

BSR/ASHRAE Addendum *d*  
to ANSI/ASHRAE Standard 135.1-2007

# Public Review Draft

ASHRAE® Standard

## Proposed Addendum *d* to Standard 135.1-2007, *Method of Test for Conformance to BACnet®*

First Public Review (March 2009)  
(Draft Shows Proposed Changes to  
Current Standard)

This draft has been recommended for public review by the responsible project committee. To submit a comment on this proposed addendum, go to the ASHRAE website at <http://www.ashrae.org/technology/page/331> and access the online comment database. The draft is subject to modification until it is approved for publication by the Board of Directors and ANSI. Until this time, the current edition of the standard (as modified by any published addenda on the ASHRAE web site) remains in effect. The current edition of any standard may be purchased from the ASHRAE Bookstore @ <http://www.ashrae.org> or by calling 404-636-8400 or 1-800-527-4723 (for orders in the U.S. or Canada).

This standard is under continuous maintenance. To propose a change to the current standard, use the change submittal form available on the ASHRAE web site @ <http://www.ashrae.org>.

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

© March 20, 2009. This draft is covered under ASHRAE copyright. Permission to reproduce or redistribute all or any part of this document must be obtained from the ASHRAE Manager of Standards, 1791 Tullie Circle, NE, Atlanta, GA 30329-2305. Phone: 404-636-8400, Ext1125. Fax: 404-321-5478. E-mail: [standards.section@ashrae.org](mailto:standards.section@ashrae.org).

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



**[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.

135.1-2007*d*-1. Add test to verify that COV subscription lifetimes are not affected by time-sync requests, p.1.

135.1-2007*d*-2. Add new Active\_COV\_Subscription tests, p. 3.

In the following document, language to be added to existing clauses of ANSI/ASHRAE 135.1-2007 and Addenda is indicated through the use of *italics*, while deletions are indicated by ~~striketrough~~. Where entirely new subclauses are proposed to be added, plain type is used throughout. Only this new and deleted text is open to comment as this time. All other material in this addendum is provided for context only and is not open for public review comment except as it relates to the proposed changes.

**135.1-2007d-1. Add test to verify that COV subscription lifetimes are not affected by time-sync requests.**

**Rationale**

A test is required to verify that a COV subscription's lifetime is not affected by changing the time in a device via TimeSynchronization-Request or UTCTimeSynchronization-Request.

**Addendum 135.1-2007d-1**

[Add new Clause **9.10.X**, p. 222.]

**9.10.X Ensuring Subscription Lifetimes Are Not Affected By Time Changes**

Purpose: To verify that the IUT correctly responds to a SubscribeCOV request to establish a subscription with a temporary lifetime and that the lifetime is not affected by TimeSynchronization or UTC-TimeSynchronization service requests.

Test Concept: The TD subscribes to an object that supports COV reporting in the IUT. The time is changed to  $D_1$ , which is more than 300 seconds in the future. The object is made to change such that a COV notification would be generated and it is verified that a COV notification is issued. The subscription is then allowed to expire and the object is changed again such that a COV notification would be generated, if a subscription were present. It is verified that no notification is generated.

Test Configuration: The IUT contains an object that supports COV reporting. If the IUT does not support TimeSynchronization or UTC-TimeSynchronization, then this test shall be omitted.

1. TRANSMIT SubscribeCOV-Request,
  - 'Subscriber Process Identifier' = (any valid process identifier),
  - 'Monitored Object Identifier' = (any object supporting COV notifications),
  - 'Issue Confirmed Notifications' = TRUE | FALSE,
  - 'Lifetime' = (a value between 60 seconds and 300 seconds)
2. RECEIVE BACnet-SimpleACK-PDU
3. BEFORE **Notification Fail Time**
  - IF (the subscription was for confirmed notifications) THEN
    - RECEIVE ConfirmedCOVNotification-Request,
      - 'Subscriber Process Identifier' = (the same identifier used in the subscription),
      - 'Initiating Device Identifier' = IUT,
      - 'Monitored Object Identifier' = (the same object used in the subscription),
      - 'Time Remaining' = (the requested subscription lifetime),
      - 'List of Values' = (values appropriate to the object type of the monitored object)
  - ELSE
    - RECEIVE UnconfirmedCOVNotification-Request,
      - 'Subscriber Process Identifier' = (the same identifier used in the subscription),
      - 'Initiating Device Identifier' = IUT,
      - 'Monitored Object Identifier' = (the same object used in the subscription),
      - 'Time Remaining' = (the requested subscription lifetime),
      - 'List of Values' = (values appropriate to the object type of the monitored object)
4. MAKE (a change to the monitored object that should cause a COV notification)
5. BEFORE **NotificationFailTime**
  - IF (the subscription was for confirmed notifications) THEN
    - RECEIVE ConfirmedCOVNotification-Request,
      - 'Subscriber Process Identifier' = (the same identifier used in the subscription),
      - 'Initiating Device Identifier' = IUT,
      - 'Monitored Object Identifier' = (the same object used in the subscription),
      - 'Time Remaining' = (a value greater than 0 and less than the requested subscription lifetime),

- 'List of Values' = (values appropriate to the object type of the monitored object)
- ELSE
  - RECEIVE UnconfirmedCOVNotification-Request,
    - 'Subscriber Process Identifier' = (the same identifier used in the subscription),
    - 'Initiating Device Identifier' = IUT,
    - 'Monitored Object Identifier' = (the same object used in the subscription),
    - 'Time Remaining' = (a value greater than 0 and less than the requested subscription lifetime),
  - 'List of Values' = (values appropriate to the object type of the monitored object including the changed value of that triggered the notification)
- 6. TRANSMIT
  - DA = GLOBAL BROADCAST,
  - SA = TD,
  - BACnet-Unconfirmed-Request-PDU,
    - 'Service Choice' = TimeSynchronization-Request,
    - date = D<sub>1</sub>,
    - time = D<sub>1</sub>
- 7. TRANSMIT
  - DA = GLOBAL BROADCAST,
  - SA = TD,
  - BACnet-Unconfirmed-Request-PDU,
    - 'Service Choice' = UTC-TimeSynchronization-Request,
    - date = D<sub>1</sub>,
    - time = D<sub>1</sub>
- 8. MAKE (a change to the monitored object that should cause a COV notification)
- 9. BEFORE **NotificationFailTime**
  - IF (the subscription was for confirmed notifications) THEN
    - RECEIVE ConfirmedCOVNotification-Request,
      - 'Subscriber Process Identifier' = (the same identifier used in the subscription),
      - 'Initiating Device Identifier' = IUT,
      - 'Monitored Object Identifier' = (the same object used in the subscription),
      - 'Time Remaining' = (a value greater than 0 and less than the requested subscription lifetime),
    - 'List of Values' = (values appropriate to the object type of the monitored object)
  - ELSE
    - RECEIVE UnconfirmedCOVNotification-Request,
      - 'Subscriber Process Identifier' = (the same identifier used in the subscription),
      - 'Initiating Device Identifier' = IUT,
      - 'Monitored Object Identifier' = (the same object used in the subscription),
      - 'Time Remaining' = (a value greater than 0 and less than the requested subscription lifetime),
    - 'List of Values' = (values appropriate to the object type of the monitored object including the changed value of that triggered the notification)
- 10. WAIT (the remaining lifetime of the subscription)
- 11. MAKE (a change to the monitored object that would cause a COV notification if there were an active subscription)
- 12. CHECK (verify that the IUT did not transmit a COV notification message)

### 135.1-2007d-2. Add new Active\_COV\_Subscription tests.

#### Rationale

There are currently no tests in Standard 135.1-2007 to test the functionality of the Active\_COV\_Subscriptions property, but tests for this functionality are needed.

#### Addendum 135.1-2007d-2

[Change Clause 7.3.2.10, p. 66]

##### 7.3.2.10 Device Object ~~Test~~Tests

All *Most* necessary tests for functionality of the Device object are covered by tests for the application service or special functionality to which they correspond.

[Add new Clause 7.3.2.10.X, p. 66]

##### 7.3.2.10.X Active\_COV\_Subscriptions SubscribeCOV Test

Purpose: This test case verifies that the IUT correctly updates the Active\_COV\_Subscriptions property when COV subscriptions are created, cancelled and timed-out using SubscribeCOV.

Configuration Requirements: In this test, the tester shall choose three standard objects, O<sub>1</sub>, O<sub>2</sub>, and O<sub>3</sub>, for which the device supports COV. O<sub>1</sub>, O<sub>2</sub>, and O<sub>3</sub> are not required to refer to different objects. The tester shall also choose three non-zero unique process identifiers, P<sub>1</sub>, P<sub>2</sub>, and P<sub>3</sub>, and three non-zero lifetimes L<sub>1</sub>, L<sub>2</sub> and L<sub>3</sub>. Lifetime L<sub>1</sub> shall be long enough to allow the initial part of the test to run through to step 14. Lifetimes L<sub>2</sub> and L<sub>3</sub> shall be long enough for the whole test to be completed without expiring.

The IUT shall start the test with no entries in its Active\_COV\_Subscriptions property.

Test Steps:

1. TRANSMIT SubscribeCOV-Request,  
    'Subscriber Process Identifier' = P<sub>1</sub>,  
    'Monitored Object Identifier' = O<sub>1</sub>,  
    'Issue Confirmed Notifications' = TRUE,  
    'Lifetime' = L<sub>1</sub>
2. RECEIVE BACnet-SimpleACK-PDU
3. BEFORE **NotificationFailTime**  
    RECEIVE ConfirmedCOVNotification-Request,  
    'Subscriber Process Identifier' = P<sub>1</sub>,  
    'Initiating Device Identifier' = IUT,  
    'Monitored Object Identifier' = O<sub>1</sub>,  
    'Time Remaining' = (a value approximately equal to L<sub>1</sub>),  
    'List of Values' = (values appropriate to the object type of the monitored object)
4. TRANSMIT Simple-Ack
5. VERIFY Active\_COV\_Subscriptions = { { {TD, P<sub>1</sub>}, {O<sub>1</sub>, Present\_Value }, TRUE, (a value less than L<sub>1</sub>), (a valid Increment if the property is REAL) } }
6. TRANSMIT SubscribeCOV-Request,  
    'Subscriber Process Identifier' = P<sub>2</sub>,  
    'Monitored Object Identifier' = O<sub>2</sub>,  
    'Issue Confirmed Notifications' = FALSE,  
    'Lifetime' = L<sub>2</sub>
7. RECEIVE BACnet-SimpleACK-PDU
8. BEFORE **NotificationFailTime**  
    RECEIVE UnconfirmedCOVNotification-Request,

- 'Subscriber Process Identifier' = P<sub>2</sub>,
- 'Initiating Device Identifier' = IUT,
- 'Monitored Object Identifier' = O<sub>2</sub>,
- 'Time Remaining' = (a value approximately equal to L<sub>2</sub>),
- 'List of Values' = (values appropriate to the object type of the monitored object)
- 9. VERIFY Active\_COV\_Subscriptions = { { {TD, P<sub>1</sub>}, {O<sub>1</sub>, Present\_Value}, TRUE, (a value less than L<sub>1</sub>),  
(a valid Increment if the property is REAL) },  
{ {TD, P<sub>2</sub>}, {O<sub>2</sub>, Present\_Value}, FALSE, (a value less than L<sub>2</sub>),  
(a valid Increment if the property is REAL) } }
- 10. TRANSMIT SubscribeCOV-Request,
  - 'Subscriber Process Identifier' = P<sub>3</sub>,
  - 'Monitored Object Identifier' = O<sub>3</sub>,
  - 'Issue Confirmed Notifications' = FALSE,
  - 'Lifetime' = L<sub>3</sub>
- 11. RECEIVE BACnet-SimpleACK-PDU
- 12. BEFORE **NotificationFailTime**  
 RECEIVE UnconfirmedCOVNotification-Request,
  - 'Subscriber Process Identifier' = P<sub>3</sub>,
  - 'Initiating Device Identifier' = IUT,
  - 'Monitored Object Identifier' = O<sub>3</sub>,
  - 'Time Remaining' = (a value approximately equal to L<sub>3</sub>),
  - 'List of Values' = (values appropriate to the object type of the monitored object)
- 13. VERIFY Active\_COV\_Subscriptions = { { {TD, P<sub>1</sub>}, {O<sub>1</sub>, Present\_Value}, TRUE, (a value less than L<sub>1</sub>),  
(a valid Increment if the property is REAL) },  
{ {TD, P<sub>2</sub>}, {O<sub>2</sub>, Present\_Value}, FALSE, (a value less than L<sub>2</sub>),  
(a valid Increment if the property is REAL) },  
{ {TD, P<sub>3</sub>}, {O<sub>3</sub>, Present\_Value}, FALSE, (a value less than L<sub>3</sub>),  
(a valid Increment if the property is REAL) } }
- 14. WAIT L<sub>1</sub> + the IUT's timer granularity
- 15. VERIFY Active\_COV\_Subscriptions = { { {TD, P<sub>2</sub>}, {O<sub>2</sub>, Present\_Value}, FALSE, (a value less than L<sub>2</sub>),  
(a valid Increment if the property is REAL) },  
{ {TD, P<sub>3</sub>}, {O<sub>3</sub>, Present\_Value}, FALSE, (a value less than L<sub>3</sub>),  
(a valid Increment if the property is REAL) } }
- 16. TRANSMIT SubscribeCOV-Request,
  - 'Subscriber Process Identifier' = P<sub>3</sub>,
  - 'Monitored Object Identifier' = O<sub>3</sub>
- 17. VERIFY Active\_COV\_Subscriptions = { { {TD, P<sub>2</sub>}, {O<sub>2</sub>, Present\_Value}, FALSE, (a value less than L<sub>2</sub>),  
(a valid Increment if the property is REAL) } }
- 18. TRANSMIT SubscribeCOV-Request,
  - 'Subscriber Process Identifier' = P<sub>2</sub>,
  - 'Monitored Object Identifier' = O<sub>2</sub>
- 19. VERIFY Active\_COV\_Subscriptions = { }

[Add new Clause **7.3.2.10.Y**, p. 66]

### **7.3.2.10.Y Active\_COV\_Subscriptions SubscribeCOVProperty Test**

Purpose: This test case verifies that the IUT correctly updates the Active\_COV\_Subscriptions property when COV subscriptions are created, cancelled and timed-out using SubscribeCOV.

Configuration Requirements: In this test, the tester shall choose three objects and properties, O<sub>1</sub>, O<sub>2</sub>, and O<sub>3</sub> for which the device supports SubscribeCOVProperty. O<sub>1</sub>, O<sub>2</sub>, and O<sub>3</sub> need not refer to different objects and properties. The tester shall also choose three non-zero unique process identifiers, P<sub>1</sub>, P<sub>2</sub>, and P<sub>3</sub>, and three non-zero lifetimes L<sub>1</sub>, L<sub>2</sub> and L<sub>3</sub>. Lifetime L<sub>1</sub> shall be long enough to allow the initial part of the test to run through to step 14. Lifetimes L<sub>2</sub> and L<sub>3</sub> shall be long enough for the whole test to complete without expiring.

Test Steps: The test steps for this test case are identical to the test steps in 7.3.2.10.X except that `SubscribeCOVProperty` is used instead of `SubscribeCOV` and the `Active_COV_Subscriptions` entries shall reflect the property actually subscribed to (and not `Present_Value` if the subscribed property is not `Present_Value`).