



**BACnet[®] TESTING LABORATORIES
ADDENDA**

**Addendum I to
BTL Test Package 15.2**

**Revision 3
Revised 07/8/2019**

Approved by the BTL Working Group on May 23, 2019;
Approved by the BTL Working Group Voting Members on July 18, 2019;
Published on July 23, 2019.

[This foreword and the “Overview” on the following pages are not part of this Test Package. They are merely informative and do not contain requirements necessary for conformance to the Test Package.]

FOREWORD

The purpose of this addendum is to present current changes being made to the BTL Test Package. These modifications are the result of change proposals made pursuant to the continuous maintenance procedures and of deliberations within the BTL-WG Committee. The changes are summarized below.

BTL-15.21-1: Add Notification Forwarder Object Testing [BTLWG-289]2
BTL-15.21-2: Add Load Control Object [BTLWG-380] 12
BTL-15.21-3: Non-Pattern Testing to Notification Forwarder Object [BTLWG-538] 16
BTL-15.21-4: Acknowledge Alarm changes for Notification Forwarder Object [BTLWG-463] 17

In the following document, language to be added to existing clauses within the BTL Test Package 15.2 is indicated through the use of *italics*, while deletions are indicated by ~~strike through~~. Where entirely new subclauses are proposed to be added, plain type is used throughout

In addition, changes to BTL Specified Tests also contain a **yellow** highlight to indicate the changes made by this addendum.

When this addendum is applied, all highlighting will be removed. Change markings on tests will remain to indicate the difference between the new test and an existing 135.1 test. If a test being modified has never existed in 135.1, the applied result should not contain any change markings. When this is the case, square brackets will be used to describe the changes required for this test.

Each addendum can stand independently unless specifically noted via dependency within the addendum. If multiple addenda change the same test or section, each future released addendum that changes the same test or section will note in square brackets whether or not those changes are reflected.

BTL-15.21-1: Add Notification Forwarder Object Testing [BTLWG-289]

Overview:

Add testing for new Notification Forwarder Object introduced in in 135-2010af.

Changes:

[In Checklist, add new entries to Notification Forwarder Object]

3 Objects

Support	Listing	Option
Notification Forwarder Object		
	R ⁺	Base Requirements
	R	Supports DM-LM-B
	BTL-C ¹	Supports AE-NF-B
	BTL-C ¹	Supports AE-NF-I-B
	C ²	Supports Port Filter Property
	O	Supports writable Out Of Service Properties
	O	Supports Reliability Evaluation Inhibit Property
⁺ Contact BTL for interim tests for this object. ¹ You must support one of these BIBBs if your device contains a Notification Forwarder object. ² A Port_Filter property is required if the IUT is a router.		

[In Checklist, update the following BIBBs in the Alarm and Event Management section.]

5 Alarm and Event Management BIBBs

Support	Listing	Option
...		
Alarm and Event - Notification Forwarder - B		
	R ⁺	Base Requirements
	O	Supports forwarding of events received from the local device
⁺ Contact BTL for interim tests for this BIBB		
Alarm and Event - Notification Forwarder - Internal - B		
	R ⁺	Base Requirements
	O	Supports forwarding of events received from an external device
	C ¹	Supports unconfigurable Process Identifier Filter
	C ¹	Supports configurable Process Identifier Filter
⁺ Contact BTL for interim tests for this BIBB ¹ One of these options must be supported		
Alarm and Event - Configurable Recipient Lists - B		
	R ⁺	Base Requirements
	R	Supports DS-WP-B
	R	Supports DM-DDB-A
	C ¹	Supports writable Recipient List property in Notification Class Objects
	C ¹	Supports writable Recipient List property in Notification Forwarder Object

Support	Listing	Option
	<i>O</i>	<i>Supports DM-LM-B for Recipient List property</i>
[†] Contact BTL for interim tests for this BIBB ¹ Must contain at least one instance of one of these objects to claim this BIBB		
Alarm and Event - Temporary Event Subscriptions - A		
	<i>R⁺</i>	Base Requirements
	<i>R</i>	<i>Supports DM-LM-A for Subscribed Recipients property</i>
[†] Contact BTL for interim tests for this BIBB		

[In BTL Test Plan, update the existing Notification Forwarder Section.]

3.51 Notification Forwarder Object

3.51.1 Base Requirements

Contact BTL for interim tests for this object.

Base requirements must be met by any IUT that can contain Notification Forwarder objects.

Verify Checklist		
	Test Conditionality	<i>Must be executed.</i>
	Test Directives	<i>Verify the IUT claims Protocol Revision 13 or higher.</i>
	Testing Hints	
BTL - 7.2.X5 - Time Non-Pattern Properties Test		
	Test Conditionality	<i>Must be executed.</i>
	Test Directives	<i>Apply to the fromTime and again to the toTime element in a Recipient_List property and to the Subscribed_Recipients property in a Notification Forwarder object.</i>
	Testing Hints	
135.1-2013q - 7.3.2.30.8.1 - Time Count Down Test		
	Test Conditionality	<i>Must be executed.</i>
	Test Directives	
	Testing Hints	
135.1-2013q - 7.3.2.30.8.2 - Expiration Test		
	Test Conditionality	<i>Must be executed.</i>
	Test Directives	
	Testing Hints	
135.1-2013q - 7.3.2.30.8.3 - Time Renewal Test		
	Test Conditionality	<i>Must be executed.</i>
	Test Directives	
	Testing Hints	
135.1-2013q - 7.3.2.30.8.4 - Resubscription Update Test		
	Test Conditionality	<i>Must be executed.</i>
	Test Directives	
	Testing Hints	
135.1-2013q - 7.3.2.30.8.5 - Delete Test		
	Test Conditionality	<i>Must be executed.</i>
	Test Directives	<i>Perform this using base Setup 1.</i>
	Testing Hints	
135.1-2013q - 7.3.2.30.8.6 - Subscription Of Similar Entries Test		
	Test Conditionality	<i>Must be executed.</i>
	Test Directives	<i>Perform this using base Setup 1.</i>
	Testing Hints	
135.1-2013q - 7.3.2.30.13.1 - Recipient_List Persistence Test		
	Test Conditionality	<i>Must be executed.</i>

	<i>Test Directives</i>	
	<i>Testing Hints</i>	
135.1-2013q - 7.3.2.21.3.7 - Recipient List non-volatility test		
	<i>Test Conditionality</i>	<i>Must be executed.</i>
	<i>Test Directives</i>	
	<i>Testing Hints</i>	
135.1-2013q - 7.3.2.30.13.2 - Subscribed Recipients Persistence Test		
	<i>Test Conditionality</i>	<i>Must be executed.</i>
	<i>Test Directives</i>	
	<i>Testing Hints</i>	
135.1-2013q - 7.3.2.30.14.1 - Time Remaining Range Test		
	<i>Test Conditionality</i>	<i>Must be executed.</i>
	<i>Test Directives</i>	
	<i>Testing Hints</i>	
135.1-2013q - 7.3.2.30.14.2 - Recipient Capacity Test		
	<i>Test Conditionality</i>	<i>Must be executed.</i>
	<i>Test Directives</i>	
	<i>Testing Hints</i>	
135.1-2013q - 7.3.2.X.12.3.2 - Recipient List precludes Global Broadcast Restriction Test		
	<i>Test Conditionality</i>	<i>Must be executed.</i>
	<i>Test Directives</i>	
	<i>Testing Hints</i>	
135.1-2013q - 7.3.2.30.12.3.4 - Subscribed Recipients preclude Global Broadcast Restriction Test		
	<i>Test Conditionality</i>	<i>Must be executed.</i>
	<i>Test Directives</i>	
	<i>Testing Hints</i>	

[In BTL Test Plan, add these new sections to the Object, Notification Forwarder Object section]

3.51.2 Supports DM-LM-B

The IUT must support DM-LM-B if it supports Notification Forwarder objects.

Verify Checklist		
	Test Conditionality	Must be executed.
	Test Directives	Verify the IUT claims support for the BIBB DM-LM-B.
	Testing Hints	

3.51.3 Supports AE-NF-B

The IUT must support either AE-NF-B or AE-NF-I-B if it claims support for Notification Forwarder objects.

Verify Checklist		
	Test Conditionality	Must be executed.
	Test Directives	Verify the IUT claims support for the BIBB AE-NF-B.
	Testing Hints	

3.51.4 Supports AE-NF-I-B

The IUT must support either AE-NF-B or AE-NF-I-B if it claims support for Notification Forwarder objects.

Verify Checklist		
	Test Conditionality	Must be executed.
	Test Directives	Verify the IUT claims support for the BIBB AE-NF-I-B.
	Testing Hints	

3.51.5 Supports Port_Filter Property

The IUT contains the Port_Filter property. This is a required capability for Notification Forwarder objects if the IUT is a router.

135.1-2013q - 7.3.2.30.10 - Port_Filter Test		
---	--	--

	Test Conditionality	Must be executed.
	Test Directives	Test each port value, if the IUT is a router.
	Testing Hints	
Verify EPICS		
	Test Conditionality	Must be executed.
	Test Directives	Every Notification Forwarder object shall contain a Port_Filter property if the IUT is a router. This property shall be writable if present, though neither the size of the array nor the Port_ID portion of the BACnetPortPermission entries shall be modifiable via writes to this property. The number of entries in the array shall match the number of BACnet ports currently defined in the device.
	Testing Hints	

3.51.6 Supports Writable Out_Of_Service Properties

The IUT contains, or can be made to contain, an Out_Of_Service property that is writable.

BTL - 7.3.2.30.6 - Out Of Service Property Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

3.51.7 Supports Reliability_Evaluation_Inhibit Property

The IUT contains, or can be made to contain, a Reliability_Evaluation_Inhibit property that is configurable to the value of TRUE.

135.1-2013q - 7.3.2.30.X8.1 - Reliability Evaluation Inhibit Test		
	Test Conditionality	Must be executed.
	Test Directives	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	Testing Hints	

[In BTL Test Plan, update the AE-NF-B BIBB section]

5.23 Alarm and Event Management - Notification Forwarder - B

5.23.1 Base Requirements

Contact BTL for interim tests for this BIBB.

Base requirements must be met by any IUT claiming conformance to this BIBB.

135.1-2013q - 7.3.2.30.2 - Recipient List Forwarding Test		
	Test Conditionality	Must be executed with Local Forwarding Only = FALSE.
	Test Directives	
	Testing Hints	
135.1-2013q - 7.3.2.30.3 - Subscribed Recipients Forwarding Test		
	Test Conditionality	Must be executed with Local Forwarding Only = FALSE.
	Test Directives	Perform this using base Setup 2.
	Testing Hints	
135.1-2013q - 7.3.2.30.7.1 - Destination Date Filtering Test		
	Test Conditionality	Must be executed.
	Test Directives	Perform this using base Setup 1.
	Testing Hints	
135.1-2013q - 7.3.2.30.7.2 - Destination Time Filtering Test		
	Test Conditionality	Must be executed.
	Test Directives	Perform this using base Setup 1.
	Testing Hints	
135.1-2013q - 7.3.2.30.7.3 - Process Identifier Test		
	Test Conditionality	Must be executed.

	Test Directives	<i>Perform this using base Setup 1.</i>
	Testing Hints	
135.1-2013q - 7.3.2.30.7.4 - Destination Transition Filtering Test		
	Test Conditionality	<i>Must be executed.</i>
	Test Directives	
	Testing Hints	
135.1-2013q - 7.3.2.30.11.2 - Forwards Locally and Remotely When False		
	Test Conditionality	<i>Must be executed.</i>
	Test Directives	
	Testing Hints	
135.1-2013q - 7.3.2.30.5 - Character Encoding Test		
	Test Conditionality	<i>Must be executed.</i>
	Test Directives	
	Testing Hints	
135.1-2013q - 7.3.2.30.12.1 - Local Broadcast To Receiving Port Restriction Test		
	Test Conditionality	<i>Must be executed.</i>
	Test Directives	
	Testing Hints	
135.1-2013q - 7.3.2.30.12.2 - Globally Broadcast Event Notification Received Restriction Test		
	Test Conditionality	<i>Must be executed.</i>
	Test Directives	
	Testing Hints	
135.1-2013q - 7.3.2.30.12.3 - Forwarding As Global Broadcast Restriction Test		
	Test Conditionality	<i>Must be executed.</i>
	Test Directives	<i>Perform this using base Setup 1.</i>
	Testing Hints	
135.1-2013q - 7.3.2.30.12.4 - Directed Broadcast Received Forwarding To BACnetAddress Restriction Test		
	Test Conditionality	<i>Must be executed.</i>
	Test Directives	<i>Perform this using base Setup 1.</i>
	Testing Hints	
135.1-2013q - 7.3.2.30.12.5 - Directed Broadcast Received Forwarding To Object Identifier Restriction Test		
	Test Conditionality	<i>Must be executed.</i>
	Test Directives	<i>Perform this using base Setup 1.</i>
	Testing Hints	
135.1-2013q - 7.3.2.30.12.6 - Port Restriction Test		
	Test Conditionality	<i>Must be executed.</i>
	Test Directives	
	Testing Hints	

[In BTL Test Plan, add these new sections to AE-NF-B]

5.23.2 Supports Forwarding of Events Received From the Local Device

The IUT contains or can be made to contain the Local_Forwarding_Only property set to TRUE.

Verify Checklist		
	Test Conditionality	<i>Must be executed.</i>
	Test Directives	Verify that the IUT claims support for the BIBB AE-NF-I-B
	Testing Hints	

[In BTL Test Plan, update existing section in AE-NF-I-B]

5.24 Alarm and Event - Notification Forwarder - Internal - B

5.24.1 Base Requirements

Contact BTL for interim tests for this BIBB.

Base requirements must be met by any IUT claiming conformance to this BIBB.

135.1-2013q - 7.3.2.30.2 - Recipient List Forwarding Test		
	Test Conditionality	Must be executed with Local_Forwarding_Only = TRUE
	Test Directives	REPEAT this test with SRC_CONF_DEV equal to the IUT, and for both settings of DEST_CONF_NOTIF TRUE/FALSE (confirmed/unconfirmed).
	Testing Hints	
135.1-2013q - 7.3.2.30.3 - Subscribed Recipients Forwarding Test		
	Test Conditionality	Must be executed with Local_Forwarding_Only = TRUE.
	Test Directives	Perform this using base Setup 2.
	Testing Hints	REPEAT this test with SRC_CONF_DEV equal to the IUT, and for both settings of DEST_CONF_NOTIF TRUE/FALSE (confirmed/unconfirmed).
135.1-2013q - 7.3.2.30.5 - Character Encoding Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
135.1-2013q - 7.3.2.30.12.1 - Local Broadcast To Receiving Port Restriction Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
135.1-2013q - 7.3.2.30.12.2 - Globally Broadcast Event Notification Received Restriction Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
135.1-2013q - 7.3.2.30.12.3 - Forwarding As Global Broadcast Restriction Test		
	Test Conditionality	Must be executed.
	Test Directives	Perform this using base Setup 1.
	Testing Hints	
135.1-2013q - 7.3.2.30.12.4 - Directed Broadcast Received Forwarding To BACnetAddress Restriction Test		
	Test Conditionality	Must be executed.
	Test Directives	Perform this using base Setup 1.
	Testing Hints	
135.1-2013q - 7.3.2.30.12.5. - Directed Broadcast Received Forwarding To Object Identifier Restriction Test		
	Test Conditionality	Must be executed.
	Test Directives	Perform this using base Setup 1.
	Testing Hints	
135.1-2013q - 7.3.2.30.12.6 - Port Restriction Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
Verify Checklist		
	Test Conditionality	Must be executed.
	Test Directives	Verify that the IUT claims initiate support for both Unconfirmed and Confirmed Event Notification services.
	Testing Hints	
135.1-2013q - 7.3.2.30.11.1 Only Forwards Locally When True		
	Test Conditionality	Must be executed.
	Test Directives	Perform this using base Setup 1.
	Testing Hints	

[In BTL Test Plan, add these new sections to AE-NF-I-B]

5.24.2 Supports Forwarding of Events Received From an External Device

The IUT contains, or can be made to contain, a Local_Forwarding_Only property which can be set to FALSE.

Verify Checklist		
	Test Conditionality	Must be executed.

Test Directives	Verify that the IUT claims support for AE-NF-B
Testing Hints	

5.24.3 Supports Unconfigurable Process_Identifier_Filter Property

The IUT contains a read-only version of the required property Process_Identifier_Filter.

135.1-2013q - 7.3.2.30.9.5 - Fixed Process_Identifier_Filter Test	
Test Conditionality	Must be executed.
Test Directives	
Testing Hints	

5.24.4 Supports Configurable Process_Identifier_Filter Property

The one or more Notification Forwarder objects in the IUT contain a configurable Process_Identifier_Filter property

135.1-2013q - 7.3.2.30.9.1 - NULL and Unsigned32 Choice Test	
Test Conditionality	Must be executed.
Test Directives	
Testing Hints	
135.1-2013q - 7.3.2.30.9.3 - Zero Unfiltered Process Identifier Test	
Test Conditionality	Must be executed.
Test Directives	Perform this using base Setup 2.
Testing Hints	

[In BTL Test Plan, update existing section AE-CRL-B]

5.25 Alarm and Event Management - Configurable Recipient Lists - B

5.25.1 Base Requirements

~~Contact BTL for interim tests for this BIBB.~~

There are no base requirement tests for this section. Existing tests in the Notification Class ensure Recipient_List in Notification Class objects support writing all forms.

[In BTL Test Plan, add these new sections to AE-CRL-B]

5.25.2 Supports DS-WP-B

The IUT supports the Write Property service for its Recipient_List in Notification Class or Notification Forwarder objects.

Verify Checklist or BTL - 9.22.1.X2 - Writing to Properties Based on Data Type	
Test Conditionality	Must be executed.
Test Directives	Verify the IUT claims support for DS-WP-B option ‘contains writable list properties’ and this test is executed with the Recipient_List property for the supported Notification Class or Notification Forwarder object in the IUT.
Testing Hints	

5.25.3 Supports DM-DDB-A

The IUT supports DM-DDB-A. The IUT must be able to use the DM-DDB-A functionality to locate alarm recipients.

Verify Checklist	
Test Conditionality	Must be executed.
Test Directives	Verify that the IUT claims support for DM-DDB-A in the Checklist.
Testing Hints	

5.25.4 Supports Writable Recipient_List Property in Notification Class Objects

The IUT must support a writable Recipient_List property in its Notification Class objects to claim this BIBB.

Verify Checklist		
	Test Conditionality	Must be executed.
	Test Directives	Verify the ‘Supports Writable Recipient_List Properties’ is checked in the Notification Class Object section.
	Testing Hints	
Verify Checklist		
	Test Conditionality	Must be executed.
	Test Directives	Verify the device supports 1 or more Notification class objects.
	Testing Hints	

5.25.5 Supports Writable Recipient_List Property in Notification Forwarder Objects

The IUT must support a writable Recipient_List property in its Notification Forwarder objects to claim this BIBB.

Verify Checklist		
	Test Conditionality	Must be executed.
	Test Directives	Verify the ‘Supports Writable Recipient_List Properties’ is checked in the Notification Forwarder Object section.
	Testing Hints	
Verify Checklist		
	Test Conditionality	Must be executed.
	Test Directives	Verify device contains 1 or more Notification Forwarder objects.
	Testing Hints	

5.25.6 Supports DM-LM-B for Recipient_List Property

If the IUT supports AddListElement and RemoveListElement services for the Recipient_List property this item must be checked.

Verify Checklist		
	Test Conditionality	Must be executed.
	Test Directives	Verify DM-LM-B support for both Notification Forwarder and Notification Class objects.
	Testing Hints	

[In BTL Test Plan, Update existing section for AE-TEs-A]

5.26 Alarm and Event Management - Temporary Event Subscriptions - A

5.26.1 Base Requirements

Contact BTL for interim tests for this BIBB.

Base requirements must be met by any IUT claiming conformance to this BIBB.

Verify EPICS		
	Test Conditionality	Verify the EPICS claims execute support for ConfirmedEventNotification AND UnconfirmedEventNotification services.
	Test Directives	
	Testing Hints	
Verify EPICS		
	Test Conditionality	Verify the EPICS contains 1 or more Notification Forwarder objects.
	Test Directives	
	Testing Hints	

[In BTL Test Plan, add these new sections to AE-TES-A]

5.26.2 Supports DM-LM-A for Subscribed_Recipients Property

The IUT must support the DM-LM-A services for the Subscribed_Recipients property of the Notification Forwarder object.

Verify Checklist	
Test Conditionality	Must be executed.
Test Directives	Verify DM-LM-A for the Subscribed_Recipients property is supported.
Testing Hints	

[In BTL Specified Tests, modify test that exists in **135.1-2013q** to correct the expected Status_Flags]

7.3.2.30.6 Out_Of_Service Property Test

BACnet Reference Clauses: 12.51.7

Purpose: This test case verifies that event forwarding is not done while Out_Of_Service is TRUE.

Test Concept: Set up both Recipient_List and Subscribed_Recipient recipient entries with no filters specified and then send event notifications to the Notification Forwarder while the value of the Out_Of_Service property is TRUE. Subscribed_Recipients are configured as part of base setup 2 for Notification Forwarder object tests. Verify that forwarding of the event notifications is not performed.

If the Out_Of_Service property of the object under test is not writable, and if the value of the property cannot be changed by other means, then this test shall be omitted.

Configuration Requirements: Base setup 2 for Notification Forwarder object tests with TR lifetime sufficient for this test.

Test Steps:

1. MAKE (Recipient_List = { (all), --Valid Days
 (all), --From Time, To Time
 DEST_OBJ_ID2, --Recipient D2
 DEST_PROCESS_ID, --Process Identifier
 FALSE, --Issue Confirmed Notifications
 {T, T, T} --Transitions
 }) --One list element
2. MAKE (Out_Of_Service = TRUE)
3. VERIFY Out_Of_Service = TRUE
4. VERIFY Status_Flags = (?FALSE, FALSE, ?FALSE, TRUE)
5. TRANSMIT SOURCE = DS, UnconfirmedEventNotification-Request,
 'Process Identifier' = SRC_PROCESS_ID,
 'Initiating Device Identifier' = SRC_NOTIF_DEV,
 'Event Object Identifier' = SRC_NOTIF_OBJ,
 'Time Stamp' = (any valid time stamp),
 'Notification Class' = SRC_NOTIF_CLS,
 'Priority' = (any valid priority),
 'Event Type' = (any valid event type),
 'Message Text' = (optional, any valid message text),
 'Notify Type' = SRC_NOTIF_TYP,
 'AckRequired' = (any valid value), -- absent if 'Notify Type' is ACK_NOTIFICATION
 'From State' = (any valid From_State), -- absent if 'Notify Type' is ACK_NOTIFICATION
 'To State' = (any valid To_State),
 'Event Values' = (any valid event values) -- absent if 'Notify Type' is ACK_NOTIFICATION
6. WAIT Notification Fail Time
7. CHECK (the IUT did not transmit an event notification)
8. MAKE (Out_Of_Service = FALSE)
9. VERIFY Out_Of_Service = FALSE

- 10. VERIFY Status_Flags = (?FALSE, ?, ?FALSE, FALSE)
- 11. TRANSMIT SOURCE = DS, UnconfirmedEventNotification-Request,
 - 'Process Identifier' = SRC_PROCESS_ID,
 - 'Initiating Device Identifier' = SRC_NOTIF_DEV,
 - 'Event Object Identifier' = SRC_NOTIF_OBJ,
 - 'Time Stamp' = (any valid time stamp),
 - 'Notification Class' = SRC_NOTIF_CLS,
 - 'Priority' = (any valid priority),
 - 'Event Type' = (any valid event type),
 - 'Message Text' = (optional, any valid message text),
 - 'Notify Type' = SRC_NOTIF_TYP,
 - 'AckRequired' = (any valid value), -- absent if 'Notify Type' is ACK_NOTIFICATION
 - 'From State' = (any valid From_State), -- absent if 'Notify Type' is ACK_NOTIFICATION
 - 'To State' = (any valid To_State),
 - 'Event Values' = (any valid event values) -- absent if 'Notify Type' is ACK_NOTIFICATION
- 12. BEFORE **Notification Fail Time** --The following can be in any order
 - RECEIVE DESTINATION = D1, UnconfirmedEventNotification-Request
 - RECEIVE DESTINATION = D2, UnconfirmedEventNotification-Request

BTL-15.21-2: Add Load Control Object [BTLWG-380]

Overview:

Addendum 135-2004e added the Load Control object. This document makes needed changes in the BTL Test Package to claim Load Control object.

Changes:

[In BTL Checklist, add Load Control object type to Section 3, Objects]

Support	Listing	Option
Load Control Object		
	R ⁺	Base Requirements
	R	<i>Supports writable Requested Shed Level to LEVEL choice</i>
	O	<i>Supports writable Reliability property</i>
	O	<i>Supports writable Requested Shed Level to PERCENT choice</i>
	O	<i>Supports writable Requested Shed Level to AMOUNT choice</i>
†Contact BTL for interim tests for this object.		

[In BTL Test Plan, update the existing Load Control Object section]

3.43 Load Control Object

3.43.1 Base Requirements

Contact BTL for interim tests for this object.

Base requirements must be met by any IUT that can contain Load Control objects.

BTL - 7.3.2.X53.2 - Shed Levels property test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

[In BTL Test Plan, add the following new sections to 3.43 Load Control Object.]

3.43.2 Supports Writable Requested_Shed_Level to LEVEL Choice

The Requested_Shed_Level property in Load Control objects is writable to LEVEL choice.

BTL - 7.3.2.X53.1 - Requested_Shed_Level property test with LEVEL choice		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

3.43.3 Supports Writable Reliability Property

The Reliability property in Load Control objects is writable.

BTL - 7.3.2.X53.3 - Load Control Status Flags and Reliability Test		
	Test Conditionality	If Reliability is writable, this test must be executed.
	Test Directives	
	Testing Hints	

3.43.4 Supports Writable Requested_Shed_Level to PERCENT Choice

The Requested_Shed_Level property in Load Control objects is writable to PERCENT choice.

BTL - 7.3.2.X53.4 - Requested_Shed_Level property test with PERCENT choice	
Test Conditionality	If no object can be made to meet the configuration requirements, this test shall be skipped.
Test Directives	
Testing Hints	

3.43.5 Supports Writable Requested_Shed_Level to AMOUNT Choice

The Requested_Shed_Level property in Load Control objects is writable to AMOUNT choice.

BTL - 7.3.2.X53.5 - Requested_Shed_Level property test with AMOUNT choice	
Test Conditionality	If no object can be made to meet the configuration requirements, this test shall be skipped.
Test Directives	
Testing Hints	

[In BTL Specified Tests, add Load Control object tests in section 3.X53. Since these are entirely new tests, these are indicated with plain text.]

7.3.2.X53 Load Control Object Tests

The Load Control object defines a standardized object whose properties represent the externally visible characteristics of a mechanism for controlling load requirements. A BACnet device can use a Load Control object to allow external control over the shedding of a load that it controls. The mechanisms by which the loads are shed are not visible to the BACnet client. The Load Control Object utilizes parameter control through its writable Requested_Shed_Level, Start_Time, Shed_Duration, Duty_Window, Enable and Shed_Levels properties.

7.3.2.X53.1 Requested_Shed_Level property test with LEVEL choice

Reason for Change: This test is not specified in any SSPC proposal.

Purpose: To verify the performance of a shed request with LEVEL choice.

Test Concept: The Requested_Shed_Level property of the Load Control object is set to a LEVEL choice and it is verified that the series of required actions which that sets into operation occur correctly.

Configuration Requirements: The IUT shall be configured so that Present_Value is equal to SHED_INACTIVE, preceding the beginning of this test. Writing Start_Time and/or Shed_Duration with values such that current time is after ST+SD forces Present_Value to become equal to SHED_INACTIVE.

Test Steps:

1. VERIFY Requested_Shed_Level = (one of the default Requested_Shed_Level values for a previous shed request, not necessarily the LEVEL default of 0)
2. VERIFY Expected_Shed_Level = (that same default Requested_Shed_Level value)
3. VERIFY Actual_Shed_Level = (that same default Requested_Shed_Level value)
4. VERIFY Present_Value = SHED_INACTIVE
5. VERIFY Shed_Duration = 0
6. VERIFY Start_Time = (the fully unspecified datetime value)
7. VERIFY Duty_Window = (PAV, the pre-agreed upon value)
8. WRITE Enable = TRUE
9. WRITE Shed_Duration = (SD, any value appropriate to the object)
10. WRITE Start_Time = (ST, any value preceding the beginning of this test)
11. WRITE Duty_Window = (DW, any value appropriate to the object)
12. WRITE Requested_Shed_Level = (a value appropriate to the object with a LEVEL choice, that is not equal to the default value: 0)

-- the above four writes can occur in any order, but it is needful that Enable becomes TRUE before the others. Each of these writes is a reconfiguration if the current time is prior to Start_Time. A reconfiguration is what forces the Present_Value to

SHED_REQUEST_PENDING, so that can be observed after the first write and also is observable in-between any of the writes.

13. VERIFY Present_Value = SHED_REQUEST_PENDING
14. WAIT (until the shed request has started, typically at Start_Time but it can start earlier to achieve compliance at Start_Time)
15. VERIFY Present_Value = (SHED_REQUEST_PENDING or SHED_COMPLIANT or SHED_NONCOMPLIANT)
16. IF (current time is before ST, but the shed request has started) THEN
 VERIFY Present_Value = SHED_NONCOMPLIANT
17. IF (current time is at or after ST)
 VERIFY Present_Value = (SHED_COMPLIANT or SHED_NONCOMPLIANT)
18. IF (current time is after ST+DW and Actual_Shed_Level does not comply with Requested_Shed_Value)
 VERIFY Present_Value = SHED_NONCOMPLIANT
19. VERIFY Shed_Duration = SD
20. VERIFY Start_Time = ST
21. VERIFY Duty_Window = DW
22. VERIFY Expected_Shed_Level = (any value appropriate to the choice, that is not equal to the default value)
23. VERIFY Actual_Shed_Level = (any value appropriate to the choice, that is not equal to the default value)
- the above VERIFY statements apply all through the time that there is a pending or active shed event
24. WAIT (until the shed request has completed, at ST+SD)
25. VERIFY Requested_Shed_Level = 0
26. VERIFY Expected_Shed_Level = (0, that same Default LEVEL value)
27. VERIFY Actual_Shed_Level = (0, that same Default LEVEL value)
28. VERIFY Shed_Duration = 0
29. VERIFY Start_Time = (the fully unspecified datetime value)
30. VERIFY Duty_Window = PAV

Notes to Tester: The writing of Duty_Window can be skipped, for the tester to see that the VERIFY Duty_Window = DW during a pending or active shed event, that property takes on PAV, the pre-agreed upon value.

7.3.2.X53.2 Shed_Levels property test

Reason for Change: This test is not specified in any SSPC proposal.

Purpose: To verify writability of Shed_Levels property and verify that when commanded with the LEVEL choice, the Load Control object shall take a shedding action described by the corresponding element in the Shed_Level_Descriptions array.

Test Concept: The Shed_Levels property of the Load Control object being tested is written to BACnetARRAY of unsigned integers representing the shed levels for the LEVEL choice of BACnetShedLevel that have meaning for this particular Load Control object. Verify that is updating correctly. The array shall be ordered by increasing shed amount.

Test Steps:

1. READ N1 = Shed_Levels, ARRAY_INDEX = 0
2. VERIFY (Shed_Level_Descriptions = N1, ARRAY_INDEX = 0)
3. WRITE Shed_Levels = (any content that is different from the current value, but nonetheless still ordered by increasing shed amount)
4. READ N2 = Shed_Levels, ARRAY_INDEX = 0 -- obtaining the length of the new value
5. VERIFY (Shed_Level_Descriptions = N2, ARRAY INDEX = 0)

7.3.2.X53.3 Load Control Status_Flags and Reliability Test

Purpose: To ensure Status_Flags reflects the Reliability property value.

Test Concept: Write to Reliability and verify the interrelationship between the Status_Flags and Reliability.

Configuration Requirements: The selected object is configured such that its Reliability is NO_FAULT_DETECTED before execution of this test. If the Reliability property is not present or not writable, then this test shall be skipped.

Test Steps:

1. VERIFY Reliability = NO_FAULT_DETECTED

```
2. VERIFY Status_Flags = (?, FALSE, ?, FALSE)
3. REPEAT X = (all values of the Reliability enumeration appropriate to the object type except
    NO_FAULT_DETECTED) DO {
    WRITE Reliability = X
    VERIFY Reliability = X
    VERIFY Status_Flags = (TRUE, TRUE, ?, FALSE)
    WRITE Reliability = NO_FAULT_DETECTED
    VERIFY Reliability = NO_FAULT_DETECTED
    VERIFY Status_Flags = (? FALSE, ?, FALSE)
    }
```

7.3.2.X53.4 Requested_Shed_Level property test with PERCENT choice

Reason for Change: This test is not specified in any SSPC proposal.

Purpose: To verify the performance of a shed request with PERCENT choice.

Test Concept: The Requested_Shed_Level property of the Load Control object is set to a PERCENT choice and it is verified that the series of required actions which that sets into operation occur correctly.

Test Steps: The test steps defined in test **7.3.2.X53.1** shall be followed except that the Requested_Shed_Level property of the Load Control object is written to a PERCENT choice, and the default value for a shed request with PERCENT choice in Requested_Shed_Level, Expected_Shed_Level, and Actual_Shed_Level properties is 100

7.3.2.X53.5 Requested_Shed_Level property test with AMOUNT choice

Reason for Change: This test is not specified in any SSPC proposal.

Purpose: To verify the performance of a shed request with AMOUNT choice.

Test Concept: The Requested_Shed_Level property of the Load Control object is set to an AMOUNT choice and it is verified that the series of required actions which that sets into operation occur correctly.

Test Steps: The test steps defined in test **7.3.2.X53.1** shall be followed except that the Requested_Shed_Level property of the Load Control object is written to an AMOUNT choice, and the default value for a shed request with AMOUNT choice in Requested_Shed_Level, Expected_Shed_Level, and Actual_Shed_Level properties is 0.0

BTL-15.21-3: Non-Pattern Testing to Notification Forwarder Object [BTLWG-538]

Overview:

Add Non-Pattern WPM testing to the Notification Forwarder Object.

Changes:

[In BTL Test Plan, add reference to new test to the NF Base Requirements]

3.51 Notification Forwarder Object

3.51.1 Base Requirements

Base requirements must be met by any IUT that can contain Notification Forwarder objects.

...	
BTL - 9.23.2.X10 Time Non-Pattern Properties Test using WritePropertyMultiple service	
Test Conditionality	This test shall only be applied to devices claiming Protocol_Revision 11 or higher and which supports execution of WritePropertyMultiple.
Test Directives	Apply to the fromTime and again to the toTime element in a Recipient_List property in a Notification Forwarder Object.
Testing Hints	

BTL-15.21-4: Acknowledge Alarm changes for Notification Forwarder Object [BTLWG-463]

Overview:

When acknowledging an event notification that has been forwarded by a Notification Forwarder; the acknowledging device shall send the AcknowledgeAlarm service request directly to the device indicated by the 'Initiating Device Identifier' parameter of the event notification.

Changes:

[In Test Plan, add reference to this new test as indicated in existing section of Alarm and Event Management - Acknowledge - A Base Requirements]

5.4.1 Base Requirements

...		
BTL - 8.1.X2 - Successful Alarm Acknowledgment of Confirmed Event Notifications Using the 'Initiating Device Identifier' Parameter		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

[In BTL Specified Tests, add a new test in section 8.1 after the sole existing test in that section]

8.1.X2 Successful Alarm Acknowledgment of Confirmed Event Notifications Using the 'Initiating Device Identifier' Parameter

Reason for Change: Added a new test ensuring correct behaviour acknowledging an event notification that has been forwarded by a Notification Forwarder.

Purpose: To verify that the IUT is correctly implemented to send AcknowledgeAlarm directly to the device indicated by the 'Initiating Device Identifier' parameter of the event notification, in alarms and events that are reported to the IUT via the ConfirmedEventNotification and UnconfirmedEventNotification services. Requests having the 'Initiating Device Identifier' parameter equal-to and different from SOURCE, are both tested.

Test Concept: Two times. a purposefully constructed event notification is sent, and it is observed that IUT is acknowledging directly to the device indicated by the 'Initiating Device Identifier' parameter of the event notification.

Configuration: For this test, the tester shall choose an object O1, and tester places O1 into an alarm state such that the transition requires an acknowledgment. Then purposefully the EventNotification packet which is sent is crafted to represent that a Notification Forwarder was involved, so though SOURCE is from the TD, the O1 resides in a different device TD1. Then the steps are repeated but with 'Initiating Device Identifier' representing that O1 is in TD and thus the same device. Each time it is observed that the AcknowledgeAlarm is sent to the device represented as the 'Initiating Device Identifier'.

Test Steps:

- TRANSMIT ConfirmedEventNotification-Request | UnconfirmedEventNotification-Request,
 - 'Subscriber Process Identifier' = (a value acceptable to the IUT configured in the Notification Class object for the IUT),
 - 'Initiating Device Identifier' = TD1, // representing that O1 is in different device from SOURCE
 - 'Event Object Identifier' = O1,
 - 'Time Stamp' = (any valid value, T1),
 - 'Notification Class' = (the value configured in O1),
 - 'Priority' = (any value selected by the TD),
 - 'Event Type' = (any value selected by the TD),
 - 'Notify Type' = ALARM | EVENT,

- 'AckRequired' = TRUE,
- 'From State' = (any valid value),
- 'To State' = (any valid value, S1),
- 'Event Values' = (any event values appropriate to the event type)
- 2. IF (the ConfirmedEventNotification choice was selected) THEN
RECEIVE BACnet-SimpleACK-PDU
- 3. MAKE (the IUT acknowledge O1)
- 4. RECEIVE AcknowledgeAlarm-Request, DESTINATION=TD1
 - 'Acknowledging Process Identifier' = (any process identifier),
 - 'Event Object Identifier' = O1,
 - 'Event State Acknowledged' = S1, or OFFNORMAL if S1 is an off-normal state
 - 'Time Stamp' = T1,
 - 'Acknowledgement Source' = (any valid value),
 - 'Time of Acknowledgement' = (any valid value)
- 5. TRANSMIT BACnet-SimpleACK-PDU
- 6. TRANSMIT ConfirmedEventNotification-Request | UnconfirmedEventNotification-Request,
 - 'Subscriber Process Identifier' = (a value acceptable to the IUT configured in the Notification Class object for the IUT),
 - 'Initiating Device Identifier' = TD, // representing that O1 is present in same device as SOURCE
 - 'Event Object Identifier' = O1,
 - 'Time Stamp' = (any valid value, T1),
 - 'Notification Class' = (the value configured in O1),
 - 'Priority' = (any value selected by the TD),
 - 'Event Type' = (any value selected by the TD),
 - 'Notify Type' = ALARM | EVENT,
 - 'AckRequired' = TRUE,
 - 'From State' = (any valid value),
 - 'To State' = (any valid value, S1),
 - 'Event Values' = (any event values appropriate to the event type)
- 7. IF (the ConfirmedEventNotification choice was selected) THEN
RECEIVE BACnet-SimpleACK-PDU
- 8. MAKE (the IUT acknowledge O1)
- 9. RECEIVE AcknowledgeAlarm-Request,
 - 'Acknowledging Process Identifier' = (any process identifier),
 - 'Event Object Identifier' = O1,
 - 'Event State Acknowledged' = S1, or OFFNORMAL if S1 is an off-normal state
 - 'Time Stamp' = T1,
 - 'Acknowledgement Source' = (any valid value),
 - 'Time of Acknowledgement' = (any valid value)
- 10. TRANSMIT BACnet-SimpleACK-PDU