERVICE — Logical TRUE (1) if the Out_Of_Service property has a value of TRUE, otherwise logical FALSE (0).

If the object supports event reporting, then this property shall be the pStatusFlags parameter for the object's event algorithm. See Clause 13.3 for event algorithm parameter descriptions.

12.Y.6 Elevator_Group
This property, of type BACnetObjectIdentifier, shall reference the Elevator Group object whose Group_Members property contains a reference to this Lift object.

If there is no such Elevator Group object, this property shall contain an object instance of 4194303.

12.Y.7 Group_ID
This property, of type Unsigned8, shall represent the identification number for the group of lifts that contains the lift represented by this object.

12.Y.8 Installation_ID

© ASHRAE (www.ashrae.org). For personal use only. Additional reproduction, distribution,
or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

This property, of type Unsigned8, shall represent the identification number for the lift represented by this object. This identification number shall be unique for the lift in this group, but might not be otherwise unique for other lifts in the machine room or the building.

12.Y.9 Floor_Text
This property, of type BACnetARRAY of CharacterString, represents the descriptions or names for the floors. The universal floor number serves as an index into this array. The size of this array shall match the highest universal floor number served by this lift.

12.Y.10 Car_Door_Text
This property, of type BACnetARRAY of CharacterString, represents the descriptions or names for the doors of the lift car. Each array element represents the description or name for the door of the car assigned to this array element.

12.Y.11Assigned_Landing_Calls
This property, of type BACnetARRAY of BACnetAssignedLandingCalls, shall represent the current landing calls and their direction for the lift represented by this object. Each array element represents the list of assigned landing calls for the door of the car assigned to this array element.

Each element in BACnetAssignedLandingCalls consists of the universal floor number and the direction, of type BACnetLiftCarDirection, which may be one of these values:

UP	The landing call is for upward travel.
DOWN	The landing call is for downward travel.
UP_AND_DOWN	The landing call is for both upward and downward travel having been initiated by two different passengers.

12.Y.12 Making_Car_Call
This property, of type BACnetARRAY of Unsigned8, indicates the last car calls written to this property. Writing to this property is equivalent to a passenger requesting that the car stop at the designated floor. Each array element represents the last car call written to this property for the door of the car assigned to this array element. If no car call has been written to an array element, the array element shall indicate a value of zero.

12.Y.13 Registered_Car_Call
This property, of type BACnetARRAY of BACnetLiftCarCallList, represents the lists of currently registered car calls (requests to stop at particular floors using a particular door) for this lift. Each array element represents the list of universal floor numbers for which calls are registered for the door of the car assigned to this array element.

12.Y.14 Car_Position
This property, of type Unsigned8, indicates the universal floor number of this lift's car position.

12.Y.15 Car_Moving_Direction
This property, of type BACnetLiftCarDirection, represents whether or not this lift's car is moving, and if so, in which direction. Car_Moving_Direction can take on one of these values:

UNKNOWN	The current moving direction of the lift is unknown.
STOPPED	The lift car is not moving.
UP	The lift car is moving upward.
DOWN	The lift car is moving downward.

12.Y.16 Car_Assigned_Direction
This property, of type BACnetLiftCarDirection, represents the direction the lift is assigned to move, based on current car calls. Car_Assigned_Direction can take on these values:

UNKNOWN	The direction assigned to the lift is unknown.
NONE	No direction is assigned to the lift car.
UP	The lift car is assigned to move upward.
DOWN	The lift car is assigned to move downward.

© ASHRAE (www.ashrae.org). For personal use only. Additional reproduction, distribution,
or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

UP_AND_DOWN	The lift car is assigned to either move upward or downward.
<Proprietary Enum Values>	A vendor may use other proprietary enumeration values to allow proprietary lift car direction assignments other than those defined by the standard. For proprietary extensions of this enumeration, see Clause 23.1 of this standard.

12.Y.17 Car_Door_Status
This property, of type BACnetARRAY of BACnetDoorStatus, indicates the status of the doors on the car. Each array element indicates the status of the car door assigned to this array element. Each array element can have one of these enumerated values:

UNKNOWN	The status of the car door is unknown.
CLOSING	The car door is closing.
CLOSED	The car door is fully closed but not yet locked.
OPENING	The car door is opening.
OPENED	The car door is fully opened.
SAFETY_LOCKED	The car door is fully closed and locked.
LIMITED_OPENED	The car door remains in a position between fully closed and fully opened.

12.Y.18 Car_Door_Command
This property, of type BACnetARRAY of BACnetLiftCarDoorCommand, indicates the last pending car door commands written to this property. Writing to this property is equivalent to a passenger requesting that the respective car door be opened or closed. Each array element represents the last pending car door command for the door of the car assigned to this array element.

Once the respective car door command is executed or no longer applicable, e.g., the car is now moving, the respective array element shall revert to NONE.

NONE	No car door command was written, or there is no pending or executing car door command.
OPEN	The car door was commanded to open, and execution is pending or in progress.
CLOSE	The car door was commanded to close, and execution is pending or in progress.

12.Y.19 Car_Door_Zone
This property, of type BOOLEAN, indicates whether (TRUE) or not (FALSE) the car is in the door zone, the region near the landing where the door is permitted to start opening.

12.Y.20 Car_Mode
This property, of type BACnetLiftCarMode, shall indicate the current operational mode of the car. Car_Mode can take on these values:

UNKNOWN	The current operational mode of the lift is unknown.
NORMAL	The lift is operating normally.
VIP	The lift is operating in "very important person" mode between a particular floor and the main terminal. In this mode, a particular lift is reserved for the use of the "VIPs."
HOMING	The lift is returning to the main terminal and is going to stay there and not provide any further service.
PARKING	The lift car is manually or automatically parked at a particular floor and will not provide any further service. This usually happens in a low traffic condition for the purpose of energy saving.
ATTENDANT_CONTROL	The lift is being manually controlled by an attendant in the car.
FIREFIGHTER_CONTROL	The lift is under a firefighter lift or firefighting lift mode of control. This usually happens during a fire alarm when firemen are on the site.
EMERGENCY_POWER	The lift is operating on emergency power. The lift may be limited in its operation, such as moving to a predefined floor only.
INSPECTION	The lift is under inspection. Control of the lift is performed from a control panel on the

© ASHRAE (www.ashrae.org). For personal use only. Additional reproduction, distribution,
or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

	car roof. The lift is not available for normal operation.
CABINET_RECALL	Control of the lift is performed from a control panel in the control cabinet. The lift is not available for normal operation.
EARTHQUAKE_OPERATION	The lift will stop operation at a predefined floor for earthquake evacuation.
FIRE_OPERATION	The lift is returning to the fire evacuation terminal and may stay there and may not provide any further service.
OUT_OF_SERVICE	The lift is not available for service.
OCCUPANT_EVACUATION	The lift is under an occupant evacuation mode of control. This usually happens during a fire alarm when the lift is used for evacuation of occupants.
<Proprietary Enum Values>	A vendor may use other proprietary enumeration values to allow proprietary car operation modes other than those defined by the standard. For proprietary extensions of this enumeration, see Clause 23.1 of this standard.

12.Y.21 Car_Load
This property, of type REAL, indicates the load in the car, both passengers and goods. The value of the Car_Load property shall be in units as indicated by the Car_Load_Units property..

12.Y.22 Car_Load_Units
This property, of type BACnetEngineeringUnits, indicates the measurement units of the Car_Load property. See the BACnetEngineeringUnits ASN.1 production in Clause 21 for a list of engineering units defined by this standard.

12.Y.23 Next_Stopping_Floor
This property, of type Unsigned8, indicates the universal floor number where the car will stop next when underway. If the car is not in motion, this property indicates the current universal floor number.

12.Y.24 Passenger_Alarm
This property, of type BOOLEAN, indicates whether (TRUE) or not (FALSE) the passenger alarm has been activated.

If the object supports event reporting, then this property shall be the pMonitoredValue parameter for the object's event algorithm. See Clause 13.3 for event algorithm parameter descriptions.

12.Y.25 Time_Delay
This property, of type Unsigned, is the pTimeDelay parameter for the object's event algorithm. See Clause 13.3 for event algorithm parameter descriptions.

12.Y.26 Time_Delay_Normal
This property, of type Unsigned, is the pTimeDelayNormal parameter for the object's event algorithm. See Clause 13.3 for event algorithm parameter descriptions.

12.Y.27 Energy_Meter
This property, of type REAL, indicates the accumulated energy consumption by the lift. The units shall be kilowatt-hours. When this value reaches 99999 kWh, it shall wrap to a value near zero; the particular value to which it wraps is a local matter.

If the Energy_Meter_Ref property is present and initialized (contains an instance other than 4194303), then the Energy_Meter property, if present, shall contain a value of 0.0.

12.Y.28 Energy_Meter_Ref
This property, of type BACnetDeviceObjectReference, references the object which indicates the accumulated energy consumption by the lift.

12.Y.29 Reliability
The Reliability property, of type BACnetReliability, provides an indication of whether the properties of this object or the operation of the lift represented by this object are "reliable" as far as the BACnet Device can determine and, if not, why.

If a fault algorithm is applied, then this property shall be the pCurrentReliability parameter for the object's fault algorithm. See Clause 13.4 for fault algorithm parameter descriptions.

12.Y.30 Out_Of_Service

© ASHRAE (www.ashrae.org). For personal use only. Additional reproduction, distribution,
or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

The Out_Of_Service property, of type BOOLEAN, is an indication whether (TRUE) or not (FALSE) the object is decoupled from the lift that this object represents. This means that the object does not track the status of the lift, and the object will not control the lift operation. The value of this property shall have no effect on the operation of the lift this object represents.

While this property has a value of TRUE, the status properties Assigned_Landing_Calls, Registered_Car_Call, Car_Position, Car_Moving_Direction, Car_Assigned_Direction, Car_Door_Status, Car_Door_Zone, Car_Load, Next_Stopping_Floor, Passenger_Alarm, Energy_Meter, Car_Drive_Status, Fault_Signals, and Landing_Door_Status shall not track the status of the lift. These properties shall be writable while Out_Of_Service is TRUE.

While this property has a value of TRUE, the properties Making_Car_Call, Car_Door_Command, and Car_Mode, shall not have any effect on the operation of the lift. In addition, these properties shall not track the respective values currently applied by the lift. These properties shall be writable while Out_Of_Service is TRUE.

While the Out_Of_Service property is TRUE, the properties listed in this clause normally indicating status or currently applied control values may be changed to any value as a means of simulating specific fixed conditions or for testing purposes. Object functions that depend on the state of these properties shall respond to changes made to these properties while Out_Of_Service is TRUE, as if those changes had occurred in the lift.

12.Y.31 Car_Drive_Status

This property, of type BACnetLiftCarDriveStatus, shall indicate the current status of the lift's motor drive system. Car_Drive_Status can take on these values:

UNKNOWN	The status of the lift's motor drive system is unknown.
STATIONARY	The motor is not moving.
BRAKING	The brake is operating.
ACCELERATE	The lift car is moving along an acceleration profile, up or down.
DECELERATE	The lift car is moving along a deceleration profile, up or down.
RATED_SPEED	The lift car is moving under rated speed, up or down.
SINGLE_FLOOR_JUMP	The lift car is going from one floor to the next consecutive floor, up or down.
TWO_FLOOR_JUMP	The lift car is moving two floors up or down.
THREE_FLOOR_JUMP	The lift car is moving three floors up or down.
MULTI_FLOOR_JUMP	The lift car is moving four or more floors up or down.
<Proprietary Enum Values>	A vendor may use other proprietary enumeration values to allow proprietary car drive status values other than those defined by the standard. For proprietary extensions of this enumeration, see Clause 23.1 of this standard.

12.Y.32 Fault_Signals

This property, of type BACnetLIST of BACnetLiftFault, represents a list of values that indicates fault conditions of the lift. The Fault_Signals property may be empty or contain any set of the following values, without duplicates:

CONTROLLER_FAULT	The fault is due to a malfunctioning controller.
DRIVE_AND_MOTOR_FAULT	The fault is related to the motor drive, either electrical or mechanical.
GOVERNOR_AND_SAFETY_GEAR_FAULT	The fault is related to the governor and safety gear system.
LIFT_SHAFT_DEVICE_FAULT	The fault is related to a device inside the lift shaft, such as the position detector, a limit switch, etc.
POWER_SUPPLY_FAULT	The fault is related to the supply of electrical power, not to the lift system itself.
SAFETY_INTERLOCK_FAULT	There is a fault with the chain of car and landing doors locks which must indicate a fully closed and locked condition before a lift car can move.
DOOR_CLOSING_FAULT	A door is failing to close.
DOOR_OPENING_FAULT	A door is failing to open.
CAR_STOPPED_OUTSIDE_LANDING_ZONE	The car stopped outside the landing zone while in normal operation.

© ASHRAE (www.ashrae.org). For personal use only. Additional reproduction, distribution,
or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

CALL_BUTTON_STUCK	Any car or landing call continues to remain registered after the car has stopped at the floor.
START_FAILURE	The lift failed to start moving.
CONTROLLER_SUPPLY_FAULT	The power supply for the lift controller is out of its specified range or failed.
SELF_TEST_FAILURE	Any self-test function failed.
RUNTIME_LIMIT_EXCEEDED	The lift did not reach the expected zone.
POSITION_LOST	The lift control has lost information on the position of the car.
DRIVE_TEMPERATURE_EXCEEDED	The temperature of the drive system of the lift exceeded its limits.
LOAD_MEASUREMENT_FAULT	The car load measurement system is in a fault condition.
<Proprietary Enum Values>	A vendor may use other proprietary enumeration values to allow proprietary lift fault signals other than those defined by the standard. For proprietary extensions of this enumeration, see Clause 23.1 of this standard.

The mechanism for determining the existence of a fault condition is a local matter.

This property is the value of the pMonitoredList parameter of the object's fault algorithm. See Clause 13.4 for fault algorithm parameter descriptions.

12.Y.33 Landing_Door_Status
This property, of type BACnetARRAY of BACnetLandingDoorStatus, represents the status of the landing doors on the floors served by this lift. Each element of this array represents the list of landing doors for the door of the car assigned to this array element.

A landing door status includes the universal floor number and the currently active door status for the landing door. The status values that each landing door status can take on are:

UNKNOWN	The landing door status is unknown.
NONE	There is no landing door for the respective car door.
CLOSING	The landing door is closing.
CLOSED	The landing door is fully closed but not locked.
OPENING	The landing door is opening.
OPENED	The landing door is fully opened.
SAFETY_LOCK	The landing door is fully closed and locked.
LIMITED_OPENED	The landing door remains in a state between fully closed and fully opened.

12.Y.34 Higher_Deck
This property, of type BACnetObjectIdentifier, references the Lift object that is representing the car deck above the car deck represented by this object.

If this property is present, and there is no higher deck, then the object instance shall be 4194303.

12.Y.35 Lower_Deck
This property, of type BACnetObjectIdentifier, references the Lift object that is representing the car deck below the car deck represented by this object.

If this property is present, and there is no lower deck, then the object instance shall be 4194303.

12.Y.36 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.

© ASHRAE (www.ashrae.org). For personal use only. Additional reproduction, distribution,
or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

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.Y.37 Notification_Class
This property, of type Unsigned, shall specify the instance of the Notification Class object to use for event-notification-distribution.

12.Y.38 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 (TRUE, TRUE, TRUE) at a minimum.

12.Y.39 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.Y.40 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.Y.41 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.Y.42 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). Timestamps of type Time or Date shall have X'FF' in each octet, and Sequence number timestamps shall have the value 0 if no event of that type has ever occurred for the object.

12.Y.43 Event_Message_Texts
This read-only property, of type BACnetARRAY[3] of CharacterString, shall convey the message text values of the last TO-OFFNORMAL, TO-FAULT, and TO-NORMAL events (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.Y.44 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.Y.45 Event_Algorithm_Inhibit_Ref
This property, of type BACnetObjectPropertyReference, indicates the property which controls the value of property Event_Algorithm_Inhibit. When this property is present and initialized (contains an instance other than 4194303), the referenced property shall be of type BACnetBinaryPV or BOOLEAN.

12.Y.46 Event_Algorithm_Inhibit
This property, of type BOOLEAN, indicates whether (TRUE) or not (FALSE) the event algorithm has been disabled for the object (see Clause 13.2.2.1). This property is a runtime override that allows temporary disabling of the event algorithm.

If the Event_Algorithm_Inhibit_Ref property is present and initialized (contains an instance other than 4194303), then the Event_Algorithm_Inhibit property shall be read-only and shall reflect the value of the property referenced by Event_Algorithm_Inhibit_Ref. A BACnetBinaryPV value of INACTIVE shall map to a value of FALSE and a value of ACTIVE shall map to a value of TRUE. If the referenced property does not exist, it shall be assumed to have a value of FALSE.

© ASHRAE (www.ashrae.org). For personal use only. Additional reproduction, distribution,
or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

If the Event_Algorithm_Inhibit_Ref property is absent or is uninitialized and Event_Detection_Enable is TRUE, then the Event_Algorithm_Inhibit property shall be writable.

12.Y.47 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.Y.48 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.Y.49 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.

© ASHRAE (www.ashrae.org). For personal use only. Additional reproduction, distribution,
or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

[Add new clause **12.Z**, p. 459]

12.Z Escalator Object Type

The Escalator object type defines a standardized object whose properties represent the externally visible characteristics of an escalator. The object and its properties are summarized in Table 12-Z and described in detail in this subclause.

Escalator objects that support intrinsic reporting shall apply the CHANGE_OF_STATE event algorithm on the Passenger_Alarm property. The pAlarmValues parameter shall contain the value TRUE.

For reliability-evaluation, the FAULT_LISTED fault algorithm can be applied.

The Escalator object type and its properties are summarized in Table 12-Z and described in detail in this subclause.

Table 12-Z. Properties of the Escalator 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
Elevator_Group	BACnetObjectIdentifier	R
Group_ID	Unsigned8	R
Installation_ID	Unsigned8	R
Power_Mode	BOOLEAN	O
Operation_Direction	BACnetEscalatorOperationDirection	R
Escalator_Mode	BACnetEscalatorMode	O
Energy_Meter	REAL	O
Energy_Meter_Ref	BACnetDeviceObjectReference	O
Reliability	BACnetReliability	O
Out_Of_Service	BOOLEAN	R
Fault_Signals	BACnetLIST of BACnetEscalatorFault	O
Passenger_Alarm	BOOLEAN	R
Time_Delay	Unsigned	O[1,2]
Time_Delay_Normal	Unsigned	O[2]
Event_Detection_Enable	BOOLEAN	O[1,2]
Notification_Class	Unsigned	O[1,2]
Event_Enable	BACnetEventTransitionBits	O[1,2]
Event_State	BACnetEventState	O[1,2]
Acked_Transitions	BACnetEventTransitionBits	O[1,2]
Notify_Type	BACnetNotifyType	O[1,2]
Event_Time_Stamps	BACnetARRAY[3] of BACnetTimeStamp	O[1,2]
Event_Message_Texts	BACnetARRAY[3] of CharacterString	O[2]
Event_Message_Texts_Config	BACnetARRAY[3] of CharacterString	O[2]
Event_Algorithm_Inhibit	BOOLEAN	O[2,3]
Event_Algorithm_Inhibit_Ref	BACnetObjectPropertyReference	O[2]
Reliability_Evaluation_Inhibit	BOOLEAN	O[4]
Property_List	BACnetARRAY[N] of BACnetPropertyIdentifier	R
Profile_Name	CharacterString	O

[1] These properties are required if the object supports intrinsic reporting.
[2] These properties shall be present only if the object supports intrinsic reporting.
[3] Event_Algorithm_Inhibit shall be present if Event_Algorithm_Inhibit_Ref is present.
[4] If this property is present, then the Reliability property shall be present.

12.Z.1 Object_Identifier

© ASHRAE (www.ashrae.org). For personal use only. Additional reproduction, distribution,
or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

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**.**

12.Z.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.Z.3 Object_Type
This property, of type BACnetObjectType, indicates membership in a particular object type class. The value of this property shall be ESCALATOR.

12.Z.4 Description
This property, of type CharacterString, is a string of printable characters whose content is not restricted.

12.Z.5 Status_Flags
This property, of type BACnetStatusFlags, represents four Boolean flags that indicate the general "health" of the escalator. Three of the flags are associated with the values of other properties of this object. A more detailed status could be determined by reading the properties that are linked to these flags. The relationship between individual flags is not defined by the protocol. The four flags are

{IN_ALARM, FAULT, OVERRIDDEN, OUT_OF_SERVICE}

where:

IN_ALARM — Logical FALSE (0) if the Event_State property has a value of NORMAL, otherwise logical TRUE (1).

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

OVERRIDDEN — Logical TRUE (1) if the property Escalator_Mode has been overridden by some mechanism local to the escalator this object represents. In this context, "overridden" is taken to mean that the escalator operation is no longer tracking changes to these properties, and these properties are no longer indicating the respective values applied by the escalator. Other properties indicating status of the escalator are still tracking the current status of the escalator. Otherwise logical FALSE (0).

OUT_OF_SERVICE — Logical TRUE (1) if the Out_Of_Service property has a value of TRUE, otherwise logical FALSE (0).

If the object supports event reporting, then this property shall be the pStatusFlags parameter for the object's event algorithm. See Clause 13.3 for event algorithm parameter descriptions.

12.Z.6 Elevator_Group
This property, of type BACnetObjectIdentifier, shall reference the Elevator Group object whose Group_Members property contains a reference to this Escalator object.

If there is no such Elevator Group object, this property shall contain an object instance of 4194303.

12.Z.7 Group_ID
This property, of type Unsigned8, shall represent the identification number for the group of escalators that contains the escalator represented by this object.

12.Z.8 Installation_ID
This property, of type Unsigned8, shall represent the identification number for the escalator represented by this object.

12.Z.9 Power_Mode
This property, of type BOOLEAN, indicates whether (TRUE) or not (FALSE) the escalator is powered (independent of whether it is moving).

12.Z.10 Operation_Direction
This property, of type BACnetEscalatorOperationDirection, represents the direction and speed in which this escalator is presently moving. Operation_Direction can take on these values:

© ASHRAE (www.ashrae.org). For personal use only. Additional reproduction, distribution,
or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

UNKNOWN	The current operation direction is unknown.
STOPPED	The escalator or slanted passenger conveyor is not moving.
UP_RATED_SPEED	The escalator or slanted passenger conveyor is moving upward at rated speed.
UP_REDUCED_SPEED	The escalator or slanted passenger conveyor is moving upward at a reduced speed. This is for energy conservation when there are no passengers.
DOWN_RATED_SPEED	The escalator or slanted passenger conveyor is moving downward at rated speed.
DOWN_REDUCED_SPEED	The escalator or slanted passenger conveyor is moving downward under reduced speed. Again, this is to save energy.
<Proprietary Enum Values>	A vendor may use other proprietary enumeration values to allow proprietary escalator operation direction values other than those defined by the standard. For proprietary extensions of this enumeration, see Clause 23.1 of this standard.

12.Z.11 Escalator_Mode
This property, of type BACnetEscalatorMode, shall indicate the current operational mode of the escalator. Escalator_Mode can take on these values:

UNKNOWN	The current operational mode of the escalator or slanted passenger conveyor is unknown.
STOP	The escalator or slanted passenger conveyor is not moving.
UP	The escalator or slanted passenger conveyor is moving upward.
DOWN	The escalator or slanted passenger conveyor is moving downward.
INSPECTION	The escalator or slanted passenger conveyor is under inspection. Control of the escalator is performed from a control panel on the escalator. The escalator is not available for normal operation.
OUT_OF_SERVICE	The escalator or slanted passenger conveyor is not available for service.
<Proprietary Enum Values>	A vendor may use other proprietary enumeration values to allow proprietary escalator or slanted passenger conveyor operation modes other than those defined by the standard. For proprietary extensions of this enumeration, see Clause 23.1 of this standard.

12.Z.12Energy_Meter
This property, of type REAL, indicates the accumulated energy consumption by the escalator. The units shall be kilowatt-hours. When this value reaches 99999 kWh, it shall wrap to a value near zero; the particular value to which it wraps is a local matter.

If the Energy_Meter_Ref property is present and initialized (contains an instance other than 4194303), then the Energy_Meter property, if present, shall have a value of 0.0.

12.Z.13 Energy_Meter_Ref
This property, of type BACnetDeviceObjectReference, references the object which indicates the accumulated energy consumption by the escalator.

12.Z.14 Reliability
The Reliability property, of type BACnetReliability, provides an indication of whether the properties of this object or the operation of the escalator represented by this object are "reliable" as far as the BACnet Device can determine and, if not, why.

If a fault algorithm is applied, then this property shall be the pCurrentReliability parameter for the object's fault algorithm. See Clause 13.4 for fault algorithm parameter descriptions.

12.Z.15 Out_Of_Service
The Out_Of_Service property, of type BOOLEAN, is an indication whether (TRUE) or not (FALSE) the object is decoupled from the escalator that this object represents. This means that the object does not track the status of the escalator, and the object will not control the escalator operation. The value of this property shall have no effect on the operation of the escalator this object represents.

© ASHRAE (www.ashrae.org). For personal use only. Additional reproduction, distribution,
or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

While this property has a value of TRUE, the status properties Power_Mode, Operation_Direction, Energy_Meter, Fault_Signals, and Passenger_Alarm shall not track the status of the escalator. These properties shall be writable while Out_Of_Service is TRUE.

While this property has a value of TRUE, the property Escalator_Mode shall not have any effect on the operation of the escalator. In addition, this property shall not track the respective value currently applied by the lift. The property Escalator_Mode shall be writable while Out_Of_Service is TRUE.

While the Out_Of_Service property is TRUE, the properties listed in this clause normally indicating status or currently applied control values may be changed to any value as a means of simulating specific fixed conditions or for testing purposes. Object functions that depend on the state of these properties shall respond to changes made to these properties while Out_Of_Service is TRUE, as if those changes had occurred in the escalator.

12.Z.16 Fault_Signals
This property, of type BACnetLIST of BACnetEscalatorFault, represents a list of values that indicates fault conditions of the escalator. The Fault_Signals property may be empty or contain any set of the following values, without duplicates.

CONTROLLER_FAULT	The fault is due to a malfunctioning controller.
DRIVE_AND_MOTOR_FAULT	The fault is related to the motor drive, either electrical or mechanical.
MECHANICAL_COMPONENT_FAULT	The fault is related to the failure of a mechanical component.
OVERSPEED_FAULT	The fault is due to overspeed operation, either up or down.
POWER_SUPPLY_FAULT	The fault is related to the electric power supply; one or more phases of the electrical power has failed.
SAFETY_DEVICE_FAULT	The fault is due to the triggering of any of the escalator's safety devices.
CONTROLLER_SUPPLY_FAULT	The power supply for the escalator controller is out its specified range or failed.
DRIVE_TEMPERATURE_EXCEEDED	The temperature of the drive system of the escalator exceeded its limits.
COMB_PLATE_FAULT	A comb plate safety switch is activated. This may indicate that debris is lodged between the comb and steps of the escalator.
<Proprietary Enum Values>	A vendor may use other proprietary enumeration values to allow proprietary escalator fault signals other than those defined by the standard. For proprietary extensions of this enumeration, see Clause 23.1 of this standard.

The mechanism for determining the existence of a fault condition is a local matter.

This property is the value of the pMonitoredList parameter of the object's fault algorithm. See Clause 13.4 for fault algorithm parameter descriptions.

12.Z.17 Passenger_Alarm
This property, of type BOOLEAN, indicates whether (TRUE) or not (FALSE) the passenger alarm has been activated, thus stopping the escalator, and the alarm has not yet been cleared by a maintenance technician.

If the object supports event reporting, then this property shall be the pMonitoredValue parameter for the object's event algorithm. See Clause 13.3 for event algorithm parameter descriptions.

12.Z.18 Time_Delay
This property, of type Unsigned, is the pTimeDelay parameter for the object's event algorithm. See Clause 13.3 for event algorithm parameter descriptions.

12.Z.19 Time_Delay_Normal
This property, of type Unsigned, is the pTimeDelayNormal parameter for the object's event algorithm. See Clause 13.3 for event algorithm parameter descriptions.

12.Z.20 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.

© ASHRAE (www.ashrae.org). For personal use only. Additional reproduction, distribution,
or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

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.Z.21 Notification_Class
This property, of type Unsigned, shall specify the instance of the Notification Class object to use for event-notification-distribution.

12.Z.22 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 (TRUE, TRUE, TRUE) at a minimum.

12.Z.23 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.Z.24 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.Z.25 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.Z.26 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). Timestamps of type Time or Date shall have X'FF' in each octet, and Sequence number timestamps shall have the value 0 if no event of that type has ever occurred for the object.

12.Z.27 Event_Message_Texts
This read-only property, of type BACnetARRAY[3] of CharacterString, shall convey the message text values of the last TO-OFFNORMAL, TO-FAULT, and TO-NORMAL events (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.Z.28 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.Z.29 Event_Algorithm_Inhibit_Ref
This property, of type BACnetObjectPropertyReference, indicates the property which controls the value of property Event_Algorithm_Inhibit. When this property is present and initialized (contains an instance other than 4194303), the referenced property shall be of type BACnetBinaryPV or BOOLEAN.

12.Z.30 Event_Algorithm_Inhibit
This property, of type BOOLEAN, indicates whether (TRUE) or not (FALSE) the event algorithm has been disabled for the object (see Clause 13.2.2.1). This property is a runtime override that allows temporary disabling of the event algorithm.

If the Event_Algorithm_Inhibit_Ref property is present and initialized (contains an instance other than 4194303), then the Event_Algorithm_Inhibit property shall be read-only and shall reflect the value of the property referenced by Event_Algorithm_Inhibit_Ref. A BACnetBinaryPV value of INACTIVE shall map to a value of FALSE and a value of ACTIVE shall map to a value of TRUE. If the referenced property does not exist, it shall be assumed to have a value of FALSE.

© ASHRAE (www.ashrae.org). For personal use only. Additional reproduction, distribution,
or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

If the Event_Algorithm_Inhibit_Ref property is absent or is uninitialized and Event_Detection_Enable is TRUE, then the Event_Algorithm_Inhibit property shall be writable.

12.Z.31 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.Z.32 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.Z.33 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.

© ASHRAE (www.ashrae.org). For personal use only. Additional reproduction, distribution,
or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

[Add new productions to **Clause 21**, pp. 661-713, distributed alphabetically in section "Base Types"]

BACnetAssignedLandingCalls ::= SEQUENCE {
    landing-calls [0] SEQUENCE OF SEQUENCE {
        floor-number [0] Unsigned8,
        direction [1] BACnetLiftCarDirection
        }
    }

BACnetEscalatorFault ::= ENUMERATED {
    controller-fault (0),
    drive-and-motor-fault (1),
    mechanical-component-fault (2),
    overspeed-fault (3),
    power-supply-fault (4),
    safety-device-fault (5),
    controller-supply-fault (6),
    drive-temperature-exceeded (7),
    comb-plate-fault (8),
    ...
    }
-- Enumerated values 0-1023 are reserved for definition by ASHRAE. Enumerated values
-- 1024-65535 may be used by others subject to the procedures and constraints described
-- in Clause 23.

BACnetEscalatorMode ::= ENUMERATED {
    unknown (0),
    stop (1),
    up (2),
    down (3),
    inspection (4),
    out-of-service (5),
    ...
    }
-- Enumerated values 0-1023 are reserved for definition by ASHRAE. Enumerated values
-- 1024-65535 may be used by others subject to the procedures and constraints described
-- in Clause 23.

BACnetEscalatorOperationDirection ::= ENUMERATED {
    unknown (0),
    stopped (1),
    up-rated-speed (2),
    up-reduced-speed (3),
    down-rated-speed (4),
    down-reduced-speed (5),
    ...
    }
-- Enumerated values 0-1023 are reserved for definition by ASHRAE. Enumerated values
-- 1024-65535 may be used by others subject to the procedures and constraints described
-- in Clause 23.

BACnetLandingCallStatus ::= SEQUENCE {
    floor-number [0] Unsigned8,
    command CHOICE {
        direction [1] BACnetLiftCarDirection,
        destination [2] Unsigned8
        },
    floor-text [3] CharacterString OPTIONAL
    }

BACnetLandingDoorStatus ::= SEQUENCE {

© ASHRAE (www.ashrae.org). For personal use only. Additional reproduction, distribution,
or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

```
        landing-doors   [0]     SEQUENCE OF SEQUENCE {
                                        floor-number            [0] Unsigned8,
                                        door-status             [1] BACnetDoorStatus
                                        }
        }

BACnetLiftCarCallList ::= SEQUENCE {
        floor-numbers   [0] SEQUENCE OF Unsigned8
        }

BACnetLiftCarDirection ::= ENUMERATED {
        unknown         (0),
        none            (1),
        stopped         (2),
        up              (3),
        down            (4),
        up-and-down     (5),
        ...
        }
-- Enumerated values 0-1023 are reserved for definition by ASHRAE. Enumerated values
-- 1024-65535 may be used by others subject to the procedures and constraints described
-- in Clause 23.

BACnetLiftCarDoorCommand ::= ENUMERATED {
        none            (0),
        open            (1),
        close           (2)
        }


BACnetLiftCarDriveStatus ::= ENUMERATED {
        unknown             (0),
        stationary          (1),
        braking             (2),
        accelerate          (3),
        decelerate          (4),
        rated-speed         (5),
        single-floor-jump   (6),
        two-floor-jump      (7),
        three-floor-jump    (8),
        multi-floor-jump    (9),
        ...
        }
-- Enumerated values 0-1023 are reserved for definition by ASHRAE. Enumerated values
-- 1024-65535 may be used by others subject to the procedures and constraints described
-- in Clause 23.

BACnetLiftCarMode ::= ENUMERATED {
        unknown             (0),
        normal              (1), -- in service
        vip                 (2),
        homing              (3),
        parking             (4),
        attendant-control   (5),
        firefighter-control (6),
        emergency-power     (7),
        inspection          (8),
        cabinet-recall      (9),
        earthquake-operation (10),
        fire-operation      (11),
        out-of-service      (12),
```

© ASHRAE (www.ashrae.org). For personal use only. Additional reproduction, distribution,
or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

```
        occupant-evacuation         (13),
        ...
        }
-- Enumerated values 0-1023 are reserved for definition by ASHRAE. Enumerated values
-- 1024-65535 may be used by others subject to the procedures and constraints described
-- in Clause 23.

BACnetLiftFault ::= ENUMERATED {
        controller-fault                (0),
        drive-and-motor-fault           (1),
        governor-and-safety-gear-fault  (2),
        lift-shaft-device-fault         (3),
        power-supply-fault              (4),
        safety-interlock-fault          (5),
        door-closing-fault              (6),
        door-opening-fault              (7),
        car-stopped-outside-landing-zone (8),
        call-button-stuck               (9),
        start-failure                   (10),
        controller-supply-fault         (11),
        self-test-failure               (12),
        runtime-limit-exceeded          (13),
        position-lost                   (14),
        drive-temperature-exceeded      (15),
        load-measurement-fault          (16),
        ...
        }
-- Enumerated values 0-1023 are reserved for definition by ASHRAE. Enumerated values
-- 1024-65535 may be used by others subject to the procedures and constraints described
-- in Clause 23.

BACnetLiftGroupMode ::= ENUMERATED {
        unknown             (0),
        normal              (1),
        down-peak           (2),
        two-way             (3),
        four-way            (4),
        emergency-power     (5),
        up-peak             (6)
        }
```

[Change **Clause 21**, pp. 669]

```
BACnetDoorStatus ::= ENUMERATED {
        closed          (0),
        opened          (1),
        unknown         (2),
        door-fault      (3),
        unused          (4),
        none            (5),
        closing         (6),
        opening         (7),
        safety-locked   (8),
        limited-opened  (9),
        ...
        }
-- Enumerated values 0-1023 are reserved for definition by ASHRAE. Enumerated values
-- 1024-65535 may be used by others subject to the procedures and constraints described
-- in Clause 23.
```

© ASHRAE (www.ashrae.org). For personal use only. Additional reproduction, distribution,
or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

BACnetObjectType ::= ENUMERATED { -- see below for numerical order

```
    ...
    device                      (8),
    elevator-group              (57),
    escalator                   (58),
    ...
    life-safety-zone            (22),
    lift                        (59),
    ...
    -- see lighting-output      (54),
    -- see elevator-group       (57),
    -- see escalator            (58),
    -- see lift                 (59),
    ...
    }
```

-- 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),
    elevator-group              (57),
    escalator                   (58),
    lift                        (59)
    }
```

BACnetPropertyIdentifier ::= ENUMERATED { -- see below for numerical order

```
    ...
    assigned-access-rights      (256),
    assigned-landing-calls      (447),
    ...
    buffer-size                 (126),
    car-assigned-direction      (448),
    car-door-command            (449),
    car-door-status             (450),
    car-door-text               (451),
    car-door-zone               (452),
    car-drive-status            (453),
    car-load                    (454),
    car-load-units              (455),
    car-mode                    (456),
    car-moving-direction        (457),
    car-position                (458),
    ...
    elapsed-active-time         (33),
    elevator-group              (459),
    energy-meter                (460),
    energy-meter-ref            (461),
    ...
    error-limit                 (34),
    escalator-mode              (462),
    ...
    fault-parameters            (358),
    fault-signals               (463),
    ...
    firmware-revision           (44),
    floor-text                  (464),
```

© ASHRAE (www.ashrae.org). For personal use only. Additional reproduction, distribution,
or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

```
        ...
        global-identifier                   (323),
        group-id                            (465),
        ...
        group-member-names                  (346),
        group-mode                          (467),
        ...
        high-limit                          (45),
        higher-deck                         (468),
        ...
        input-reference                     (181),
        installation-id                     (469),
        ...
        key-sets                            (330),
        landing-calls                       (470),
        landing-call-control                (471),
        landing-door-status                 (472),
        ...
        low-limit                           (59),
        lower-deck                          (473),
        machine-room-id                     (474),
        maintenance-required                (158),
        making-car-call                     (475),
        ...
        network-access-security-policies    (332),
        next-stopping-floor                 (476),
        ...
        occupancy-upper-limit-enforced      (298),
        operation-direction                 (477),
        ...
        passback-timeout                    (301),
        passenger-alarm                     (478),
        ...
        power                               (384),
        power-mode                          (479),
        ...
        record-count                        (141),
        registered-car-call                 (480),
        ...
-- -numerical order reference
        ...
        -- see egress-active                (386),
        -- see assigned-landing-calls       (447),
        -- see car-assigned-direction       (448),
        -- see car-door-command             (449),
        -- see car-door-status              (450),
        -- see car-door-text                (451),
        -- see car-door-zone                (452),
        -- see car-drive-status             (453),
        -- see car-load                     (454),
        -- see car-load-units               (455),
        -- see car-mode                     (456),
        -- see car-moving-direction         (457),
        -- see car-position                 (458),
        -- see elevator-group               (459),
        -- see energy-meter                 (460),
        -- see energy-meter-ref             (461),
        -- see escalator-mode               (462),
        -- see fault-signals                (463),
        -- see floor-text                   (464),
        -- see group-id                     (465),
```

© ASHRAE (www.ashrae.org). For personal use only. Additional reproduction, distribution,
or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

```
        -- see group-mode                    (467),
        -- see higher-deck                   (468),
        -- see installation-id               (469),
        -- see landing-calls                 (470),
        -- see landing-call-control          (471),
        -- see landing-door-status           (472),
        -- see lower-deck                    (473),
        -- see machine-room-id               (474),
        -- see making-car-call               (475),
        -- see next-stopping-floor           (476),
        -- see operation-direction           (477),
        -- see passenger-alarm               (478),
        -- see power-mode                    (479),
        -- see registered-car-call           (480),
        ...
        }
...

BACnetReliability ::= ENUMERATED {
        ...
        tripped                 (15),
        faults-listed           (23),
        ...
        }
-- 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.

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.
        ...
        lighting-transition             [40] BACnetLightingTransition,
        escalator-operation-direction   [49] BACnetEscalatorOperationDirection,
        escalator-fault                 [50] BACnetEscalatorFault,
        escalator-mode                  [51] BACnetEscalatorMode,
        lift-car-direction              [52] BACnetLiftCarDirection,
        lift-car-door-command           [53] BACnetLiftCarDoorCommand,
        lift-car-drive-status           [54] BACnetLiftCarDriveStatus,
        lift-car-mode                   [55] BACnetLiftCarMode,
        lift-group-mode                 [56] BACnetLiftGroupMode,
        lift-fault                      [57] BACnetLiftFault,
        ...
        }
-- 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.
```

© ASHRAE (www.ashrae.org). For personal use only. Additional reproduction, distribution,
or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

[Change **Table 13-5**, p. 473]

Table 13-5. Properties Reported in CHANGE_OF_RELIABILITY Notifications

Object Type	Properties
...	...
Program	Program_State Reason_For_Halt[2,3] Description_Of_Halt[2,3]
Escalator *Lift*	*Fault_Signals*[2]

[1] This value may be excluded from the property-value parameter due to security requirements.
[2] This property is, or may be, from a referenced object. If the value is not known by the event-initiating object, then it shall not be included in the property-value parameter.
[3] These properties are optional and are included only if present in the object.

[Change **Table 23.1**, p. 718]

Table 23-1. Extensible Enumerations

Enumeration Name	Reserved Range	Maximum Value
….	…	…
BACnetLightingTransition	0-63	255
BACnetDoorStatus	*0-1023*	*65535*
BACnetEscalatorFault	*0-1023*	*65535*
BACnetEscalatorMode	*0-1023*	*65535*
BACnetEscalatorOperationDirection	*0-1023*	*65535*
BACnetLiftCarDirection	*0-1023*	*65535*
BACnetLiftCarDriveStatus	*0-1023*	*65535*
BACnetLiftCarMode	*0-1023*	*65535*
BACnetLiftFault	*0-1023*	*65535*

© ASHRAE (www.ashrae.org). For personal use only. Additional reproduction, distribution,
or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

135-2012*aq*-2 Add COV Property Multiple Services

Rationale

See the section 1 rationale for an overview of elevator systems in BACnet. The services added here address the requirements for a large number of values to be subscribed to and for the notifications to have individual timestamps for those data changes.

However, these new services are not limited to elevator applications. They are designed to be general purpose complements to the current COV services and can be used for any property(ies) of any object(s).

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

Table 12-13. Properties of the Device Object Type

Property Identifier	Property Datatype	Conformance Code
...	...	...
Property_List	BACnetARRAY[N] of BACnetPropertyIdentifier	R
Active_COV_Multiple_Subscriptions	*BACnetLIST of BACnetCOVMultipleSubscription*	*O*[x]
Profile_Name	CharacterString	O

...

[x] *This property is required if, and shall be present only if, the device supports execution of the SubscribeCOVPropertyMultiple service.*

[Add new **Clause 12.11.X**, p. 207]

12.11.X Active_COV_Multiple_Subscriptions

This property, of type BACnetLIST of BACnetCOVMultipleSubscription, provides a network-visible indication of those COV-multiple subscriptions that are active at any given time. Each list entry constitutes a COV-multiple context and consists of a Recipient, an Issue Confirmed Notifications flag, a Time Remaining value, a Maximum Notification Delay timeout, and a list of monitored objects. For each monitored object, a list of COV Specifications is present with each containing a Monitored Property reference, an optional COV Increment, and a Timestamped flag. Only one COV-multiple context shall be present in this property for a given Recipient and form of notification indicated by the Issue Confirmed Notifications flag. A Recipient is identified by the BACnet address of the COV-client and the Subscriber Process Identifier.

Whenever a COV-multiple context is created with the SubscribeCOVPropertyMultiple service, a new entry is added to the Active_COV_Multiple_Subscriptions list if no entry is present for the Recipient and form of notification. If an entry exists, COV subscription specifications are added or modified in the list of COV subscription specifications of the entry. Similarly, whenever a COV-multiple subscription is terminated, the corresponding COV subscription specifications of the entry shall be removed. If no COV subscription specifications remain in the entry, or the remaining time indicated in the entry reaches zero, the entire entry is removed from the Active_COV_Multiple_Subscriptions list.

[Change **Clause 13.1**, p. 461]

Change of value (COV) reporting allows a COV-client to subscribe with a COV-server, on a permanent or temporary basis, to receive reports of some changes of value of some referenced property based on fixed criteria. If an object provides COV reporting, then changes of value of any subscribed-to properties of the object, in some cases based on programmable increments, trigger COV notifications to be sent to subscribing clients. Typically, COV notifications are sent to supervisory programs in COV-client devices or to operators or logging devices. Any object, proprietary or standard, may support COV reporting at the implementor's option.

COV subscriptions are established using the SubscribeCOV ~~service or the~~ *service, the* SubscribeCOVProperty ~~service~~ *service, or the SubscribeCOVPropertyMultiple service*. The subscription establishes a connection between the change of value detection and reporting mechanism within the COV-server device and a "process" within the COV-client device. Notifications of changes are issued by the COV-server when changes occur after the subscription has been established. The ConfirmedCOVNotification and UnconfirmedCOVNotification ~~services~~ *services, or the ConfirmedCOVNotificationMultiple and UnconfirmedCOVNotificationMultiple services,* are used by the COV-server to convey change notifications. The choice of confirmed or unconfirmed service is specified in the subscription.

© ASHRAE (www.ashrae.org). For personal use only. Additional reproduction, distribution,
or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

When a BACnet standard object, of a type listed in Table 13-1, supports COV reporting it shall support COV reporting for the property as listed in Table 13-1. At the implementor's discretion, COV reporting may also be supported for any other property of the object. For properties listed in Table 13-1 that have a numeric datatype, the COV increment used to determine when to generate notifications will be the COV_Increment property of the object unless a COV_Increment parameter is supplied in the SubscribeCOVProperty *or SubscribeCOVPropertyMultiple* service. For other properties that have a numeric datatype, the COV increment to use when not supplied with the SubscribeCOVProperty *or SubscribeCOVPropertyMultiple* service shall be a local matter. This is to allow multiple subscribers that do not require a specific increment to use a common increment to allow for the reduction of the processing burden on the COV-server. The criteria for COV reporting for properties other than those listed in Table 13-1 is based on the datatype of the property subscribed to and is described in Table 13-1a.

If an object supports the COV_Period property and COV_Period is non-zero, it shall issue COV notifications to all subscribed recipients at the regular interval specified by COV_Period, in addition to the notifications initiated by the change of value of the monitored property. The value of the monitored property conveyed by the periodic COV notification shall be the basis for determining whether a subsequent COV notification is required by the change in value of the monitored property. If COV_Period is zero, the periodic notifications shall not be issued.

It is the responsibility of the COV-server to maintain the list of active subscriptions for each object that supports COV notification. This list of subscriptions shall be capable of holding at least a single subscription for each object that supports COV notification, although multiple subscriptions may be supported at the implementor's option. The list of subscriptions is network-visible through the device object's Active_COV_Subscriptions property. Subscriptions may be created with finite lifetimes, meaning that the subscription may lapse and be automatically canceled after a period of time. Optionally *with SubscribeCOV,* the lifetime may be specified as infinite, meaning that no automatic cancellation occurs. However, the COV-server is not required to guarantee preservation of subscriptions across power failures or "restarts." Periodic resubscription is allowed and expected and shall simply succeed as if the subscription were new, extending the lifetime of the subscription.

For COV-servers that support SubscribeCOVPropertyMultiple, it is the responsibility of the COV-server to maintain the list of active COV-multiple contexts. A COV-multiple context shall hold all COV-multiple subscription specifications for a COV-client and form of notification, established through one or multiple SubscribeCOVPropertyMultiple requests from that COV-client. This list of COV-multiple contexts shall be capable of holding at least a single subscription for each object that supports COV-multiple notification, although multiple subscriptions may be supported at the implementer's option. The list of COV-multiple contexts is network-visible through the device object's Active_COV_Multiple_Subscriptions property. Subscriptions are created with finite lifetimes, meaning that the subscription will lapse and be automatically canceled after a period of time. The COV-server is not required to guarantee preservation of COV-multiple subscriptions across power failures or "restarts." Periodic resubscription is expected and shall simply succeed as if the subscription were new, extending the lifetime of the subscription. For simplified re-subscription, the COV-client may resubscribe without being required to provide all COV notification specifications again in the SubscribeCOVPropertyMultiple service request for re-subscription. See Clause 13.X.

For COV-multiple subscriptions that request notification of timestamped changes, the COV-server may queue up such timestamped changes and initiate COV-multiple notifications later. When queueing changes, the server shall not delay the sending of notifications longer than 'Max Notification Delay' after the earliest timestamped change in the queue. The notification conveys all timestamped changes to subscribed-to properties since the last COV-multiple notification. Multiple notification messages may be required to convey all changes. If a change occurs to a property that is subscribed to without timestamps in the same COV-multiple context, a notification is initiated immediately and all changes that are currently queued up for timestamped notification shall be sent together with the change that is not timestamped.

The different standard objects that support standardized COV reporting use different criteria for determining that a "change of value" has occurred, which are summarized in Table 13-1. Proprietary object types, or other standard object types not listed in Table 13-1, that support COV reporting of the Present_Value property, should follow these criteria whenever possible. Any objects that may optionally provide COV *or COV-multiple* support and the change of value algorithms they shall employ are summarized in Tables 13-1 and 13-1a.

...

© ASHRAE (www.ashrae.org). For personal use only. Additional reproduction, distribution,
or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

[Add new **Clause 13.X**, p. 533]

13.X SubscribeCOVPropertyMultiple Service

The SubscribeCOVPropertyMultiple service is used by a COV-client to subscribe for the receipt of notifications of changes that may occur to multiple properties of multiple objects. Any object may optionally support COV reporting. If an object provides COV reporting, then changes of value of subscribed-to properties of the object, in some cases based on programmable increments, trigger COV notifications to be sent to one or more subscriber clients. Typically, COV notifications are sent to supervisory programs in BACnet client devices or to operators or logging devices.

The subscription establishes a connection between the change of value detection and reporting mechanism within the COV-server device and a "process" within the COV-client device. Notifications of changes are issued by the COV-server device when changes occur after the subscription has been established. The ConfirmedCOVNotificationMultiple and UnconfirmedCOVNotificationMultiple services are used by the COV-server device to convey change notifications. The choice of confirmed or unconfirmed service is made by the COV-client at the time the subscription is established.

The SubscribeCOVPropertyMultiple service differs from the SubscribeCOVProperty service in that the former allows multiple properties to be monitored via a single subscription request. For establishing or cancelling subscriptions, the effect of the SubscribeCOVPropertyMultiple service is similar to a series of independent SubscribeCOVProperty requests, one difference being that subscription via SubscribeCOVPropertyMultiple results in the use of the ConfirmedCOVNotificationMultiple and UnconfirmedCOVNotificationMultiple services to convey COV notifications of only those properties whose values changed.

The SubscribeCOVPropertyMultiple service also supports requests for timestamped data, for situations where the client requires the time associated with the value change.

13.X.1 Structure

The structure of the SubscribeCOVPropertyMultiple service primitives is shown in Table 13-X1. The terminology and symbology used in this table are explained in Clause 5.6.

Table 13-X1. Structure of SubscribeCOVPropertyMultiple Service Primitives

Parameter Name	Req	Ind	Rsp	Cnf
Argument	M	M(=)		
Subscriber Process Identifier	M	M(=)		
Issue Confirmed Notifications	M	M(=)		
Lifetime	U	U(=)		
Max Notification Delay	U	U(=)		
List of COV Subscription Specifications	M	M(=)		
Result(+)			S	S(=)
Result(-)			S	S(=)
Error Type			S	S(=)
First Failed Subscription			S	S(=)

13.X.1.1 Argument

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

13.X.1.1.1 Subscriber Process Identifier

This parameter, of type Unsigned32, shall convey a numeric "handle" meaningful to the subscriber. This handle shall be used to match future re-subscriptions and cancellations from the subscriber with the COV-multiple context that exists within the COV-server device and with confirmed or unconfirmed COV notifications to identify the process within the COV-client that should receive them.

13.X.1.1.2 Issue Confirmed Notifications

© ASHRAE (www.ashrae.org). For personal use only. Additional reproduction, distribution,
or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

This parameter, of type BOOLEAN, shall convey the form of notification, i.e., whether the COV-server device shall issue ConfirmedCOVNotificationMultiple requests (TRUE) or UnconfirmedCOVNotificationMultiple requests (FALSE) when changes occur. This parameter shall be used to match future re-subscriptions and cancellations from the subscriber with the COV-multiple context that exists within the COV-server device.

13.X.1.1.3 Lifetime

This parameter, of type Unsigned, shall convey the desired lifetime of the subscription in seconds. A value of zero shall not be allowed. A non-zero value shall indicate the number of seconds that may elapse before all subscriptions shall be automatically canceled for the recipient and form of notification.

Devices that execute this service shall accept, at a minimum, lifetime values up to and including 28800 seconds (8 hours). Devices may optionally support lifetime values larger than 28800. Devices that initiate this service shall be capable of providing Lifetime values less than or equal to 28800.

This parameter shall be present if the request is a subscription, or re-subscription, and absent if the request is a cancellation.

13.X.1.1.4 Max Notification Delay

This parameter, of type Unsigned, shall convey the maximum notification delay for the subscription in seconds. This parameter indicates the maximum number of seconds that may elapse before a notification is issued if changes of properties occurred for which the 'Timestamped' parameter is TRUE and which were queued up for notification. The value of this parameter shall be less than the value of the Lifetime parameter.

The range for this parameter shall be 0 to 3600 seconds.

This parameter shall be present if the request is a subscription, or re-subscription, and absent if the request is a cancellation.

13.X.1.1.5 List of COV Subscription Specifications

This parameter shall consist of a list of zero or more 'COV Subscription Specifications'. Each specification shall consist of two parameters: (1) an 'Object Identifier' and (2) a 'List of COV References'. See Clause 13.X.3.1.

This parameter may be an empty list for re-subscriptions that only update the lifetime of the COV-multiple subscription or for a cancellation of a subscription.

13.X.1.2 Result(+)

The 'Result(+)' parameter shall indicate that the requested service has succeeded.

13.X.1.3 Result(-)

The 'Result(-)' parameter shall indicate that the service request has failed. The reason for failure shall be specified by the 'Error Type' parameter in the case of a general error in executing the service, or by the 'First Failed Subscription' parameter if an error occurred in processing the 'List of COV Subscription Specifications' parameter.

13.X.1.3.1 Error Type

This parameter shall be used to report general service execution errors not related to a specific COV subscription specification of the request. See Clause 13.X.3.2.1.

13.X.1.3.2 First Failed Subscription

This parameter shall consist of three parameters: (1) an 'Object Identifier', (2) a 'Property Reference', and (3) an 'Error' parameter, indicating the reason for failure of the subscription. See Clause 13.X.3.2.2.

13.X.2 Service Procedure

If the parameters 'Lifetime' and 'Max Notification Delay' are not present, then the request shall be considered to be a cancellation. All COV subscription specifications that already exist for the same recipient and form of notification contained

© ASHRAE (www.ashrae.org). For personal use only. Additional reproduction, distribution,
or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

in the PDU that carries the SubscribeCOVPropertyMultiple request (i.e., a COV-multiple context exists) and that appear in the list of 'COV Subscription Specifications' shall be removed from the respective COV-multiple context and a 'Result(+)' returned. If no COV subscription specifications remain in a COV-multiple context, the COV-multiple context is removed in its entirety. If the list of COV subscription specifications conveyed in a cancellation is empty, the COV-multiple context shall be removed. Cancellations that are issued for which no matching COV-multiple context, or no COV subscription specifications in an existing COV-multiple context can be found shall succeed as if a COV-multiple context or COV subscription specification had existed and was removed.

If the 'Lifetime' and 'Max Notification Delay' parameters are present, the COV subscriptions and the COV references present in the request shall be processed in the order specified in the request. If any of the properties to be monitored do not support COV-multiple reporting, then a 'Result(-)' shall be returned. If the properties to be monitored do support COV-multiple reporting, then a check shall be made to locate an existing COV-multiple context for the same recipient and form of notification contained in the PDU that carries the SubscribeCOVPropertyMultiple request. If an existing COV-multiple context is found, then this request shall be considered a re-subscription and shall succeed as if the subscription had been newly created. For COV subscription specifications conveyed in the request for which no COV subscription specifications exist in the COV-multiple context, respective COV subscription specifications shall be added to the COV-multiple context.

If the 'Lifetime' and 'Max Notification Delay' parameters are present, and no COV-multiple context can be found that matches the source BACnet address, the recipient process and the form of notification conveyed in the subscription request, then a new COV-multiple context shall be established that contains the BACnet address from the PDU that carries the SubscribeCOVPropertyMultiple request, the 'Subscriber Process Identifier', the 'Issue Confirmed Notifications' flag, the 'Lifetime', and the 'Max Notification Delay' timeout. All COV subscription specifications conveyed in the request shall be added to the COV-multiple context. The new COV-multiple context shall be included in the Active_COV_Multiple_Subscriptions property of the Device object. If no COV-multiple context can be created, then a 'Result(-)' shall be returned.

If a new COV-multiple context is created, or re-subscribed, then it shall be initialized and given a lifetime as specified by the 'Lifetime' parameter. The subscription shall be automatically canceled and the COV-multiple context shall be removed after ’Lifetime’ seconds have elapsed unless a re-subscription is received.

A 'Result(+)' shall be returned and a ConfirmedCOVNotificationMultiple or one or multiple UnconfirmedCOVNotificationMultiple notifications shall be issued for each COV subscription specification specified in the SubscribeCOVPropertyMultiple request as soon as possible after the successful completion of a subscription or re-subscription request, as specified by the 'Issue Confirmed Notifications' parameter.

A Result(-) with parameter 'Error Type' shall be returned in case of a failure in processing the request before any COV subscription specification and COV reference has been processed. The Result(-) shall convey the 'Error Class' and 'Error Code' indicating the error occurred. No COV-multiple notifications shall be issued in this case.

A 'Result(-)' with parameter 'First Failed Subscription' shall be returned in the case that an error occurred in processing a particular COV subscription specification and COV reference of the request. Remaining COV references, if any, and the remaining COV subscription specifications shall not be processed. The 'Result(-)' shall convey the object identifier of the failing COV subscription specification in parameter 'Monitored Object Identifier', the property reference of the failing COV reference in parameter 'Monitored Property Reference', and the error type of the error that occurred for this COV subscription specification in parameter 'Error Type'. COV-multiple notifications shall be issued for those COV subscription specifications and COV references already processed successfully, as defined for the 'Result(+)' case.

13.X.3 Parameters Referenced by the SubscribeCOVPropertyMultiple Service

The following parameters appear in the SubscribeCOVPropertyMultiple service primitives.

13.X.3.1 COV Subscription Specification Parameter

The 'COV Subscription Specification' parameter is shown in Table 13-X2. The terminology and symbology used in this table are explained in Clause 5.6.

Table 13-X2. Structure of 'COV Subscription Specification' Parameter

Parameter Name	Req	Ind	Datatype
Monitored Object	M	M(=)	BACnetObjectIdentifier
List of COV References	M	M(=)	
Monitored Property	M	M(=)	BACnetPropertyReference

© ASHRAE (www.ashrae.org). For personal use only. Additional reproduction, distribution,
or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

COV Increment	U	U(=)	REAL
Timestamped	M	M(=)	BOOLEAN

13.X.3.1.1 Monitored Object

This parameter, of type BACnetObjectIdentifier, shall convey the identifier of the object within the receiving device that contains one or more properties for which a subscription is desired.

13.X.3.1.2 List of COV References

This parameter shall be a list of one or more 'COV References', each of which corresponds directly to a specific property of the object identified above.

Each 'COV Reference' shall convey the property identifier and optional array index for which a subscription is desired. If COV reporting is supported for a property that has an array datatype, it is a local matter to determine whether to support COV subscriptions for all elements of the array or only for particular elements in the array. The property identifier in a 'COV Reference' shall not be one of the special property identifiers ALL, REQUIRED, or OPTIONAL.

13.X.3.1.2.1 Monitored Property

This parameter, of type BACnetPropertyReference, shall convey the property identifier and optional array index for which a subscription is desired.

13.X.3.1.2.2 COV Increment

The 'COV Increment' parameter, of type REAL, shall specify the minimum change in the monitored property that will cause a COV notification to be queued up or issued to subscriber COV-clients. This parameter shall be ignored if the datatype of the monitored property is not REAL or array of REAL.

If the monitored property is Present_Value, its datatype is REAL, this parameter is not present, and the monitored object has a COV_Increment property, then the COV increment to use is taken from the COV_Increment property of the monitored object. Otherwise, if this parameter is not present, the value used for the COV increment shall be a local matter. The intent is to allow the subscriber to use a previously established COV increment from another subscription or to allow use of the COV_Increment property in the monitored object.

13.X.3.1.2.3 Timestamped

This parameter, of type BOOLEAN, shall convey whether the COV-server device shall timestamp (TRUE) or not (FALSE) the changes that occur to the monitored property. If TRUE, the COV-server shall queue up all changes of the monitored property until a COV-multiple notification is sent to the subscriber, and shall, in this notification, provide all changes that were queued up to be sent in the COV-server.

If FALSE, the COV-server shall initiate a COV-multiple notification when the value of the property changes, by 'COV-Increment' if one is specified, and include all pending timestamped changes in the notification that are queued up for the subscriber.

13.X.3.2 Parameters Referenced by the Result(-) Error Return

The following parameters appear in the Result(-) error return, as shown in Table 13-X1.

13.X.3.2.1 Error Type

This parameter shall consist of two components: (1) 'Error Class' and (2) 'Error Code'. See Clause 18.

© ASHRAE (www.ashrae.org). For personal use only. Additional reproduction, distribution,
or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

The 'Error Class' and 'Error Code' to be returned for specific situations are as follows:

Situation	Error Class	Error Code
No COV-multiple context can be created due to resource limitations	RESOURCES	NO_SPACE_TO_ADD_LIST_ELEMENT
The 'Lifetime' parameter is outside of the range supported by the device	SERVICES	VALUE_OUT_OF_RANGE
The 'Max Notification Delay' parameter is outside of the range supported by the device.	SERVICES	VALUE_OUT_OF_RANGE
The 'Max Notification Delay' parameter is greater than the 'Lifetime' parameter.	SERVICES	VALUE_OUT_OF_RANGE

13.X.3.2.2 First Failed Subscription

This parameter identifies the first failed subscription, corresponding to a specific object and property identified in the request for which the subscription failed. The terminology and symbology used in this table are explained in Clause 5.6.

Table 13-X3. Structure of 'Subscription Failure' Parameter

Parameter Name	Rsp	Cnf	Datatype
Monitored Object Identifier	M	M(=)	BACnetObjectIdentifier
Monitored Property Reference	M	M(=)	BACnetPropertyReference
Error Type	M	M(=)	Error

13.X.3.2.2.1 Monitored Object Identifier

This parameter, of type BACnetObjectIdentifier, shall convey the identifier of an object containing one or more properties for which a subscription failed.

13.X.3.2.2.2 Monitored Property Reference

This parameter, of type BACnetPropertyReference, shall convey the property identifier and array index (if one appeared in the corresponding request) for which a subscription failed.

13.X.3.2.2.3 Error Type

This parameter shall consist 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 in processing an individual subscription for a property are as follows:

Situation	Error Class	Error Code
Specified object does not exist	OBJECT	UNKNOWN_OBJECT
Specified property does not exist	PROPERTY	UNKNOWN_PROPERTY
Specified object does not support COV-multiple notifications	OBJECT	OPTIONAL_FUNCTIONALITY_NOT_SUPPORTED
Specified property does not support COV-multiple notifications	PROPERTY	NOT_COV_PROPERTY
An array index is provided but the property is not an array.	PROPERTY	PROPERTY_IS_NOT_AN_ARRAY
An array index is provided that is outside the range existing in the property.	PROPERTY	INVALID_ARRAY_INDEX
No COV subscription specification can be created in the COV-multiple context due to resource limitations	RESOURCES	NO_SPACE_TO_ADD_LIST_ELEMENT

© ASHRAE (www.ashrae.org). For personal use only. Additional reproduction, distribution,
or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

[Add new **Clause 13.Y,** p. 533]

13.Y ConfirmedCOVNotificationMultiple Service

The ConfirmedCOVNotificationMultiple service is used to notify subscribers about changes that may have occurred to one or more properties of one or more objects. Subscriptions for ConfirmedCOVNotificationMultiple are made using the SubscribeCOVPropertyMultiple service.

13.Y.1 Structure

The structure of the ConfirmedCOVNotificationMultiple service primitives is shown in Table 13-Y1. The terminology and symbology used in this table are explained in Clause 5.6.

Table 13-Y1. Structure of ConfirmedCOVNotificationMultiple Service Primitives

Parameter Name	Req	Ind	Rsp	Cnf
Argument	M	M(=)		
Subscriber Process Identifier	M	M(=)		
Initiating Device Identifier	M	M(=)		
Time Remaining	M	M(=)		
Timestamp	U	U(=)		
List of COV Notifications	M	M(=)		
Result(+)			S	S(=)
Result(-)			S	S(=)
Error Type			M	M(=)

13.Y.1.1 Argument

This parameter shall convey the parameters for the ConfirmedCOVNotificationMultiple service request.

13.Y.1.1.1 Subscriber Process Identifier

This parameter, of type Unsigned32, shall convey a numeric "handle" meaningful to the subscriber. This handle shall be used to identify the process within the COV client that should receive the notification.

13.Y.1.1.2 Initiating Device Identifier

This parameter, of type BACnetObjectIdentifier, shall convey the Device Object_Identifier of the device that initiated the ConfirmedCOVNotificationMultiple service request.

13.Y.1.1.3 Time Remaining

This parameter, of type Unsigned, shall convey the remaining lifetime of the COV-multiple subscription, in seconds.

13.Y.1.1.4 Timestamp

This parameter, of type BACnetDateTime, shall convey the date and time of the last change conveyed in the notification. This parameter shall be present if, and only if, any of the COV notifications conveyed in the 'List of COV Notifications' parameter contain the 'Time of Change' parameter.

13.Y.1.1.5 List of COV Notifications

This parameter shall be a list of one or more 'COV Notifications', each of which corresponds directly to a specific property of the object identified and whose value has changed, resulting in the COV notification to be issued. See 13.Y.3.1.

The ConfirmedCOVNotificationMultiple request may convey COV notifications from any objects and properties that are subscribed to by the same BACnet address, the same Process Identifier, and requested to issue ConfirmedCOVNotificationMultiple notifications. These subscriptions may have been established through one or multiple SubscribeCOVPropertyMultiple requests. The ConfirmedCOVNotificationMultiple request shall also contain all timestamped changes of properties that were queued up since the last notification sent to the subscriber specified by the respective COV-Multiple context.

13.Y.1.5 Result(+)

The 'Result(+)' parameter shall indicate that the requested service has succeeded.

© ASHRAE (www.ashrae.org). For personal use only. Additional reproduction, distribution,
or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

13.Y.1.6 Result(-)
The 'Result(-)' parameter shall indicate that the service request has failed. The reason for failure shall be specified by the 'Error Type' parameter.

13.Y.1.6.1 Error Type
This parameter shall consist 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
No subscription exists for one of the specified objects, properties, and process identifier. Devices may ignore this condition and return a BACnet-SimpleACK-PDU.	SERVICES	UNKNOWN_SUBSCRIPTION

13.Y.2 Service Procedure
After verifying the validity of the request, the responding BACnet-user shall take whatever local actions have been assigned to the indicated COV subscriptions and issue a 'Result(+)' service primitive. If the service request fails in its entirety or for any COV notification conveyed, a 'Result(-)' service primitive may be issued indicating the error encountered.

13.Y.3 Parameters Referenced by the ConfirmedCOVNotificationMultiple Service
The following parameters appear in the ConfirmedCOVNotificationMultiple service primitives.

13.Y.3.1 COV Notification Parameter
The 'COV Notification' parameter is shown in Table 13-Y2. The terminology and symbology used in this table are explained in Clause 5.6.

Table 13-Y2. Structure of 'COV Notification' Parameter

Parameter Name	Req	Ind	Datatype
Monitored Object Identifier	M	M(=)	BACnetObjectIdentifier
List of Values	M	M(=)	
Property Identifier	M	M(=)	BACnetPropertyIdentifier
Property Array Index	U	U(=)	Unsigned
Value	M	M(=)	ANY
Time of Change	U	U(=)	Time

13.Y.3.1.1 Monitored Object Identifier
This parameter, of type BACnetObjectIdentifier, shall convey the Object_Identifier of an object with one or more properties subscribed via SubscribeCOVPropertyMultiple and whose values have changed.

13.Y.3.1.2 List of Values
This parameter shall convey a list of one or more property values that have changed, possibly conveyed as timestamped data. This parameter may include multiple entries for the same property if the 'Timestamped' flag is TRUE in the respective COV subscription specification and multiple changes of that property have been queued up since the last notification.

13.Y.3.1.2.1 Property Identifier
This parameter, of type BACnetPropertyIdentifier, shall convey the property identifier of the property.

13.Y.3.1.2.2 Property Array Index
If, and only if, the subscription corresponding to this notification specified an array index and the property is of datatype array, this parameter, of type Unsigned, shall convey that array index.

13.Y.3.1.2.3 Value
This parameter, of type ANY, shall convey the new value of the property.

13.Y.3.1.2.4 Time of Change
This parameter, of type Time, shall be present if, and only if, the SubscribeCOVPropertyMultiple request was made with its 'Timestamped' parameter set to TRUE in the COV subscription specification for the property and shall convey the local time when the data value changed.

© ASHRAE (www.ashrae.org). For personal use only. Additional reproduction, distribution,
or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

[Add new **Clause 13.Z,** p. 533]

13.Z UnconfirmedCOVNotificationMultiple Service

The UnconfirmedCOVNotificationMultiple service is used to notify subscribers about changes that may have occurred to one or more properties of one or more objects, or to distribute object properties of wide interest (such as outside air conditions) to many devices simultaneously without a subscription. Subscriptions for UnconfirmedCOVNotificationMultiple are made using the SubscribeCOVPropertyMultiple service. For unsubscribed notifications, the algorithm for determining when to issue this service is a local matter and may be based on a change of value, periodic updating, or some other criteria.

13.Z.1 Structure

The structure of the UnconfirmedCOVNotificationMultiple service primitives is shown in Table 13-Z1. The terminology and symbology used in this table are explained in Clause 5.6.

Table 13-Z1. Structure of UnconfirmedCOVNotificationMultiple Service Primitives

Parameter Name	Req	Ind
Argument	M	M(=)
Subscriber Process Identifier	M	M(=)
Initiating Device Identifier	M	M(=)
Time Remaining	M	M(=)
Timestamp	U	U(=)
List of COV Notifications	M	M(=)

13.Z.1.1 Argument

This parameter shall convey the parameters for the UnconfirmedCOVNotificationMultiple service request.

13.Z.1.1.1 Subscriber Process Identifier

This parameter, of type Unsigned32, shall convey a numeric "handle" meaningful to the subscriber. This handle shall be used to identify the process within the COV client that should receive the notification. The value zero is reserved for unsubscribed COV notifications.

13.Z.1.1.2 Initiating Device Identifier

This parameter, of type BACnetObjectIdentifier, shall convey the Device Object_Identifier of the device that initiated the UnconfirmedCOVNotificationMultiple service request.

13.Z.1.1.3 Time Remaining

This parameter, of type Unsigned, shall convey the remaining lifetime of the COV-multiple subscription, in seconds. A value of zero shall be used in unsubscribed notifications.

13.Z.1.1.4 Timestamp

This parameter, of type BACnetDateTime, shall convey the date and time of the last change conveyed in the notification. This parameter shall be present if, and only if, any of the COV notifications conveyed in the 'List of COV Notifications' contain the 'Time of Change' value.

13.Z.1.1.5 List of COV Notifications

This parameter shall be a list of one or more 'COV Notifications', each of which corresponds directly to a specific property of the object identified and whose value has changed, resulting in the COV notification to be issued. See Clause 13.Z.3.1

The UnconfirmedCOVNotificationMultiple request may convey COV notifications from any objects and properties that are subscribed to by the same BACnet address, the same Process Identifier, and requesting to issue UnconfirmedCOVNotificationMultiple notifications. These subscriptions may have been established through one or multiple SubscribeCOVPropertyMultiple requests. The UnconfirmedCOVNotificationMultiple request shall also contain all timestamped changes of properties that were queued up since the last notification sent to the subscriber specified by the respective COV-multiple context.

If the number of changes exceeds the number of changes conveyable in a single notification request, the COV-server shall initiate as many UnconfirmedCOVNotificationMultiple requests that are required to notify all changes.

13.Z.2 Service Procedure

© ASHRAE (www.ashrae.org). For personal use only. Additional reproduction, distribution,
or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

Since this is an unconfirmed service, no response primitives are expected. Actions taken in response to this notification are a local matter.

13.Z.3 Parameters Referenced by the UnconfirmedCOVNotificationMultiple Service

The following parameters appear in the UnconfirmedCOVNotificationMultiple service primitives.

13.Z.3.1 COV Notification Parameter

The 'COV Notification' parameter is shown in Table 13-Z2. The terminology and symbology used in this table are explained in Clause 5.6.

Table 13-Z2. Structure of 'COV Notification' Parameter

Parameter Name	Req	Ind	Datatype
Monitored Object Identifier	M	M(=)	BACnetObjectIdentifier
List of Values	M	M(=)	
Property Identifier	M	M(=)	BACnetPropertyIdentifier
Property Array Index	U	U(=)	Unsigned
Value	M	M(=)	ANY
Time of Change	U	U(=)	Time

13.Z.3.1.1 Monitored Object Identifier

This parameter, of type BACnetObjectIdentifier, shall convey the Object_Identifier of an object with one or more properties subscribed via SubscribeCOVPropertyMultiple and whose values have changed.

13.Z.3.1.2 List of Values

This parameter shall convey a list of one or more property values that have changed, possibly conveyed as timestamped data. This parameter may include multiple entries for the same property if the 'Timestamped' flag is TRUE in the respective COV subscription specification and multiple changes of that property have been queued up since the last notification.

13.Z.3.1.2.1 Property Identifier

This parameter, of type BACnetPropertyIdentifier, shall convey the property identifier of the property.

13.Z.3.1.2.2 Property Array Index

If, and only if, the subscription corresponding to this notification specified an array index and the property is of datatype array, this parameter, of type Unsigned, shall convey that array index.

13.Z.3.1.2.3 Value

This parameter, of type ANY, shall convey the new value of the property.

13.Z.3.1.2.4 Time of Change

This parameter, of type Time, shall be present if, and only if, the SubscribeCOVPropertyMultiple request for the property was made with its 'Timestamped' parameter set to TRUE in the COV subscription specification for the property and shall convey the local time when the data value changed.

© ASHRAE (www.ashrae.org). For personal use only. Additional reproduction, distribution,
or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

[Change **Clause 21**, p. 641]

BACnetConfirmedServiceChoice ::= ENUMERATED {
-- Alarm and Event Services
...
lifeSafetyOperation (27),
subscribeCOVPropertyMultiple (30),
confirmedCOVNotificationMultiple (31),
...
-- *Services added after 2012*
-- *subscribeCOVPropertyMultiple (30), see Alarm and Event Services*
-- *confirmedCOVNotificationMultiple (31) see Alarm and Event Services*
}

[Change Clause **21**, p. 642]

BACnet-Confirmed-Service-Request ::= CHOICE {
-- Alarm and Event Services
...
subscribeCOVPropertyMultiple [30] SubscribeCOVPropertyMultiple-Request,
confirmedCOVNotificationMultiple [31] ConfirmedCOVNotificationMultiple-Request,
...
-- *Services added after 2012*
-- *subscribeCOVPropertyMultiple [30] see Alarm and Event Services*
-- *confirmedCOVNotificationMultiple [31] see Alarm and Event Services*
}

[Add new productions to **Clause 21**, pp. 643-645, arranged alphabetically in section "Confirmed Alarm and Event Services"]

ConfirmedCOVNotificationMultiple-Request ::= SEQUENCE {
subscriberProcessIdentifier [0] Unsigned32,
initiatingDeviceIdentifier [1] BACnetObjectIdentifier,
timeRemaining [2] Unsigned,
timestamp [3] BACnetDateTime OPTIONAL,
listOfCOVNotifications [4] SEQUENCE OF SEQUENCE {
monitoredObject [0] BACnetObjectIdentifier,
listOfValues [1] SEQUENCE OF SEQUENCE {
propertyIdentifier [0] BACnetPropertyIdentifier,
propertyArrayIndex [1] Unsigned OPTIONAL,
value [2] ABSTRACT-SYNTAX.&Type,
timeOfChange [3] Time OPTIONAL
}
}
}

SubscribeCOVPropertyMultiple-Request ::= SEQUENCE {
subscriberProcessIdentifier [0] Unsigned32,
issueConfirmedNotifications [1] BOOLEAN,
lifetime [2] Unsigned OPTIONAL,
maxNotificationDelay [3] Unsigned OPTIONAL,
listOfCOVSubscriptionSpecifications [4] SEQUENCE OF SEQUENCE {
monitoredObject [0] BACnetObjectIdentifier,
listOfCOVReferences [1] SEQUENCE OF SEQUENCE {
monitoredProperty [0] BACnetPropertyReference,
covIncrement [1] REAL OPTIONAL,
timestamped [2] BOOLEAN
}
}

© ASHRAE (www.ashrae.org). For personal use only. Additional reproduction, distribution,
or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

}

[Change Clause **21**, p. 650]

BACnetUnconfirmedServiceChoice ::= ENUMERATED {
...
utcTimeSynchronization (9),
writeGroup ~~(10)~~*(10),*
unconfirmedCOVNotificationMultiple (11)
}

[Change Clause **21**, p. 650]

BACnet-Unconfirmed-Service-Request ::= CHOICE {
...
utcTimeSynchronization [9] UnconfirmedCOVNotification-Request
writeGroup [10] ~~WriteGroup-Request~~ *WriteGroup-Request,*
unconfirmedCOVNotificationMultiple [11] UnconfirmedCOVNotificationMultiple-Request
}

[Add new production in **Clause 21**, p. 650, arranged alphabetically in section "Unconfirmed Alarm and Event Services"]

UnconfirmedCOVNotificationMultiple-Request ::= SEQUENCE {
subscriberProcessIdentifier [0] Unsigned32,
initiatingDeviceIdentifier [1] BACnetObjectIdentifier,
timeRemaining [2] Unsigned,
timestamp [3] BACnetDateTime OPTIONAL,
listOfCOVNotifications [4] SEQUENCE OF SEQUENCE {
monitoredObject [0] BACnetObjectIdentifier,
listOfValues [1] SEQUENCE OF SEQUENCE{
propertyIdentifier [0] BACnetPropertyIdentifier,
propertyArrayIndex [1] Unsigned OPTIONAL,
value [2] ABSTRACT-SYNTAX.&Type,
timeOfChange [3] Time OPTIONAL
}
}
}

[Change **Clause 21**, p. 652-653]

BACnet-Error ::= CHOICE {
...
confirmedCOVNotification [1] Error,
subscribeCOVPropertyMultiple [30] SubscribeCOVPropertyMultiple-Error,
confirmedCOVNotificationMultiple [31] Error,
...
-- Services added after 2012
-- subscribeCOVPropertyMultiple [30] see Alarm and Event Services
--confirmedCOVNotificationMultiple [31] see Alarm and Event Services
}

© ASHRAE (www.ashrae.org). For personal use only. Additional reproduction, distribution,
or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

[Add new production in **Clause 21**, p. 652, arranged alphabetically in section "Error Productions"]

SubscribeCOVPropertyMultiple-Error ::= CHOICE {
    error-type [0] Error,
    first-failed-subscription [1] SEQUENCE {
        monitoredObjectIdentifier [0] BACnetObjectIdentifier,
        monitoredPropertyReference [1] BACnetPropertyReference,
        errorType [2] Error
        }
    }

[Change **Clause 21**, p. 710]

BACnetServicesSupported ::= BIT STRING {
-- Alarm and Event Services
    ...
    confirmedCOVNotification (1),
    -- *confirmedCOVNotificationMultiple* *(42)*,
    ...
    -- subscribeCOVProperty (38),
    -- *subscribeCOVPropertyMultiple* *(41)*,
    ...
-- Unconfirmed Services
    ...
    unconfirmedCOVNotification (28),
    -- *unconfirmedCOVNotificationMultiple* *(43)*,
    ...
-- Services added after 1995
    ...
    writeGroup ~~(40)~~ *(40)*, -- Object Access Services
-- *Services added after 2012*
    subscribeCOVPropertyMultiple *(41)*, -- *Alarm and Event Service*
    confirmedCOVNotificationMultiple *(42)*, -- *Alarm and Event Service*
    unconfirmedCOVNotificationMultiple *(43)* -- *Alarm and Event Service*
    }

[Add new production **BACnetCOVMultipleSubcription** in **Clause 21**, p. 652, arranged alphabetically]

BACnetCOVMultipleSubscription ::= SEQUENCE {
    recipient [0] BACnetRecipientProcess,
    issueConfirmedNotifications [1] BOOLEAN,
    timeRemaining [2] Unsigned,
    maxNotificationDelay [3] Unsigned,
    listOfCOVSubscriptionSpecifications [4] SEQUENCE OF SEQUENCE {
        monitoredObject [0] BACnetObjectIdentifier,
        listOfCOVReferences [1] SEQUENCE OF SEQUENCE {
            monitoredProperty [0] BACnetPropertyReference,
            covIncrement [1] REAL OPTIONAL,
            timestamped [2] BOOLEAN
            }
        }
    }

[Change production **BACnetPropertyIdentifier** in **Clause 21**, p. 694]

BACnetPropertyIdentifier ::= ENUMERATED { -- see below for numerical order
    ...
    active-authentication-policy (255),
    active-cov-multiple-subscriptions *(481)*,
    active-cov-subscriptions (152),

© ASHRAE (www.ashrae.org). For personal use only. Additional reproduction, distribution,
or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

```
        ...
        -- see egress-active                    ~~(386)~~(386),
        -- see active-cov-multiple-subscriptions (481)
        ...
        }
-- The special property identifiers all, optional, and required are reserved for use in the
-- ReadPropertyMultiple service 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.
```

[Change **Clause 24.12.2**, p. 754]

24.12.2 Securing Response Messages

...

COV notifications shall be secured using the same key and to the same level as the ~~SubscribeCOV or SubscribeCOVProperty~~ request that created the subscription.

...

© ASHRAE (www.ashrae.org). For personal use only. Additional reproduction, distribution,
or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

[Add new **Clause E.1.X1-X3**, p. 785]

E.1.X1 Example of the SubscribeCOVPropertyMultiple Service

This example illustrates the use of the SubscribeCOVPropertyMultiple service to subscribe to COV notifications on properties of two objects.

Service =	SubscribeCOVPropertyMultiple
'Subscriber Process Identifier' =	18
'Issue Confirmed Notifications' =	TRUE
'Lifetime' =	60
'Max Notification Delay' =	5
'List of COV Subscription Specifications' =	(((Analog Input, Instance 10), ((PRESENT_VALUE, 1.0, TRUE), (RELIABILITY, , FALSE))), ((Analog Output, Instance 8), ((PRESENT_VALUE, 0.1, TRUE))))

E.1.X2 Example of the ConfirmedCOVNotificationMultiple Service

The following example illustrates a notification that two of the properties for which COV notifications were subscribed using the SubscribeCOVPropertyMultiple service have changed. The first property was subscribed with 'Timestamped' set to TRUE, the second with 'Timestamped' set to FALSE.

Service =	ConfirmedCOVNotificationMultiple
'Subscriber Process Identifier' =	18
'Initiating Device Identifier' =	(Device, Instance 4)
'Time Remaining' =	35
'Timestamp' =	((June 3, 2013 (Day of Week = Monday)), (03:23:53.47))
'List of COV Notifications' =	(((Analog Input, Instance 10), ((Present_Value, , 65.0, (03:23:52.00)))), ((Analog Output, Instance 8), ((Present_Value, ,80.1,))))

E.1.X3 Example of the UnconfirmedCOVNotificationMultiple Service

The following example illustrates a notification that one of the properties for which COV notifications were subscribed using the SubscribeCOVPropertyMultiple service has changed. Since no timestamped change of value is conveyed, the 'Timestamp' parameter is not present.

Service =	UnconfirmedCOVNotificationMultiple
'Subscriber Process Identifier' =	18
'Initiating Device Identifier' =	(Device, Instance 4)
'Time Remaining' =	27
'List of COV Notifications' =	(((Analog Input, Instance 10), ((Present_Value, , 65.0,))))

© ASHRAE (www.ashrae.org). For personal use only. Additional reproduction, distribution,
or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

[Add new **Clause F.1.X1-X3**, p. 807]

F.1.X1 Encoding for Example E.1.X1 - SubscribeCOVPropertyMultiple Service

```
X'00'           PDU Type=0 (BACnet-Confirmed-Request-PDU, SEG=0, MOR=0, SA=0)
X'02'           Maximum APDU Size Accepted=206 octets
X'0F'           Invoke ID=15
X'1E'           Service Choice=30 (SubscribeCOVPropertyMultiple-Request)

X'09'           SD Context Tag 0 (Subscriber Process Identifier, L=1)
X'12'           18
X'19'           SD Context Tag 1 (Issue Confirmed Notifications, L=1)
X'01'           1 (TRUE)
X'29'           SD Context Tag 2 (Lifetime, L=1)
X'3C'           60
X'39'           SD Context Tag 3 (Max Notification Delay, L=1)
X'05'           5
X'4E'           PD Opening Tag 4 (List of COV Subscription Specifications)
      X'0C'           SD Context Tag 0 (Monitored Object, L=4)
      X'0000000A'           Analog Input, Instance Number=10
      X'1E'                 PD Opening Tag 1 (List of COV References)
            X'0E'                   PD Opening Tag 0 (Monitored Property)
                  X'09'                     SD Context Tag 0 (Property Identifier, L=1)
                  X'55'                     85 (PRESENT_VALUE)
            X'0F'                   PD Closing Tag 0 (Monitored Property)
            X'1C'                   SD Context Tag 1 (COV Increment, L=4)
            X'3F800000'             1.0
            X'29'                   SD Context Tag 2 (Timestamped, L=1)
            X'01'                   TRUE
            X'0E'                   PD Opening Tag 0 (Monitored Property)
                  X'09'                     SD Context Tag 0 (Property Identifier, L=1)
                  X'67'                     103 (RELIABILITY)
            X'0F'                   PD Closing Tag 0 (Monitored Property)
            X'29'                   SD Context Tag 2 (Timestamped, L=1)
            X'00'                   FALSE
      X'1F'                 PD Closing Tag 1 (List of COV References)
      X'0C'                 SD Context Tag 0 (Monitored Object, L=4)
      X'00400008'           Analog Output, Instance Number=8
      X'1E'                 PD Opening Tag 1 (List of COV References)
            X'0E'                   PD Opening Tag 0 (Monitored Property)
                  X'09'                     SD Context Tag 0 (Property Identifier, L=1)
                  X'55'                     85 (PRESENT_VALUE)
            X'0F'                   PD Closing Tag 0 (Monitored Property)
            X'1C'                   SD Context Tag 1 (COV Increment, L=4)
            X'3F800000'             1.0
            X'29'                   SD Context Tag 2 (Timestamped, L=1)
            X'01'                   TRUE
      X'1F'                 PD Closing Tag 1 (List of COV References)
X'4F'           PD Closing Tag 4 (List of COV Subscription Specifications)
```

Assuming the service procedure executes correctly, a simple acknowledgment is returned:

```
X'20'           PDU Type=2 (BACnet-SimpleACK-PDU)
X'0F'           Invoke ID=15
X'1E'           Service ACK Choice=30 (SubscribeCOVPropertyMultiple)
```

F.1.X2 Encoding for Example E.1.X2 - ConfirmedCOVNotificationMultiple Service

```
X'00'                   PDU Type=0 (BACnet-Confirmed-Request-PDU, SEG=0, MOR=0, SA=0)
X'02'                   Maximum APDU Size Accepted=206 octets
```

© ASHRAE (www.ashrae.org). For personal use only. Additional reproduction, distribution,
or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

```
X'0F'                   Invoke ID=15
X'1F'                   Service Choice=31 (ConfirmedCOVNotificationMultiple-Request)

X'09'                   SD Context Tag 0 (Subscriber Process Identifier, L=1)
X'12'                   Subscriber Process Identifier=18
X'1C'                   SD Context Tag 1 (Initiating Device Identifier, L=4)
X'02000004'             Device, Instance 4
X'29'                   SD Context Tag 2 (Time Remaining, L=1)
X'27'                   35
X'3E'                   PD Opening Tag 3 (Timestamp)
        X'A4'                   Application Tag 10 (Date, L=4)
        X'71060301'             June 3, 2013 (Day of Week = Monday)
        X'B4'                   Application Tag 11 (Time, L=4)
        X'0317352F'             03:23:53.47
X'3F'                   PD Closing Tag 3 (Timestamp)
X'4E'                   PD Opening Tag 4 (List of COV Notifications)
        X'0C'                   SD Context Tag 0 (Monitored Object, L=4)
        X'0000000A'             Analog Input, Instance Number=10
        X'1E'                   PD Opening Tag 1 (List of Values)
                X'09'                   SD Context Tag 0 (Property Identifier, L=1)
                X'55'                   85 (PRESENT_VALUE)
                X'2E'                   PD Opening Tag 2 (Value)
                        X'44'                   Application Tag 4 (Real, L=4)
                        X'42820000'             65.0
                X'2E'                   PD Closing Tag 2 (Value)
                X'3C'                   SD Context Tag 3 (Time of Change)
                X'03173400'             03:23:52.00
        X'1F'                   PD Closing Tag 1 (List of Values)
        X'0C'                   SD Context Tag 0 (Monitored Object, L=4)
        X'00400005'             Analog Output, Instance Number=5
        X'1E'                   PD Opening Tag 1 (List of Values)
                X'09'                   SD Context Tag 0 (Property Identifier, L=1)
                X'55'                   85 (PRESENT_VALUE)
                X'2E'                   PD Opening Tag 2 (Value)
                        X'44'                   Application Tag 4 (Real, L=4)
                        X'42A03333'             80.1
                X'2E'                   PD Closing Tag 2 (Value)
        X'1F'                   PD Closing Tag 1 (List of Values)
X'4F'           PD Closing Tag 4 (List of COV Notifications)
```

Assuming the service procedure executes correctly, a simple acknowledgment is returned:

```
X'20'           PDU Type=2 (BACnet-SimpleACK-PDU)
X'0F'           Invoke ID=15
X'1F'           Service ACK Choice=31 (ConfirmedCOVNotificationMultiple)
```

F.1.X3 Encoding for Example E.1.X3 - UnconfirmedCOVNotificationMultiple Service

```
X'00'                   PDU Type=0 (BACnet-Confirmed-Request-PDU, SEG=0, MOR=0, SA=0)
X'02'                   Maximum APDU Size Accepted=206 octets
X'0F'                   Invoke ID=15
X'0B'                   Service Choice=11 (UnconfirmedCOVNotificationMultiple-Request)

X'09'                   SD Context Tag 0 (Subscriber Process Identifier, L=1)
X'12'                   Subscriber Process Identifier=18
X'1C'                   SD Context Tag 1 (Initiating Device Identifier, L=4)
X'02000004'             Device, Instance 4
X'29'                   SD Context Tag 2 (Time Remaining, L=1)
X'1B'                   27
X'4E'           PD Opening Tag 4 (List of COV Notifications)
```

© ASHRAE (www.ashrae.org). For personal use only. Additional reproduction, distribution,
or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

```
X'0C'                   SD Context Tag 0 (Monitored Object, L=4)
X'0000000A'             Analog Input, Instance Number=10
X'1E'                   PD Opening Tag 1 (List of Values)
        X'09'               SD Context Tag 0 (Property Identifier, L=1)
        X'55'               85 (PRESENT_VALUE)
        X'2E'               PD Opening Tag 2 (Value)
            X'44'               Application Tag 4 (Real, L=4)
            X'42820000'         65.0
        X'2F'               PD Closing Tag 2 (Value)
X'1F'                   PD Closing Tag 1 (List of Values)
X'4F'           PD Closing Tag 4 (List of COV Notifications)
```

[Add new **Clauses K.1.X1 and K.1.X2**, p. 882]

K.1.X1 BIBB - Data Sharing-COVM-A (DS-COVM-A)

The A device is a user of COV data from device B that is provided through COV-multiple notifications.

BACnet Service	Initiate	Execute
SubscribeCOVPropertyMultiple	x	
ConfirmedCOVNotificationMultiple		x[1]
UnconfirmedCOVNotificationMultiple		x[1]

[1] Execution of at least one of these services is required.

Support for subscription for time stamped changes is optional. Support for issuing both forms of 'Issue Confirmed Notifications' is not required. Support for execution of ConfirmedCOVNotificationMultiple is not required if 'Issue Confirmed Notifications' in SubscribeCOVPropertyMultiple is never sent as TRUE. Support for execution of UnconfirmedCOVNotificationMultiple is not required if 'Issue Confirmed Notifications' in SubscribeCOVPropertyMultiple is never sent as FALSE.

K.1.X2 BIBB - Data Sharing-COVM-B (DS-COVM-B)

The B device is a provider of COV data of arbitrary properties of specified objects, reported by COV multiple notifications to device A.

BACnet Service	Initiate	Execute
SubscribeCOVPropertyMultiple		x
ConfirmedCOVNotificationMultiple	x	
UnconfirmedCOVNotificationMultiple	x	

Devices claiming conformance to DS-COVM-B shall support a minimum of five COV-multiple contexts, with a minimum of five COV References per such context. Support for timestamped changes is required. Support in SubscribeCOVPropertyMultiple for execution of both 'Issue Confirmed Notifications' as TRUE and 'Issue Confirmed Notifications' as FALSE is required.

© ASHRAE (www.ashrae.org). For personal use only. Additional reproduction, distribution,
or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

135-2012*aq*-3 Add a New Fault Algorithm FAULT_LISTED

Rationale

The elevator objects require reporting of one or multiple fault conditions that may exist simultaneously in a lift or escalator. In addition, the faults indicated may be proprietary fault conditions not defined by the standard.

A new fault algorithm FAULT_LISTED is added, that is monitoring a list of fault indications. The Lift and Escalator object types are defined to use this new fault algorithm. However, this new fault algorithm is not limited to elevator applications. It allows monitoring any list of values of any datatype, either standard or proprietary, that indicate fault conditions. The Event Enrollment object type is extended to allow supporting the fault algorithm.

[Change **Clause 12.12.8, Event Enrollment Object Type**, p. 211]

12.12.8 Object_Property_Reference

...

Depending on the fault algorithm, the values of additional properties of the monitored object are used for particular fault algorithm parameters, as specified by Table 12-15.2.

Table 12-15.2. Additional Monitored Object Properties by Fault Algorithm

Fault Algorithm	Additional Monitored Object Properties	Fault Algorithm Parameters
NONE	none	none
FAULT_CHARACTERSTRING	none	none
FAULT_EXTENDED	Defined by vendor	Defined by vendor
FAULT_LIFE_SAFETY	none	none
FAULT_STATE	none	none
FAULT_STATUS_FLAGS	none	none
FAULT_LISTED	*none*	*none*

[Change **Clause 12.12.23, Event Enrollment Object Type**, p. 214]

12.12.23 Fault_Parameters

This property, of type BACnetFaultParameter, determines the fault algorithm used to monitor the referenced object and provides the parameter values needed for this fault algorithm. The mapping to the fault algorithm parameter values is defined in Table 12-15.3.

If the Event Enrollment object does not apply a fault algorithm, then the fault parameter choice NONE shall be set in this property.

© ASHRAE (www.ashrae.org). For personal use only. Additional reproduction, distribution,
or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

Table 12-15.3. Fault Algorithm, Fault Parameters and Fault Algorithm Parameters

Fault Algorithm	Fault Parameters	Fault Algorithm Parameters
NONE	none	none
FAULT_CHARACTERSTRING	List_Of_Fault_Values	pFaultValues
FAULT_EXTENDED	Vendor_Id Extended_Fault_Type Parameters	pVendorId pFaultType pParameters
FAULT_LIFE_SAFETY	List_Of_Fault_Values Mode_Property_Reference	pFaultValues Referent's value is pMode
FAULT_STATE	List_Of_Fault_Values	pFaultValues
FAULT_STATUS_FLAGS	Status_Flags_Property_Reference	pMonitoredValue
FAULT_LISTED	*Fault_List_Reference*	*Referent's value is pMonitoredList*

[Change **Table 13-8**, p. 504]

Table 13-8. Standardized Fault Algorithms

Fault Algorithm	Clause
NONE	13.4.1
FAULT_CHARACTERSTRING	13.4.2
FAULT_EXTENDED	13.4.3
FAULT_LIFE_SAFETY	13.4.4
FAULT_STATE	13.4.5
FAULT_STATUS_FLAGS	13.4.6
FAULT_LISTED	*13.4.X*

[Add new **Clause 13.4.X**, p. 508]

13.4.X FAULT_LISTED Fault Algorithm

The FAULT_LISTED fault algorithm monitors a list of values. A fault is indicated whenever there are values in the list, or the set of values changes. At the vendor's discretion, evaluation of the algorithm may be delayed by verification or filtering mechanisms local to the object that applies the fault algorithm which are used to insure that the pMonitoredList is correct.

The parameters of this fault algorithm are:

pCurrentReliability	This parameter, of type BACnetReliability, represents the current value of the Reliability property of the object that applies the fault algorithm.
pMonitoredList	This parameter, of type BACnetLIST, is the list monitored by this algorithm.

The conditions evaluated by this fault algorithm are:

(a) If pCurrentReliability is NO_FAULT_DETECTED, and pMonitoredList contains at least one element, then indicate a transition to the FAULTS_LISTED reliability.

(b) If pCurrentReliability is FAULTS_LISTED, and pMonitoredList is empty, i.e., contains no element, then indicate a transition to the NO_FAULT_DETECTED reliability.

(c) If pCurrentReliability is FAULTS_LISTED, and pMonitoredList contains a set of values that is different from the set of values that caused the last transition to FAULTS_LISTED, then indicate a transition to the FAULTS_LISTED reliability.

© ASHRAE (www.ashrae.org). For personal use only. Additional reproduction, distribution,
or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

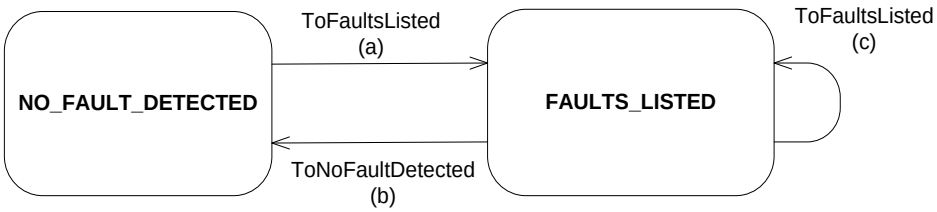

Figure 13-X. Transitions indicated by the FAULT_LISTED algorithm

[Change **Clause 21**, **BACnetFaultParameter** production, p. 683]

BACnetFaultParameter ::= CHOICE {
    none [0] NULL,

    …

    fault-status-flags [5] SEQUENCE {
        status-flags-reference [0] BACnetDeviceObjectPropertyReference
        ~~}~~ *},*
    fault-listed *[7] SEQUENCE {*
        fault-list-reference *[0] BACnetDeviceObjectPropertyReference*
        }
    }

[Change **Clause 21**, **BACnetFaultType** production, p. 684]

BACnetFaultType ::= ENUMERATED {
    none (0),
    fault-characterstring (1),
    fault-extended (2),
    fault-life-safety (3),
    fault-state (4),
    fault-status-flags ~~(5)~~ *(5),*
    fault-listed *(7)*
    }

© ASHRAE (www.ashrae.org). For personal use only. Additional reproduction, distribution,
or transmission in either print or digital form is not permitted without ASHRAE's prior written permission.

[Add a new entry to **History of Revisions**, p. 1027]

(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

…	…	**…**
1	18	**Addendum aq to ANSI/ASHRAE 135-2012** Approved by ASHRAE on February 29, 2016, and by the American National Standards Institute March 1, 2016. 1. Add Elevator Object Types 2. Add COV Property Multiple Services 3. Add a New Fault Algorithm FAULT_LISTED

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.

3/16