



BACnet<sup>®</sup> TESTING LABORATORIES

## TEST PLAN

**Revision 16.1 Final**  
**Revised December 10, 2019**

## Table Of Contents

<b>1</b>	<b>Test Plan Overview .....</b>	<b>1</b>
1.1	External Document References .....	2
<b>2</b>	<b>Basic BACnet Functionality.....</b>	<b>3</b>
2.1	Basic Functionality (applies to all BACnet Devices) .....	4
2.2	Segmentation Support.....	8
2.3	Private Transfer Services .....	9
<b>3</b>	<b>Objects.....</b>	<b>10</b>
3.1	Analog Input Object .....	11
3.2	Analog Output Object.....	12
3.3	Analog Value Object .....	13
3.4	Averaging Object.....	14
3.5	Binary Input Object .....	15
3.6	Binary Output Object.....	17
3.7	Binary Value Object .....	20
3.8	Calendar Object.....	23
3.9	Command Object.....	24
3.10	Device Object .....	26
3.11	Event Enrollment Object.....	28
3.12	Group Object .....	29
3.13	Loop Object.....	30
3.14	Multi-state Input Object.....	31
3.15	Multi-state Output Object .....	32
3.16	Multi-state Value Object.....	34
3.17	Notification Class Object.....	36
3.18	Proprietary Objects .....	38
3.19	Schedule Object.....	39
3.20	Trend Log Object.....	41
3.21	Structured View Object.....	42
3.22	Event Log Object.....	43
3.23	Trend Log Multiple Object .....	44
3.24	Bitstring Value Object .....	45
3.25	CharacterString Value Object .....	46
3.26	Date Pattern Value Object.....	47
3.27	Date Value Object .....	49
3.28	DateTime Pattern Value Object .....	51
3.29	DateTime Value Object .....	53
3.30	Integer Value Object.....	55
3.31	Large Analog Value Object .....	56
3.32	OctetString Value Object.....	57
3.33	Positive Integer Value Object .....	58
3.34	Time Pattern Value Object.....	59
3.35	Time Value Object.....	61
3.36	Global Group Object.....	63
3.37	Accumulator Object.....	68
3.38	Program Object.....	70
3.39	Life Safety Point Object.....	71
3.40	Life Safety Zone Object.....	73
3.41	Pulse Converter Object .....	75
3.42	Access Door Object .....	76
3.43	Load Control Object .....	79
3.44	Access Point Object.....	80
3.45	Access Zone Object.....	82
3.46	Access User Object.....	84
3.47	Access Rights Object.....	85

3.48	Access Credential Object.....	86
3.49	Credential Data Input Object.....	88
3.50	Network Security Object.....	89
3.51	Notification Forwarder Object.....	90
3.52	Alert Enrollment Object.....	93
3.53	Channel Object.....	94
3.54	Lighting Output Object.....	105
3.55	Binary Lighting Output Object.....	109
3.56	Network Port Object.....	112
3.57	Timer Object.....	113
3.58	Elevator Group Object.....	114
3.59	Lift Object.....	115
3.60	Escalator Object.....	116
3.61	File Object.....	117
3.62	Staging Object.....	119
3.63	Audit Reporter Object.....	120
3.64	Audit Log Object.....	121
<b>4</b>	<b>Data Sharing BIBBs.....</b>	<b>122</b>
4.1	Data Sharing - ReadProperty - A.....	123
4.2	Data Sharing - ReadProperty - B.....	129
4.3	Data Sharing - ReadPropertyMultiple - A.....	133
4.4	Data Sharing - ReadPropertyMultiple - B.....	141
4.5	Data Sharing - WriteProperty - A.....	146
4.6	Data Sharing - WriteProperty - B.....	151
4.7	Data Sharing - WritePropertyMultiple - A.....	156
4.8	Data Sharing - WritePropertyMultiple - B.....	163
4.9	Data Sharing - Change Of Value - A.....	169
4.10	Data Sharing - Change Of Value - B.....	176
4.11	Data Sharing - View - A.....	193
4.12	Data Sharing - Advanced View - A.....	194
4.13	Data Sharing - Modify - A.....	195
4.14	Data Sharing - Advanced Modify - A.....	196
4.15	Initiates ReadRange.....	197
4.16	Executes ReadRange.....	198
4.17	Data Sharing - Change Of Value Unsubscribed - A.....	199
4.18	Data Sharing - Change Of Value Unsubscribed - B.....	203
4.19	Data Sharing - Change Of Value Property - A.....	204
4.20	Data Sharing - Change Of Value Property - B.....	211
4.21	Data Sharing - WriteGroup - A.....	220
4.22	Data Sharing - WriteGroup - I - B.....	221
4.23	Data Sharing - WriteGroup - E - B.....	222
4.24	Data Sharing - Value Source Information - B.....	223
4.25	Data Sharing - Change Of Value Multiple - A.....	224
4.26	Data Sharing - Change Of Value Multiple - B.....	225
4.27	Data Sharing - Life Safety View - A.....	226
4.28	Data Sharing - Life Safety Advanced View - A.....	227
4.29	Data Sharing - Life Safety Modify - A.....	228
4.30	Data Sharing - Life Safety Advanced Modify - A.....	229
4.31	Data Sharing - Access Control View - A.....	230
4.32	Data Sharing - Access Control Advanced View - A.....	231
4.33	Data Sharing - Access Control Modify - A.....	232
4.34	Data Sharing - Access Control Advanced Modify - A.....	233
4.35	Data Sharing - Access Control User Configuration - A.....	234
4.36	Data Sharing - Access Control User Configuration - B.....	235
4.37	Data Sharing - Access Control Site Configuration - A.....	236
4.38	Data Sharing - Access Control Site Configuration - B.....	237

## BACnet Testing Laboratories - Test Plan

4.39	Data Sharing - Access Control Access Door - A .....	238
4.40	Data Sharing - Access Control Access Door - B.....	239
4.41	Data Sharing - Access Control Credential Data Input - A.....	240
4.42	Data Sharing - Access Control Credential Data Input - B.....	241
4.43	Data Sharing - Lighting Output - A.....	242
4.44	Data Sharing - Lighting Output Status - A .....	243
4.45	Data Sharing - Advanced Lighting Output - A .....	244
4.46	Data Sharing - Lighting Output - B.....	245
4.47	Data Sharing - Binary Lighting Output - B.....	246
4.48	Data Sharing - Lighting Output Management - A.....	247
4.49	Data Sharing - Lighting View - A .....	248
4.50	Data Sharing - Lighting Advanced View - A.....	249
4.51	Data Sharing - Lighting Modify - A.....	250
4.52	Data Sharing - Lighting Advanced Modify - A .....	251
<b>5</b>	<b>Alarm and Event Management BIBBs.....</b>	<b>256</b>
5.1	Alarm and Event Management - Notification - A.....	257
5.2	Alarm and Event Management - Notification - Internal - B .....	265
5.3	Alarm and Event Management - Notification - External - B.....	279
5.4	Alarm and Event Management - Acknowledge - A .....	288
5.5	Alarm and Event Management - Acknowledge - B .....	289
5.6	Alarm and Event Management - Alarm Summary - A.....	292
5.7	Alarm and Event Management - Alarm Summary - B.....	293
5.8	Alarm and Event Management - Enrollment Summary - A .....	294
5.9	Alarm and Event Management - Enrollment Summary - B.....	296
5.10	Alarm and Event Management - Information - A .....	298
5.11	Alarm and Event Management - Information - B .....	299
5.12	Alarm and Event Management - Event Log View - A .....	300
5.13	Alarm and Event Management - Event Log View and Modify - A .....	303
5.14	Alarm and Event Management - Event Log - Internal - B.....	304
5.15	Alarm and Event Management - Event Log - External - B .....	307
5.16	Alarm and Event Management - View Notifications - A .....	310
5.17	Alarm and Event Management - View Modify - A.....	311
5.18	Alarm and Event Management - Advanced View Notifications - A.....	312
5.19	Alarm and Event Management - Advanced View Modify - A .....	313
5.20	Alarm and Event Management - Alarm Summary View - A.....	314
5.21	Alarm and Event Management - LifeSafety - A .....	317
5.22	Alarm and Event Management - LifeSafety - B.....	320
5.23	Alarm and Event Management - Notification Forwarder - B .....	325
5.24	Alarm and Event Management - Notification Forwarder - I - B.....	327
5.25	Alarm and Event Management - Configurable Recipient Lists - B .....	329
5.26	Alarm and Event Management - Temporary Event Subscription - A .....	331
5.27	Alarm and Event Management - Life Safety View Notifications - A .....	332
5.28	Alarm and Event Management - Life Safety Advanced View Notifications - A .....	333
5.29	Alarm and Event Management - Life Safety View and Modify - A .....	334
5.30	Alarm and Event Management - Life Safety Advanced View and Modify - A.....	335
5.31	Alarm and Event Management - Access Control - A .....	336
5.32	Alarm and Event Management - Access Control - B .....	337
5.33	Alarm and Event Management - Access Control Advanced View Notifications - A .....	338
5.34	Alarm and Event Management - Access Control View and Modify - A.....	339
5.35	Alarm and Event Management - Access Control Advanced View and Modify - A .....	340
<b>6</b>	<b>Scheduling BIBBs.....</b>	<b>345</b>
6.1	Scheduling - Advanced View Modify - A .....	346
6.2	Scheduling - View Modify - A.....	348
6.3	Scheduling - Weekly Schedule - A.....	355
6.4	Scheduling - Internal - B.....	358
6.5	Scheduling - External - B.....	367

## BACnet Testing Laboratories - Test Plan

6.6	Scheduling - Weekly Schedule - Internal - B.....	371
6.7	Scheduling - Readonly - B .....	376
6.8	Scheduling - Schedule - A .....	384
6.9	Scheduling - Timer - I - B.....	385
6.10	Scheduling - Timer - E - B.....	386
<b>7</b>	<b>Trending BIBBs .....</b>	<b>387</b>
7.1	Trending - Viewing - A.....	388
7.2	Trending - Advanced View and Modify - A.....	393
7.3	Trending - View and Modify Trends - I - B.....	395
7.4	Trending - View and Modify Trends - E - B .....	398
7.5	Trending - Automated Trend Retrieval - A.....	401
7.6	Trending - Automated Trend Retrieval - B.....	403
7.7	Trending - View and Modify Multiple Values - I - B .....	406
7.8	Trending - View and Modify Multiple Values - E - B .....	411
7.9	Trending - Automated Multiple Value Retrieval - A .....	414
7.10	Trending - Automated Multiple Value Retrieval - B.....	415
7.11	Trending - Archival - A .....	418
7.12	Trending - View and Modify Trends - A.....	419
7.13	Trending - View and Modify Multiple Values - A.....	420
<b>8</b>	<b>Device Management BIBBs .....</b>	<b>421</b>
8.1	Device Management - Dynamic Device Binding - A.....	422
8.2	Device Management - Dynamic Device Binding - B.....	423
8.3	Device Management - Dynamic Object Binding - A .....	424
8.4	Device Management - Dynamic Object Binding - B.....	425
8.5	Device Management - Automatic Device Mapping - A.....	427
8.6	Device Management - Automatic Network Mapping - A .....	428
8.7	Device Management - Time Synchronization - A.....	429
8.8	Device Management - Time Synchronization - B.....	430
8.9	Device Management - UTC Time Synchronization - A .....	431
8.10	Device Management - UTC Time Synchronization - B .....	432
8.11	Device Management - Automatic Time Synchronization - A .....	433
8.12	Device Management - Manual Time Synchronization - A .....	434
8.13	Device Management - Device Communication Control - A.....	435
8.14	Device Management - Device Communication Control - B.....	437
8.15	Device Management - Reinitialize Device - A .....	439
8.16	Device Management - Reinitialize Device - B.....	440
8.17	Device Management - Backup and Restore - A .....	441
8.18	Device Management - Backup and Restore - B .....	442
8.19	Device Management - Restart - A .....	444
8.20	Device Management - Restart - B .....	445
8.21	Device Management - Object Creation and Deletion - A.....	446
8.22	Device Management - Object Creation and Deletion - B.....	463
8.23	Device Management - List Manipulation - A .....	487
8.24	Device Management - List Manipulation - B.....	506
8.25	Device Management - Text Message - A .....	508
8.26	Device Management - Text Message - B.....	510
8.27	Device Management - Virtual Terminal - A.....	512
8.28	Device Management - Virtual Terminal - B .....	513
8.29	Device Management - Slave Proxy - View and Modify - A.....	514
8.30	Device Management - Slave Proxy - B .....	515
<b>9</b>	<b>Data Link Layer.....</b>	<b>518</b>
9.1	Data Link Layer - MS/TP - Master Node .....	519
9.2	Data Link Layer - MS/TP - Slave Node .....	522
9.3	BACnet/IP - Annex J - non-BBMD Functionality .....	523
9.4	BACnet/IP - Annex J - BBMD.....	525
9.5	Data Link Layer - ZigBee .....	528

9.6	Data Link Layer - Ethernet .....	529
9.7	Data Link Layer - ARCNET .....	530
9.8	Data Link Layer - LonTalk .....	531
<b>10</b>	<b>Network Management.....</b>	<b>532</b>
10.1	Network Management - Routing .....	535
10.2	Network Management - Router Configuration - B.....	543
10.3	Network Management - Connection Establishment - A.....	544
10.4	Network Management - Connection Establishment - B.....	545
10.5	Network Management - Router Configuration - A .....	546
10.6	Network Management - BBMD Configuration - A.....	547
10.7	Network Management - BBMD Configuration - B.....	548
10.8	Network Management - Foreign Device Registration - A.....	549
<b>11</b>	<b>Gateway .....</b>	<b>554</b>
11.1	Gateway - Virtual Network - B .....	555
11.2	Gateway - Embedded Objects - B .....	556
<b>12</b>	<b>Network Security BIBBs.....</b>	<b>557</b>
12.1	Network Security - Secure Device .....	558
12.2	Network Security - Encrypted Device .....	559
12.3	Network Security - Multi-Application Device.....	560
12.4	Network Security - Device Master Key - A.....	561
12.5	Network Security - Device Master Key - B.....	562
12.6	Network Security - Key Server .....	563
12.7	Network Security - Temporary Key Server .....	564
12.8	Network Security - Secure Router.....	565
12.9	Network Security - Security Proxy.....	566
<b>13</b>	<b>Audit Reporting BIBBs.....</b>	<b>567</b>
13.1	Audit Reporting - Logging - A.....	568
13.2	Audit Reporting - Reporter - B.....	569
13.3	Audit Reporting - Reporter - Simple - B .....	570
13.4	Audit Reporting - Forwarder - B.....	571
13.5	Audit Reporting - View - A .....	572
13.6	Audit Reporting - Advanced View and Modify - A.....	573

# 1 Test Plan Overview

This document describes each functional item that the BTL will test and identifies the tests that will be used to test the item. Each test plan item in this document corresponds to a line item in the ***BTL Checklist*** document. It provides a more detailed description of the functional item, and the tests that the BTL will use to ensure the proper implementation of the functional item.

Each test that is to be applied for a specific functional item is described in a table entry. A sample entry is shown below.

The top line (test reference) of the table entry refers to the test. The first part of the reference specifies the test definition document that defines the test. This can be either ***ANSI/ASHRAE 135.1-2013***, the ***BTL Specified Tests*** document.

The Test Conditionality section specifies whether the test shall be executed or whether it shall be skipped. This item will either detail the conditions under which the test shall be skipped or it will refer the reader to the actual test definition. If the IUT meets the conditions laid out, then the test will not be applied.

The Testing Hints section provides useful information for the Test Planner to consider when choosing parameters for the test, or useful information for the Tester when applying the test.

The Test Directives section contains the test instructions for Verify Checklist tests or may contain the number of variants or parameters the test can run. A pretester shall always test all possible variants.

135.1-2013	7.3.2.23.4	Weekly Schedule and Exception Schedule Interaction Test
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

**Table 1:** A sample test plan entry.

## 1.1 External Document References

This version of the BTL Test Plan contains references to the following external documents.

Document name	Description
135.1-2013	SSPC Test Standard. This document is the same as 2011 with addenda <i>j, k, l, m, and n</i> .
135.1-2013 <sub>q</sub>	SSPC Test standard Addendum 135.1-2013q (This addendum is final and was approved by ASHRAE and by the American National Standards Institute on December 7, 2018.). This addenda updated alarm and event tests for protocol revisions 13 and higher.
135.1-2013 <sub>r</sub>	SSPC Test standard Addendum 135.1-2013r (This addendum is final and was approved by ASHRAE on May 31, 2018 and by the American National Standards Institute on June 1, 2018). This addenda added Property_List property tests and tests for DUPLICATE_ENTRY error code.
135-2012	BACnet standard revision number 14. This document is the same as 135-2010 with addenda <i>i, aa, ad, ae, af, ak</i> and <i>ao</i> applied.
135-2012 <sub>ar</sub>	BACnet standard revision number 15.
135-2012 <sub>an</sub> , 135-2012 <sub>at</sub> 135-2012 <sub>au</sub> 135-2012 <sub>av</sub> 135-2012 <sub>aw</sub> 135-2012 <sub>ax</sub> 135-2012 <sub>az</sub>	BACnet standard revision number 16.

BTL Specified Tests are assumed to be referenced to the “BTL Specified Tests” document which filename contains the same version number as the BTL Test Plan document.



---

## **2 Basic BACnet Functionality**

---

## 2.1 Basic Functionality (applies to all BACnet Devices)

### 2.1.1 Base Requirements

All BACnet devices must meet these base requirements.

<b>BTL - 10.1.1 - Processing Application Layer Messages Originating from Remote Networks</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 10.6.1 - Ignore Remote Packets</b>		
	<b>Test Conditionality</b>	Must be executed unless the IUT's routing functionality cannot be disabled.
	<b>Test Directives</b>	The IUT's routing functionality must be disabled. Ensure that the packet actually reaches the IUT. Test using an address which resembles the actual address of IUT. Also test using a DNET of 0 and with a packet still containing aDNET of the actual network of IUT when it reaches the IUT.
	<b>Testing Hints</b>	Note that a routed packet, which contained the DNET of the actual network of IUT before the router strips that off in transmitting it locally to the IUT, is not expected to be ignored.
<b>135.1-2013 - 10.6.2 - Ignore Who-Is-Router-To-Network</b>		
	<b>Test Conditionality</b>	Must be executed unless the IUT's routing functionality cannot be disabled.
	<b>Test Directives</b>	The IUT's routing functionality must be disabled.
	<b>Testing Hints</b>	
<b>BTL - 10.6.3 - Ignore Router Commands</b>		
	<b>Test Conditionality</b>	Must be executed unless the IUT's routing functionality cannot be disabled.
	<b>Test Directives</b>	The IUT's routing functionality must be disabled.
	<b>Testing Hints</b>	
<b>135.1-2013 - 13.4.3 - Invalid Tag</b>		
	<b>Test Conditionality</b>	Must be executed
	<b>Test Directives</b>	
	<b>Testing Hints</b>	In VTS, the needed invalid packets can be created through the dedicated menu item devoted to test 13.4.3.
<b>135.1-2013 - 13.4.4 - Missing Required Parameter</b>		
	<b>Test Conditionality</b>	Must be executed
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 13.4.5 - Too Many Arguments</b>		
	<b>Test Conditionality</b>	Must be executed
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.39.1 - Unsupported Confirmed Services Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Don't forget Confirmed Private Transfer; if the IUT supports Confirmed Private Transfer, send one with a different vendor Id. Also try services that have not yet been defined (reserved enumeration range), sending them in a BACnet-Confirmed-Request-PDU with DER bit set.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.39.2 - Unsupported Unconfirmed Services Test</b>		
	<b>Test Conditionality</b>	Must be executed.

	<b>Test Directives</b>	Don't forget Unconfirmed Private Transfer; if the IUT supports Unconfirmed Private Transfer, send one with a different vendor Id. Also try services that have not yet been defined (reserved enumeration range), sending them in a BACnet-Unconfirmed-Request-PDU with DER bit clear and with no InvokeID.
	<b>Testing Hints</b>	
<b>BTL - 13.1.12.1 - IUT Does Not Support Segmented Response</b>		
	<b>Test Conditionality</b>	Must be executed if the device does <u>not</u> support the transmission of segmented responses, otherwise the test is omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	The test assumes that the device supports DS-RPM-B. If the device does not support DS-RPM-B, substitute a ReadProperty request of the Priority Array property or another very large property. A test variation would be to change the value of the Max-APDU-Length-Accepted parameter in the request.
<b>BTL - 13.X13.1 - Ignore Confirmed Broadcast Requests</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

## 2.1.2 EPICS Consistency Tests

<b>BTL - 5 - EPICS Consistency Tests</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	The EPICS and the IUT Configuration must match exactly. The EPICS must note the existence of every object and property, resolution, and range restrictions if any for writable values. All writable standard properties must be claimed.
	<b>Testing Hints</b>	
<b>BTL - 7.2.3 - Read-only Property Test</b>		
	<b>Test Conditionality</b>	If the IUT does not support the WriteProperty service, then this test shall be skipped.
	<b>Test Directives</b>	This test ensures accuracy of the EPICS, if the EPICS is changed this test shall be run again.
	<b>Testing Hints</b>	
<b>BTL - 7.1.2 - Non-documented Property Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test ensures accuracy of the EPICS, if the EPICS is changed this test shall be run again.
	<b>Testing Hints</b>	

## 2.1.3 Supports DS-RP-B

The IUT supports DS-RP-B.

<b>Verify Checklist</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support for DS-RP-B.
	<b>Testing Hints</b>	

## 2.1.4 Uses Who-Is and I-Am Services for Router Address Discovery

The IUT can generate a broadcast Who-Is message. The IUT examines the I-Am response message(s) to determine the MAC address of a router that can forward messages to a particular network.

<b>BTL - 10.7.2 - Router Binding via Application Layer Services</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 2.1.5 Uses Who-Is-Router-To-Network (any Network) for Router Address Discovery

The IUT can generate a Who-Is-Router-To-Network message with no network number specified. The IUT examines the I-Am-Router-To-Network message(s) sent in response to determine the MAC address of a router that can forward messages to a particular network.

<b>135.1-2013 - 10.7.3 - Router Binding via Who-Is-Router-To-Network</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 10.5.1.1 - Who-Is-Router-To-Network - General Query</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 2.1.6 Uses Who-Is-Router-To-Network (Specific Network) for Router Address Discovery

The IUT can generate a Who-Is-Router-To-Network message, with a specific network specified. The IUT examines the I-Am-Router-To-Network message sent in response to determine the MAC address of a router that can forward messages to the specific network.

<b>135.1-2013 - 10.7.3 - Router Binding via Who-Is-Router-To-Network</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 10.5.1.2 - Who-Is-Router-To-Network - Specific Network Number</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 2.1.7 Uses MAC Broadcasts for Router Discovery

The IUT sends a request to a device connected to a different network using a MAC broadcast on the IUT's local network. The IUT determines the MAC address of the router to the other network by examining the message sent by the device in response to the request.

<b>135.1-2013 - 10.7.4 - Router Binding via Broadcast</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 2.1.8 Uses Static Router Address Configuration

The IUT can be statically configured with the MAC address of a router to a particular network, permitting the IUT to send requests to devices connected to that network.

<b>135.1-2013 - 10.7.1 - Static Router Binding</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 2.1.9 Supports DM-DDB-B

The IUT supports DM-DDB-B.

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

### 2.1.10 Supports DM-DOB-B

The IUT supports DM-DOB-B.

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

### 2.1.11 Initiates Confirmed Requests

The IUT initiates any BACnet confirmed requests.

135.1-2013 - 13.9.1 - APDU Retry and Timeout		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

## 2.2 Segmentation Support

### 2.2.1 Base Requirements

All BACnet devices must meet these base requirements.

<b>BTL - 7.3.2.10.X2 - Max_Segments_Accepted at least the minimum</b>		
	<b>Test Conditionality</b>	If the Segmentation_Supported property of Device object cannot be made to hold a value other than NO-SEGMENTATION, then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 2.2.2 Issues Segmented Responses

BACnet devices which are capable of issuing segmented responses shall meet these requirements.

<b>BTL - 9.18.1.X3 - Respects max-segments-accepted bit pattern</b>		
	<b>Test Conditionality</b>	Must be executed, if there is any response generated by IUT which is large enough to require segmentation.
	<b>Test Directives</b>	Test with a 'max-segments-accepted' bit pattern value that is less than TD's Max_Segments_Accepted property value, to ascertain that the 'max-segments-accepted' bit pattern value is what governs.
	<b>Testing Hints</b>	

### 2.2.3 Accepts Segmented Responses without Specifying the Maximum

BACnet devices which accept segmented responses shall meet these requirements.

<b>BTL - 13.1.12.X1 - Reading with maximum-segments-accepted bit pattern B'000'</b>		
	<b>Test Conditionality</b>	If the IUT cannot be configured to issue any BACnet-Confirmed-Request-PDU with 'segmented-response-accepted' = TRUE and the 'max-segments-accepted' parameter equal to B'000', then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

## 2.3 Private Transfer Services

### 2.3.1 Base Requirements

Base requirements must be met by any IUT that initiates or executes private transfer services.

### 2.3.2 Initiates ConfirmedPrivateTransfer

The IUT initiates ConfirmedPrivateTransfer service requests.

135.1-2013 - 8.25 - ConfirmedPrivateTransfer Service Initiation Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 2.3.3 Initiates UnconfirmedPrivateTransfer

The IUT initiates UnconfirmedPrivateTransfer service requests.

135.1-2013 - 8.26 - UnconfirmedPrivateTransfer Service Initiation Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 2.3.4 Executes ConfirmedPrivateTransfer

The IUT initiates ConfirmedPrivateTransfer service requests.

BTL - 9.25.1.1 - Correctly Executes a Supported ConfirmedPrivateTransfer Service		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
BTL - 9.25.2.1 - Correctly Executes a Non-Supported ConfirmedPrivateTransfer Service		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 2.3.5 Executes UnconfirmedPrivateTransfer

The IUT executes UnconfirmedPrivateTransfer service requests.

There are no tests for this functionality.

---

## **3 Objects**

---



## 3.1 Analog Input Object

### 3.1.1 Base Requirements

Base requirements must be met by any IUT that can contain Analog Input objects.

### 3.1.2 Supports Writable Out\_Of\_Service Property

The Out\_Of\_Service property in Analog Input objects contained in the IUT are writable.

135.1-2013 - 7.3.1.1 - Out Of Service, Status Flags, and Reliability Tests		
	<b>Test Conditionality</b>	If this property is writable, this test must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 3.1.3 Contains an object with Reliability\_Evaluation\_Inhibit Property

The IUT contains, or can be made to contain, a Reliability\_Evaluation\_Inhibit property that is configurable to a value of TRUE.

BTL - 7.3.1.X8.1 - Reliability Evaluation Inhibit Test		
	<b>Test Conditionality</b>	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
BTL - 7.3.1.X8.2 - Reliability Evaluation Inhibit Summarization Test		
	<b>Test Conditionality</b>	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

## 3.2 Analog Output Object

### 3.2.1 Base Requirements

Base requirements must be met by any IUT that can contain Analog Output objects.

### 3.2.2 Supports Command Prioritization

The objects contain a priority array and support command prioritization.

135.1-2013 - 7.3.1.2 - Relinquish Default Test		
	Test Conditionality	If no object can be made to meet the configuration requirements, this test shall be skipped.
	Test Directives	
	Testing Hints	
135.1-2013 - 7.3.1.3 - Command Prioritization Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 3.2.3 Supports Writable Out\_Of\_Service Property

The Out\_Of\_Service property in Analog Output objects contained in the IUT are writable.

135.1-2013 - 7.3.1.1 - Out Of Service, Status Flags, and Reliability Tests		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 3.2.4 Contains an object with Reliability\_Evaluation\_Inhibit Property

The IUT contains, or can be made to contain, a Reliability\_Evaluation\_Inhibit property that is configurable to a value of TRUE.

BTL - 7.3.1.X8.1 - Reliability Evaluation_Inhibit Test		
	Test Conditionality	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	Test Directives	
	Testing Hints	
BTL - 7.3.1.X8.2 - Reliability Evaluation_Inhibit Summarization Test		
	Test Conditionality	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	Test Directives	
	Testing Hints	

### 3.3 Analog Value Object

#### 3.3.1 Base Requirements

Base requirements must be met by any IUT that can contain Analog Value objects. There are no base requirements tests for this section.

#### 3.3.2 Supports Writable Out\_Of\_Service Property

The Out\_Of\_Service property in Analog Value objects contained in the IUT are writable.

135.1-2013 - 7.3.1.1 - Out_Of_Service, Status_Flags, and Reliability Tests		
	Test Conditionality	If this property is writable, this test must be executed.
	Test Directives	
	Testing Hints	

#### 3.3.3 Supports Command Prioritization

The objects contain a priority array and support command prioritization.

135.1-2013 - 7.3.1.2 - Relinquish Default Test		
	Test Conditionality	If no object can be made to meet the configuration requirements, this test shall be skipped.
	Test Directives	
	Testing Hints	
135.1-2013 - 7.3.1.3 - Command Prioritization Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

#### 3.3.4 Contains an object with Reliability\_Evaluation\_Inhibit Property

The IUT contains, or can be made to contain, a Reliability\_Evaluation\_Inhibit property that is configurable to a value of TRUE.

BTL - 7.3.1.X8.1 - Reliability_Evaluation_Inhibit Test		
	Test Conditionality	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	Test Directives	
	Testing Hints	
BTL - 7.3.1.X8.2 - Reliability_Evaluation_Inhibit Summarization Test		
	Test Conditionality	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	Test Directives	
	Testing Hints	

---

## 3.4 Averaging Object

---

### 3.4.1 Base Requirements

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

<b>BTL - 7.3.2.4.1 - Reinitializing the Samples</b>		
	<b>Test Conditionality</b>	Must be executed
	<b>Test Directives</b>	Repeat for each datatype that the IUT supports in an Averaging object.
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.4.2 - Managing the Sample Window</b>		
	<b>Test Conditionality</b>	Must be executed
	<b>Test Directives</b>	Repeat for each datatype that the IUT supports in an Averaging object.
	<b>Testing Hints</b>	

## 3.5 Binary Input Object

### 3.5.1 Base Requirements

Base requirements must be met by any IUT that can contain Binary Input objects. There are no base requirements for this object.

### 3.5.2 Supports Writable Out\_Of\_Service Property

The Out\_Of\_Service property in Binary Input objects contained in the IUT are writable.

135.1-2013 - 7.3.1.1 - Out_Of_Service, Status_Flags, and Reliability Tests		
	Test Conditionality	If this property is writable, this test must be executed.
	Test Directives	
	Testing Hints	

### 3.5.3 Supports Writable Polarity Property

The IUT supports a writable Polarity property in the Binary Input objects.

135.1-2013 - 7.3.2.5.3 - Polarity Property Tests		
	Test Conditionality	Must be executed
	Test Directives	
	Testing Hints	

### 3.5.4 Supports Change Of State Tracking

The IUT contains or can be made to contain an object with the Change\_Of\_State\_Time, Change\_Of\_State\_Count and Time\_Of\_State\_Count\_Reset properties.

BTL - 7.3.1.8 - Change of State Tests		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
BTL - 7.3.1.X18 - Non-zero Writable State Count Test		
	Test Conditionality	If no Binary Input object contains a writable Change_Of_State_Count that accepts writes of non-zero values then this test shall be skipped. If IUT claims Protocol_Revision less than 16, then this test shall be skipped.
	Test Directives	This test shall be performed using a Binary Input object.
	Testing Hints	

### 3.5.5 Supports Elapsed Active Time Tracking

The IUT contains or can be made to contain an object with the Elapsed\_Active\_Time and Time\_Of\_Active\_Time\_Reset properties.

BTL - 7.3.1.9 - Elapsed Active Time Tests		
	Test Conditionality	If no Binary Input object contains a writable Elapsed_Active_Time then this test shall be skipped.
	Test Directives	This test shall be performed using a Binary Input object.
	Testing Hints	
BTL - 7.3.1.X19 - Non-zero Writable Elapsed Active Time Test		
	Test Conditionality	If no Binary Input object contains a writable Elapsed_Active_Time that accepts writes of non-zero values then this test shall be skipped. If IUT claims Protocol_Revision less than 16, then this test shall be skipped.
	Test Directives	
	Testing Hints	

	<b>Test Directives</b>	This test shall be performed using a Binary Input object.
	<b>Testing Hints</b>	

### 3.5.6 Contains an object with Reliability\_Evaluation\_Inhibit Property

The IUT contains, or can be made to contain, a Reliability\_Evaluation\_Inhibit property that is configurable to a value of TRUE.

<b>BTL - 7.3.1.X8.1 - Reliability_Evaluation_Inhibit Test</b>		
	<b>Test Conditionality</b>	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.X8.2 - Reliability_Evaluation_Inhibit Summarization Test</b>		
	<b>Test Conditionality</b>	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

## 3.6 Binary Output Object

### 3.6.1 Base Requirements

Base requirements must be met by any IUT that can contain Binary Output objects. There are no base requirements for this object.

### 3.6.2 Supports Command Prioritization

135.1-2013 - 7.3.1.2 - Relinquish Default Test		
	Test Conditionality	If no object can be made to meet the configuration requirements, this test shall be skipped.
	Test Directives	
	Testing Hints	
135.1-2013 - 7.3.1.3 - Command Prioritization Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 3.6.3 Supports Writable Out\_Of\_Service Property

The Out\_Of\_Service property in Binary Output objects contained in the IUT are writable.

135.1-2013 - 7.3.1.1 - Out_Of_Service, Status_Flags, and Reliability Tests		
	Test Conditionality	If this property is writable, this test must be executed.
	Test Directives	
	Testing Hints	

### 3.6.4 Supports writable Polarity property

The IUT supports a writable Polarity property in the Binary Output object.

135.1-2013 - 7.3.2.6.3 - Polarity Property Tests		
	Test Conditionality	Must be executed
	Test Directives	
	Testing Hints	

### 3.6.5 Supports Change Of State Tracking

The IUT contains or can be made to contain an object with the Change\_Of\_State\_Time, Change\_Of\_State\_Count and Time\_Of\_State\_Count\_Reset properties.

BTL - 7.3.1.8 - Change of State Tests		
	Test Conditionality	Must be executed.
	Test Directives	This test shall be performed using a Binary Output Object.
	Testing Hints	
BTL - 7.3.1.X18 - Non-zero Writable State Count Test		
	Test Conditionality	If no Binary Output object contains a writable Change_Of_State_Count that accepts writes of non-zero values then this test shall be skipped. If IUT claims Protocol_Revision less than 16, then this test shall be skipped.
	Test Directives	This test shall be performed using a Binary Output object.
	Testing Hints	

### 3.6.6 Supports Elapsed Active Time Tracking

The IUT contains or can be made to contain an object with the Elapsed\_Active\_Time and Time\_Of\_Active\_Time\_Reset properties.

<b>BTL - 7.3.1.9 - Elapsed Active Time Tests</b>		
	<b>Test Conditionality</b>	If no Binary Output object contains a writable Elapsed_Active_Time then this test shall be skipped.
	<b>Test Directives</b>	This test shall be performed using a Binary Output object.
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.X19 - Non-zero Writable Elapsed Active Time Test</b>		
	<b>Test Conditionality</b>	If no Binary Output object contains a writable Elapsed_Active_Time that accepts writes of non-zero values then this test shall be skipped. If IUT claims Protocol_Revision less than 16, then this test shall be skipped
	<b>Test Directives</b>	This test shall be performed using a Binary Output object.
	<b>Testing Hints</b>	

### 3.6.7 Supports Minimum\_Off\_Time

The object contains Minimum\_Off\_Time property.

<b>135.1-2013 - 7.3.1.4 - Minimum Off Time</b>		
	<b>Test Conditionality</b>	If the property is present, it must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.6.1 - Override of Minimum Time</b>		
	<b>Test Conditionality</b>	If the property is present, it must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.6.2 - Minimum Off Time - Writing at priorities numerically greater than 6</b>		
	<b>Test Conditionality</b>	If the property is present, it must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.6.4 - Minimum Off Time - Writing at priorities numerically lesser than 6</b>		
	<b>Test Conditionality</b>	If the property is present, it must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.6.6 - Minimum Off Time - Clock is not affected by additional write operations</b>		
	<b>Test Conditionality</b>	If the property is present, it must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.6.8 - Ensuring Minimum Off Time starts at transition to INACTIVE</b>		
	<b>Test Conditionality</b>	If Minimum_On_Time and Minimum_Off_Time properties are present, this test must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.6.10 - Ensuring Minimum Times Are Not Affected By Time Changes</b>		
	<b>Test Conditionality</b>	If the property is present and the device can execute TimeSynchronization or UTCTimeSynchronization requests, this test must be executed
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 3.6.8 Supports Minimum\_On\_Time

The object contains Minimum\_On\_Time property.



<b>135.1-2013 - 7.3.1.5 - Minimum On Time</b>		
	<b>Test Conditionality</b>	If the property is present, it must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.6.1 - Override of Minimum Time</b>		
	<b>Test Conditionality</b>	If the property is present, it must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.6.3 - Minimum On Time - Writing at priorities numerically greater than 6</b>		
	<b>Test Conditionality</b>	If the property is present, it must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.6.5 - Minimum On Time - Writing at priorities numerically lesser than 6</b>		
	<b>Test Conditionality</b>	If the property is present, it must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.6.7 - Minimum On Time - Clock is not affected by additional write operations</b>		
	<b>Test Conditionality</b>	If the property is present, it must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.6.9 - Ensuring Minimum On Time starts at transition to ACTIVE</b>		
	<b>Test Conditionality</b>	If Minimum_On_Time and Minimum_Off_Time properties are present, this test must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.6.10 - Ensuring Minimum Times Are Not Affected By Time Changes</b>		
	<b>Test Conditionality</b>	If the property is present and the device can execute TimeSynchronization or UTCTimeSynchronization requests, this test must be executed
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 3.6.9 Contains an object with Reliability\_Evaluation\_Inhibit Property

The IUT contains, or can be made to contain, a Reliability\_Evaluation\_Inhibit property that is configurable to a value of TRUE.

<b>BTL - 7.3.1.X8.1 - Reliability Evaluation Inhibit Test</b>		
	<b>Test Conditionality</b>	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.X8.2 - Reliability Evaluation Inhibit Summarization Test</b>		
	<b>Test Conditionality</b>	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

## 3.7 Binary Value Object

### 3.7.1 Base Requirements

Base requirements must be met by any IUT that can contain Binary Value objects. There are no Base Requirements tests for this BIBB.

### 3.7.2 Supports Writable Out\_Of\_Service Property

The Out\_Of\_Service property in Binary Value objects contained in the IUT are writable.

135.1-2013 - 7.3.1.1 - Out_Of_Service, Status Flags, and Reliability Tests		
	Test Conditionality	If this property is writable, this test must be executed.
	Test Directives	
	Testing Hints	

### 3.7.3 Supports Change Of State Tracking

The IUT contains or can be made to contain an object with the Change\_Of\_State\_Time, Change\_Of\_State\_Count and Time\_Of\_State\_Count\_Reset properties.

BTL - 7.3.1.8 - Change of State Tests		
	Test Conditionality	Must be executed.
	Test Directives	This test shall be performed using a Binary Value object.
	Testing Hints	
BTL - 7.3.1.X18 - Non-zero Writable State Count Test		
	Test Conditionality	If no Binary Value object contains a writable Change_Of_State_Count that accepts writes of non-zero values then this test shall be skipped. If IUT claims Protocol_Revision less than 16, then this test shall be skipped.
	Test Directives	This test shall be performed using a Binary Value object.
	Testing Hints	

### 3.7.4 Supports Elapsed Active Time Tracking

The IUT contains or can be made to contain an object with the Elapsed\_Active\_Time and Time\_Of\_Active\_Time\_Reset properties.

BTL - 7.3.1.9 - Elapsed Active Time Tests		
	Test Conditionality	If no Binary Value object contains a writable Elapsed_Active_Time then this test shall be skipped.
	Test Directives	This test shall be performed using a Binary Value object.
	Testing Hints	
BTL - 7.3.1.X19 - Non-zero Writable Elapsed Active Time Test		
	Test Conditionality	If no Binary Value object contains a writable Elapsed_Active_Time that accepts writes of non-zero values then this test shall be skipped. If IUT claims Protocol_Revision less than 16, then this test shall be skipped.
	Test Directives	This test shall be performed using a Binary Value object.
	Testing Hints	

### 3.7.5 Supports Command Prioritization

The objects contain a priority array and support command prioritization.

135.1-2013 - 7.3.1.2 - Relinquish Default Test		
	Test Conditionality	If no object can be made to meet the configuration requirements, this test shall be skipped.

	Test Directives	
	Testing Hints	
	<b>135.1-2013 - 7.3.1.3 - Command Prioritization Test</b>	
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 3.7.6 Supports Minimum\_Off\_Time

The object contains Minimum\_Off\_Time property.

<b>135.1-2013 - 7.3.1.4 - Minimum Off Time</b>		
	Test Conditionality	If the property is present, it must be executed.
	Test Directives	
	Testing Hints	
<b>BTL - 7.3.1.6.1 - Override of Minimum Time</b>		
	Test Conditionality	If the property is present, it must be executed.
	Test Directives	
	Testing Hints	
<b>BTL - 7.3.1.6.2 - Minimum Off Time - Writing at priorities numerically greater than 6</b>		
	Test Conditionality	If the property is present, it must be executed.
	Test Directives	
	Testing Hints	
<b>BTL - 7.3.1.6.4 - Minimum Off Time - Writing at priorities numerically lesser than 6</b>		
	Test Conditionality	If the property is present, it must be executed.
	Test Directives	
	Testing Hints	
<b>BTL - 7.3.1.6.6 - Minimum Off Time - Clock is not affected by additional write operations</b>		
	Test Conditionality	If the property is present, it must be executed.
	Test Directives	
	Testing Hints	
<b>BTL - 7.3.1.6.8 - Ensuring Minimum_Off_Time starts at transition to INACTIVE</b>		
	Test Conditionality	If Minimum_On_Time and Minimum_Off_Time properties are present, this test must be executed.
	Test Directives	
	Testing Hints	
<b>BTL - 7.3.1.6.10 - Ensuring Minimum Times Are Not Affected By Time Changes</b>		
	Test Conditionality	If the property is present and the device can execute TimeSynchronization or UTCTimeSynchronization requests, this test must be executed
	Test Directives	
	Testing Hints	

### 3.7.7 Supports Minimum\_On\_Time

The object contains Minimum\_On\_Time property.

<b>135.1-2013 - 7.3.1.5 - Minimum On Time</b>		
	Test Conditionality	If the property is present, it must be executed.
	Test Directives	
	Testing Hints	
<b>BTL - 7.3.1.6.1 - Override of Minimum Time</b>		
	Test Conditionality	If the property is present, it must be executed.
	Test Directives	
	Testing Hints	

<b>BTL - 7.3.1.6.3 - Minimum On Time - Writing at priorities numerically greater than 6</b>		
	<b>Test Conditionality</b>	If the property is present, it must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.6.5 - Minimum On Time - Writing at priorities numerically lesser than 6</b>		
	<b>Test Conditionality</b>	If the property is present, it must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.6.7 - Minimum On Time - Clock is not affected by additional write operations</b>		
	<b>Test Conditionality</b>	If the property is present, it must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.6.9 - Ensuring Minimum On Time starts at transition to ACTIVE</b>		
	<b>Test Conditionality</b>	If Minimum_On_Time and Minimum_Off_Time properties are present, this test must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.6.10 - Ensuring Minimum Times Are Not Affected By Time Changes</b>		
	<b>Test Conditionality</b>	If the property is present and the device can execute TimeSynchronization or UTCTimeSynchronization requests, this test must be executed
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 3.7.8 Contains an object with Reliability\_Evaluation\_Inhibit Property

The IUT contains, or can be made to contain, a Reliability\_Evaluation\_Inhibit property that is configurable to a value of TRUE.

<b>BTL - 7.3.1.X8.1 - Reliability Evaluation Inhibit Test</b>		
	<b>Test Conditionality</b>	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.X8.2 - Reliability Evaluation Inhibit Summarization Test</b>		
	<b>Test Conditionality</b>	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

## 3.8 Calendar Object

### 3.8.1 Base Requirements

Base requirements must be met by any IUT that can contain Calendar Objects

<b>135.1-2013 - 7.3.2.8.1 - Single Date Rollover Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.8.2 - Date Range Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.8.3 - WeekNDay Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.2.X1 - Date Pattern Properties Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Apply to the Date_List property.
	<b>Testing Hints</b>	
<b>BTL - 7.2.X7 - BACnetDateRange Non-Pattern Properties Test</b>		
	<b>Test Conditionality</b>	This test shall only be applied to devices claiming Protocol_Revision 11 or higher.
	<b>Test Directives</b>	Apply to Date_List property.
	<b>Testing Hints</b>	
<b>BTL - 7.2.X8 - BACnetDateRange Open-Ended Pattern Properties Test</b>		
	<b>Test Conditionality</b>	This test shall only be applied to devices claiming Protocol_Revision 11 or higher.
	<b>Test Directives</b>	Apply to Date_List property.
	<b>Testing Hints</b>	
<b>BTL - 9.23.2.X12 - BACnetDateRange Non-Pattern Properties Test using WritePropertyMultiple Service</b>		
	<b>Test Conditionality</b>	This test shall only be applied to devices claiming Protocol_Revision 11 or higher and which supports execution of WritePropertyMultiple.
	<b>Test Directives</b>	Apply to the Date_List property.
	<b>Testing Hints</b>	

## 3.9 Command Object

### 3.9.1 Base Requirements

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

<b>135.1-2013 - 7.3.2.9.2 - Quit on Failure Test</b>		
	<b>Test Conditionality</b>	Must be executed
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.9.4 - Empty Action List Test</b>		
	<b>Test Conditionality</b>	Must be executed
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.9.5 - Action 0 Test</b>		
	<b>Test Conditionality</b>	Must be executed
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.9.7 - Write While In Process is TRUE Test.</b>		
	<b>Test Conditionality</b>	Must be executed
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 3.9.2 Supports Action\_Text

The size of the Action array corresponds to the size of the Action\_Text array.

<b>135.1-2013 - 7.3.2.9.6 - Action_Text Test</b>		
	<b>Test Conditionality</b>	If the property is supported, the test must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 3.9.3 Supports Post\_Delay

The properties of binary objects that collectively track state changes function as required.

<b>135.1-2013 - 7.3.2.9.1 - All Writes Successful with Post Delay Test</b>		
	<b>Test Conditionality</b>	If the prescribed test can be configured, it must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 3.9.4 Supports External Writes

A Command object can write to external objects.

<b>135.1-2013 - 7.3.2.9.3 - External Writes Test</b>		
	<b>Test Conditionality</b>	If the IUT does supports writing to external objects from a Command object, the test must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 3.9.5 Supports Resizable Action and Action\_Text Properties

The IUT is protocol revision 4 or higher and the IUT contains, or can be made to contain, Action and Action\_Text properties that are resizable by writing to the array.

<b>135.1-2013 - 7.3.2.9.8 - Action Size Changes Action_Text Size Test</b>		
---	--	--

# BTL Test Plan

	<b>Test Conditionality</b>	Must be executed if the IUT claims Protocol_Revision 4 or greater and supports a resizable Action property.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.9.9 - Action Text Size Changes Action Size Test</b>		
	<b>Test Conditionality</b>	Must be executed if the IUT claims Protocol_Revision 4 or greater and supports a resizable Action property.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

## 3.10 Device Object

### 3.10.1 Base Requirements

Base requirements must be met by any IUT that can contain Device objects. There are no base requirements tests for this section.

### 3.10.2 Supports Database\_Revision Property

Ensures that the Database\_Revision property increments correctly.

135.1-2013 - 7.3.2.10.3 - Successful Increment of the Database_Revision Property after Creating an Object		
	Test Conditionality	If the device implements protocol revision 2 or higher, this test must be executed. If the IUT does not support object creation through any means, not just through BACnet services, this test may be skipped.
	Test Directives	
	Testing Hints	
135.1-2013 - 7.3.2.10.4 - Successful Increment of the Database_Revision Property after Deleting an Object		
	Test Conditionality	If the device implements protocol revision 2 or higher, this test must be executed. If the IUT does not support object deletion through any means, not just through BACnet services, this test may be skipped.
	Test Directives	
	Testing Hints	
135.1-2013 - 7.3.2.10.5 - Successful Increment of the Database_Revision Property after Changing the Object_Name Property of an Object		
	Test Conditionality	If the device implements protocol revision 2 or higher, this test must be executed. If the IUT does not support a changeable Object_Name property in any object, this test may be skipped.
	Test Directives	
	Testing Hints	
BTL - 7.3.2.10.6 - Successful Increment of the Database_Revision Property after Changing the Object_Identifier Property of an Object		
	Test Conditionality	If the device implements protocol revision 2 or higher, this test must be executed. If the IUT does not support a changeable Object_Identifier property in any object, this test may be skipped.
	Test Directives	
	Testing Hints	

### 3.10.3 Supports Time\_Synchronization\_Recipients

The Time\_Synchronization\_Recipients property is present in the Device Object.

135.1-2013 - 13.2.1 - TimeSynchronization Recipients Test, Protocol_Revision < 7		
	Test Conditionality	Must be executed if Protocol_Revision < 7.
	Test Directives	
	Testing Hints	

### 3.10.4 Supports UTC\_Time\_Synchronization\_Recipients

The UTC\_Time\_Synchronization\_Recipients property is present in the Device Object.

Verify Checklist		
	Test Conditionality	Must be executed if Protocol_Revision ≥ 7.



	<b>Test Directives</b>	Verify that the IUT claims support for DM-ATS-A in the checklist.
	<b>Testing Hints</b>	

### 3.10.5 Contains a Writable Local\_Date Property

The IUT contains, or can be made to contain, a Device object that contains a writable Local\_Date property.

<b>BTL - 7.2.X4 - Date Non-Pattern Properties Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Apply to the Local_Date property.
	<b>Testing Hints</b>	
<b>BTL - 9.23.2.X9 - Date Non-Pattern Properties Test using WritePropertyMultiple Service</b>		
	<b>Test Conditionality</b>	This test shall only be applied to devices claiming Protocol_Revision 11 or higher and which supports execution of WritePropertyMultiple.
	<b>Test Directives</b>	Apply to the Local_Date property.
	<b>Testing Hints</b>	

### 3.10.6 Contains a Writable Local\_Time Property

The IUT contains, or can be made to contain, a Device object that contains a writable Local\_Time property.

<b>BTL - 7.2.X5 - Time Non-Pattern Properties Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Apply to the Local_Time property.
	<b>Testing Hints</b>	
<b>BTL - 9.23.2.X10 - Time Non-Pattern Properties Test using WritePropertyMultiple Service</b>		
	<b>Test Conditionality</b>	This test shall only be applied to devices claiming Protocol_Revision 11 or higher and which supports execution of WritePropertyMultiple.
	<b>Test Directives</b>	Apply to the Local_Time property in Device object
	<b>Testing Hints</b>	

## 3.11 Event Enrollment Object

### 3.11.1 Base Requirements

Base requirements must be met by any IUT that can contain Event Enrollment Objects. There are no base requirements tests for this section.

### 3.11.2 Supports AE-N-I-B

The IUT supports AE-N-I-B.

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

### 3.11.3 Contains an object with Reliability\_Evaluation\_Inhibit Property

The IUT contains, or can be made to contain, a Reliability\_Evaluation\_Inhibit property that is configurable to a value of TRUE.

BTL - 7.3.1.X8.1 - Reliability_Evaluation_Inhibit Test		
	Test Conditionality	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	Test Directives	
	Testing Hints	
BTL - 7.3.1.X8.2 - Reliability_Evaluation_Inhibit Summarization Test		
	Test Conditionality	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	Test Directives	
	Testing Hints	

---

## 3.12 Group Object

---

### 3.12.1 Base Requirements

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

135.1-2013 - 7.3.2.14 - Group Object Test		
	Test Conditionality	Must be executed
	Test Directives	
	Testing Hints	

### 3.13 Loop Object

#### 3.13.1 Base Requirements

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

135.1-2013 - 7.3.2.17.1 - Manipulated Variable Reference Tracking		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
135.1-2013 - 7.3.2.17.2 - Controlled Variable Reference Tracking		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

#### 3.13.2 Supports Writable Out\_Of\_Service Property

The Out\_Of\_Service property in Loop objects contained in the IUT are writable.

135.1-2013 - 7.3.1.1 - Out_Of_Service, Status_Flags, and Reliability Tests		
	Test Conditionality	If this property is writable, this test must be executed.
	Test Directives	
	Testing Hints	

#### 3.13.3 Contains a Non-Empty Setpoint\_Reference List

Controlled\_Variable\_Value tracks the property referenced by Controlled\_Variable\_Reference.

135.1-2013 - 7.3.2.17.3 - Setpoint Reference Tracking		
	Test Conditionality	If the prescribed control loop can be configured, it must be executed.
	Test Directives	
	Testing Hints	

#### 3.13.4 Contains an object with Reliability\_Evaluation\_Inhibit Property

The IUT contains, or can be made to contain, a Reliability\_Evaluation\_Inhibit property that is configurable to a value of TRUE.

BTL - 7.3.1.X8.1 - Reliability Evaluation Inhibit Test		
	Test Conditionality	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	Test Directives	
	Testing Hints	
BTL - 7.3.1.X8.2 - Reliability Evaluation Inhibit Summarization Test		
	Test Conditionality	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	Test Directives	
	Testing Hints	

## 3.14 Multi-state Input Object

### 3.14.1 Base Requirements

Base requirements must be met by any IUT that can contain Multi-state Input objects. There are no base requirements tests for this section.

### 3.14.2 Supports Writable Out\_Of\_Service Property

The Out\_Of\_Service property in multi-state objects contained in the IUT are writable.

135.1-2013 - 7.3.1.1 - Out_Of_Service, Status_Flags, and Reliability Tests		
	Test Conditionality	If this property is writable, this test must be executed.
	Test Directives	
	Testing Hints	

### 3.14.3 Supports State\_Text

The size of the State\_Text array corresponds to the Number\_Of\_States.

135.1-2013 - 7.3.2.18.2 - Number_Of_States and State_Text		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 3.14.4 Supports Resizable State\_Text Property

The IUT is protocol revision 4 or higher and the IUT contains, or can be made to contain, a State\_Text property that is resizable by writing to the array.

135.1-2013 - 7.3.2.18.5 - Number_Of_States and State_Text Size Change Test		
	Test Conditionality	Must be executed if the IUT claims Protocol_Revision 4 or greater and supports a resizable State_Text property.
	Test Directives	
	Testing Hints	

### 3.14.5 Contains an object with Reliability\_Evaluation\_Inhibit Property

The IUT contains, or can be made to contain, a Reliability\_Evaluation\_Inhibit property that is configurable to a value of TRUE.

BTL - 7.3.1.X8.1 - Reliability_Evaluation_Inhibit Test		
	Test Conditionality	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	Test Directives	
	Testing Hints	
BTL - 7.3.1.X8.2 - Reliability_Evaluation_Inhibit Summarization Test		
	Test Conditionality	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	Test Directives	
	Testing Hints	

### 3.15 Multi-state Output Object

#### 3.15.1 Base Requirements

Base requirements must be met by any IUT that can contain Multi-state Output objects. There are no base requirements tests for this section.

#### 3.15.2 Supports Command Prioritization

The objects contain a priority array and support command prioritization.

135.1-2013 - 7.3.1.2 - Relinquish Default Test		
	Test Conditionality	If no object can be made to meet the configuration requirements, this test shall be skipped.
	Test Directives	
	Testing Hints	
135.1-2013 - 7.3.1.3 - Command Prioritization Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

#### 3.15.3 Supports Writable Out\_Of\_Service Property

The Out\_Of\_Service property in multi-state objects contained in the IUT are writable.

135.1-2013 - 7.3.1.1 - Out_Of_Service, Status_Flags, and Reliability Tests		
	Test Conditionality	If this property is writable, this test must be executed.
	Test Directives	
	Testing Hints	

#### 3.15.4 Supports State\_Text

The size of the State\_Text array corresponds to the Number\_Of\_States.

135.1-2013 - 7.3.2.19.2 - Number_Of_States and State_Text		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

#### 3.15.5 Supports Resizable State\_Text Property

The IUT is protocol revision 4 or higher and the IUT contains, or can be made to contain, a State\_Text property that is resizable by writing to the array.

135.1-2013 - 7.3.2.19.6 - Number_Of_States and State_Text Size Change Test		
	Test Conditionality	Must be executed if the IUT claims Protocol_Revision 4 or greater and supports a resizable State_Text property.
	Test Directives	
	Testing Hints	

#### 3.15.6 Contains an object with Reliability\_Evaluation\_Inhibit Property

The IUT contains, or can be made to contain, a Reliability\_Evaluation\_Inhibit property that is configurable to a value of TRUE.

BTL - 7.3.1.X8.1 - Reliability_Evaluation_Inhibit Test		
	Test Conditionality	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.

# BTL Test Plan

	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.X8.2 - Reliability Evaluation Inhibit Summarization Test</b>		
	<b>Test Conditionality</b>	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

## 3.16 Multi-state Value Object

### 3.16.1 Base Requirements

Base requirements must be met by any IUT that can contain Multi-state Value objects. There are no base requirements tests for this section.

### 3.16.2 Supports Writable Out\_Of\_Service Property

The Out\_Of\_Service property in multi-state objects contained in the IUT are writable.

135.1-2013 - 7.3.1.1 - Out_Of_Service, Status_Flags, and Reliability Tests		
	Test Conditionality	If this property is writable, this test must be executed.
	Test Directives	
	Testing Hints	

### 3.16.3 Supports State\_Text

The size of the State\_Text array corresponds to the Number\_Of\_States.

135.1-2013 - 7.3.2.20.2 - Number_Of_States and State_Text		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 3.16.4 Supports Command Prioritization

The objects contain a priority array and support command prioritization.

135.1-2013 - 7.3.1.2 - Relinquish Default Test		
	Test Conditionality	If no object can be made to meet the configuration requirements, this test shall be skipped.
	Test Directives	
	Testing Hints	
135.1-2013 - 7.3.1.3 - Command Prioritization Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 3.16.5 Supports Resizable State\_Text Property

The IUT is protocol revision 4 or higher and the IUT contains, or can be made to contain, a State\_Text property that is resizable by writing to the array.

135.1-2013 - 7.3.2.20.5 - Number_Of_States and State_Text Size Change Test		
	Test Conditionality	Must be executed if the IUT claims Protocol_Revision 4 or greater and supports a resizable State_Text property.
	Test Directives	
	Testing Hints	

### 3.16.6 Contains an object with Reliability\_Evaluation\_Inhibit Property

The IUT contains, or can be made to contain, a Reliability\_Evaluation\_Inhibit property that is configurable to a value of TRUE.

BTL - 7.3.1.X8.1 - Reliability_Evaluation_Inhibit Test		
	Test Conditionality	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.



# BTL Test Plan

	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.X8.2 - Reliability Evaluation Inhibit Summarization Test</b>		
	<b>Test Conditionality</b>	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

## 3.17 Notification Class Object

### 3.17.1 Base Requirements

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

<b>BTL - 7.3.2.21.3.1 - ValidDays Test</b>		
	<b>Test Conditionality</b>	Must be executed if the IUT keeps time.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.21.3.2 - FromTime and ToTime Test</b>		
	<b>Test Conditionality</b>	Must be executed if the IUT keeps time.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.21.3.4 - Transitions Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.21.3.X7 - Recipient_List non-volatility Test</b>		
	<b>Test Conditionality</b>	Must be executed if the IUT claims Protocol_Revision 13 or greater.
	<b>Test Directives</b>	Repeat the test, via ReinitializeDevice command if supported, and with a power interruption.
	<b>Testing Hints</b>	

### 3.17.2 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.

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

### 3.17.3 Supports Writable Recipient\_List Property

The IUT supports Recipient\_List properties that are modifiable via write services. All required values must be accepted in write requests.

<b>Verify Checklist</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support for DS-WP-B and 'Contains writable list properties' in the Checklist.
	<b>Testing Hints</b>	
<b>BTL - 9.22.1.X2 - Writing to Properties Based on Data Type</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that all required values of BACnetDestination can be written (into a list element).
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.21.3.5 - Recipient_List Property Supports Device Identifier Recipients Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.21.3.6 - Recipient_List Property Supports Network Address Recipients</b>		
	<b>Test Conditionality</b>	Must be executed.

	<b>Test Directives</b>	Execute test multiple times using unicast, local broadcast, remote broadcast and global broadcast addresses. Use unicast MAC addresses of 1-6 octets.
	<b>Testing Hints</b>	
<b>BTL - 7.2.X5 - Time Non-Pattern Properties Test</b>		
	<b>Test Conditionality</b>	Must be executed if the IUT claims Protocol Revision 11 or greater.
	<b>Test Directives</b>	Apply to the fromTime and again to the toTime element in a Recipient_List property in a Notification Class.
	<b>Testing Hints</b>	
<b>BTL - 9.23.2.X10 - Time Non-Pattern Properties Test using WritePropertyMultiple Service</b>		
	<b>Test Conditionality</b>	This test shall only be applied to devices claiming Protocol_Revision 11 or higher and which supports execution of WritePropertyMultiple.
	<b>Test Directives</b>	Apply to the fromTime and again to the toTime element in a Recipient_List property in a Notification Class.
	<b>Testing Hints</b>	

### 3.17.4 Supports read-only Recipient\_List Property

The IUT supports read-only Recipient\_List properties.

<b>BTL - 7.3.2.21.3.X9 Read-only Recipient_List for external Notification Forwarder objects</b>		
	<b>Test Conditionality</b>	Must be executed if the IUT does not claim support for Notification Forwarder objects.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 3.17.5 Supports AE-CRL-B

The IUT supports AE-CRL-B.

<b>Verify Checklist</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support for AE-CRL-B.
	<b>Testing Hints</b>	

---

## **3.18 Proprietary Objects**

---

### **3.18.1 Base Requirements**

Base requirements must be met by any IUT that can contain Proprietary objects. There are no base requirements tests for this section.

### 3.19 Schedule Object

#### 3.19.1 Base Requirements

Base requirements must be met by any IUT that can contain Schedule objects. There are no base requirement tests for this section.

#### 3.19.2 Supports SCHED-I-B

The IUT supports SCHED-I-B.

Verify Checklist		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support for SCHED-I-B.
	<b>Testing Hints</b>	

#### 3.19.3 Supports SCHED-WS-I-B

The IUT supports SCHED-WS-I-B.

Verify Checklist		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support for SCHED-WS-I-B.
	<b>Testing Hints</b>	

#### 3.19.4 Supports SCHED-R-B

The IUT supports SCHED-R-B.

Verify Checklist		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support for SCHED-R-B.
	<b>Testing Hints</b>	

#### 3.19.5 Supports Resizable Exception\_Schedule Property

The IUT is protocol revision 4 or higher and the IUT contains, or can be made to contain, an Exception\_Schedule property that is resizable by writing to the array.

135.1-2013 - 7.3.2.23.9 - Exception_Schedule Size Change Test		
	<b>Test Conditionality</b>	Must be executed if the IUT claims Protocol_Revision 4 or greater and supports a resizable Exception_Schedule property.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 3.19.6 Contains an object with Reliability\_Evaluation\_Inhibit Property

The IUT contains, or can be made to contain, a Reliability\_Evaluation\_Inhibit property that is configurable to a value of TRUE.

BTL - 7.3.1.X8.1 - Reliability_Evaluation_Inhibit Test		
	<b>Test Conditionality</b>	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
BTL - 7.3.1.X8.2 - Reliability_Evaluation_Inhibit Summarization Test		

## BTL Test Plan

	<b>Test Conditionality</b>	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

## 3.20 Trend Log Object

### 3.20.1 Base Requirements

Base requirements must be met by any IUT that can contain Trend Log objects. There are no base requirements tests for this section.

### 3.20.2 Supports T-VMT-I-B

The IUT supports T-VMT-I-B.

Verify Checklist		
	Test Conditionality	Must be executed.
	Test Directives	Verify that the IUT claims support for T-VMT-I-B.
	Testing Hints	

### 3.20.3 Contains an object with Reliability\_Evaluation\_Inhibit Property

The IUT contains, or can be made to contain, a Reliability\_Evaluation\_Inhibit property that is configurable to a value of TRUE.

BTL - 7.3.1.X8.1 - Reliability_Evaluation_Inhibit Test		
	Test Conditionality	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	Test Directives	
	Testing Hints	
BTL - 7.3.1.X8.2 - Reliability_Evaluation_Inhibit Summarization Test		
	Test Conditionality	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	Test Directives	
	Testing Hints	

---

## 3.21 Structured View Object

---

### 3.21.1 Base Requirements

Base requirements must be met by any IUT that can contain Structured View objects. There are no base requirements tests for this section.

### 3.21.2 Supports writable and resizable Subordinate\_List and contains a Subordinate\_Annotations property

The IUT contains, or can be made to contain, a Subordinate\_Annotations property that is resizable.

135.1-2013 - 7.3.2.29.1 - Subordinate_List Size Changes Subordinate_Annotations		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
135.1-2013 - 7.3.2.29.2 - Subordinate_Annotations Size Changes Subordinate_List		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	



## 3.22 Event Log Object

### 3.22.1 Base Requirements

Base requirements must be met by any IUT that can contain Event Log objects. There are no base requirements tests for this section.

### 3.22.2 Supports AE-EL-I-B

The IUT supports AE-EL-I-B.

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

### 3.22.3 Supports AE-EL-E-B

The IUT supports AE-EL-E-B.

Verify Checklist		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support for AE-EL-E-B.
	<b>Testing Hints</b>	

### 3.22.4 Contains an object with Reliability\_Evaluation\_Inhibit Property

The IUT contains, or can be made to contain, a Reliability\_Evaluation\_Inhibit property that is configurable to a value of TRUE.

BTL - 7.3.1.X8.1 - Reliability_Evaluation_Inhibit Test		
	<b>Test Conditionality</b>	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
BTL - 7.3.1.X8.2 - Reliability_Evaluation_Inhibit Summarization Test		
	<b>Test Conditionality</b>	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 3.23 Trend Log Multiple Object

#### 3.23.1 Base Requirements

Base requirements must be met by any IUT that can contain Trend Log Multiple objects. There are no base requirements tests for this section.

#### 3.23.2 Supports T-VMMV-I-B

The IUT supports T-VMMV-I-B.

Verify Checklist		
	Test Conditionality	Must be executed.
	Test Directives	Verify that the IUT claims support for T-VMMV-I-B.
	Testing Hints	

#### 3.23.3 Contains an object with Reliability\_Evaluation\_Inhibit Property

The IUT contains, or can be made to contain, a Reliability\_Evaluation\_Inhibit property that is configurable to a value of TRUE.

BTL - 7.3.1.X8.1 - Reliability_Evaluation_Inhibit Test		
	Test Conditionality	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	Test Directives	
	Testing Hints	
BTL - 7.3.1.X8.2 - Reliability_Evaluation_Inhibit Summarization Test		
	Test Conditionality	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	Test Directives	
	Testing Hints	

## 3.24 Bitstring Value Object

### 3.24.1 Base Requirements

Base requirements must be met by any IUT that can contain Bitstring Value objects. There are no base requirements tests for this section.

### 3.24.2 Supports Writable Out\_Of\_Service Property

The Out\_Of\_Service property in Bitstring Value objects contained in the IUT are writable.

135.1-2013 - 7.3.1.1 - Out_Of_Service, Status_Flags, and Reliability Tests		
	Test Conditionality	If this property is writable, this test must be executed.
	Test Directives	
	Testing Hints	

### 3.24.3 Supports Command Prioritization

The objects contain a priority array and support command prioritization.

135.1-2013 - 7.3.1.2 - Relinquish Default Test		
	Test Conditionality	If no object can be made to meet the configuration requirements, this test shall be skipped.
	Test Directives	
	Testing Hints	
135.1-2013 - 7.3.1.3 - Command Prioritization Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 3.24.4 Contains an object with Reliability\_Evaluation\_Inhibit Property

The IUT contains, or can be made to contain, a Reliability\_Evaluation\_Inhibit property that is configurable to a value of TRUE.

BTL - 7.3.1.X8.1 - Reliability_Evaluation_Inhibit Test		
	Test Conditionality	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	Test Directives	
	Testing Hints	
BTL - 7.3.1.X8.2 - Reliability_Evaluation_Inhibit Summarization Test		
	Test Conditionality	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	Test Directives	
	Testing Hints	

## 3.25 CharacterString Value Object

### 3.25.1 Base Requirements

Base requirements must be met by any IUT that can contain CharacterString Value objects. There are no base requirements tests for this section.

### 3.25.2 Supports Writable Out\_Of\_Service Property

The Out\_Of\_Service property in CharacterString Value objects contained in the IUT are writable.

135.1-2013 - 7.3.1.1 - Out_Of_Service, Status_Flags, and Reliability Tests		
	Test Conditionality	If this property is writable, this test must be executed.
	Test Directives	
	Testing Hints	

### 3.25.3 Supports Command Prioritization

The objects contain a priority array and support command prioritization.

135.1-2013 - 7.3.1.2 - Relinquish Default Test		
	Test Conditionality	If no object can be made to meet the configuration requirements, this test shall be skipped.
	Test Directives	
	Testing Hints	
135.1-2013 - 7.3.1.3 - Command Prioritization Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 3.25.4 Contains an object with Reliability\_Evaluation\_Inhibit Property

The IUT contains, or can be made to contain, a Reliability\_Evaluation\_Inhibit property that is configurable to a value of TRUE.

BTL - 7.3.1.X8.1 - Reliability_Evaluation_Inhibit Test		
	Test Conditionality	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	Test Directives	
	Testing Hints	
BTL - 7.3.1.X8.2 - Reliability_Evaluation_Inhibit Summarization Test		
	Test Conditionality	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	Test Directives	
	Testing Hints	

## 3.26 Date Pattern Value Object

### 3.26.1 Base Requirements

Base requirements must be met by any IUT that can contain Date Pattern Value objects. There are no base requirements tests for this section.

### 3.26.2 Supports Writable Out\_Of\_Service Property

The Out\_Of\_Service property in Date Pattern Value objects contained in the IUT are writable.

135.1-2013 - 7.3.1.1 - Out_Of_Service, Status Flags, and Reliability Tests		
	Test Conditionality	If this property is writable, this test must be executed.
	Test Directives	
	Testing Hints	

### 3.26.3 Supports Command Prioritization

The objects contain a priority array and support command prioritization.

135.1-2013 - 7.3.1.2 - Relinquish Default Test		
	Test Conditionality	If no object can be made to meet the configuration requirements, this test shall be skipped.
	Test Directives	
	Testing Hints	
135.1-2013 - 7.3.1.3 - Command Prioritization Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 3.26.4 Contains a writable Present\_Value Property, or Can be Placed Out\_Of\_Service

The IUT contains, or can be made to contain, a Date Pattern Value Object that contains a writable Present\_Value property, or can have its Out\_Of\_Service property set to TRUE.

BTL - 7.2.X1 - Date Pattern Properties Test		
	Test Conditionality	Must be executed.
	Test Directives	Apply to the Present_Value property.
	Testing Hints	

### 3.26.5 Contains a writable Relinquish\_Default

The IUT contains, or can be made to contain, a Date Pattern Value Object that contains a writable Relinquish\_Default property.

BTL - 7.2.X1 - Date Pattern Properties Test		
	Test Conditionality	Must be executed.
	Test Directives	Apply to the Relinquish_Default property.
	Testing Hints	

### 3.26.6 Contains an object with Reliability\_Evaluation\_Inhibit Property

The IUT contains, or can be made to contain, a Reliability\_Evaluation\_Inhibit property that is configurable to a value of TRUE.

<b>BTL - 7.3.1.X8.1 - Reliability_Evaluation_Inhibit Test</b>		
	<b>Test Conditionality</b>	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.X8.2 - Reliability_Evaluation_Inhibit Summarization Test</b>		
	<b>Test Conditionality</b>	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

## 3.27 Date Value Object

### 3.27.1 Base Requirements

Base requirements must be met by any IUT that can contain Date Value objects. There are no base requirements tests for this section.

### 3.27.2 Supports Writable Out\_Of\_Service Property

The Out\_Of\_Service property in Date Value objects contained in the IUT are writable.

135.1-2013 - 7.3.1.1 - Out_Of_Service, Status_Flags, and Reliability Tests		
	Test Conditionality	If this property is writable, this test must be executed.
	Test Directives	
	Testing Hints	

### 3.27.3 Supports Command Prioritization

The object contains a priority array and support command prioritization.

135.1-2013 - 7.3.1.2 - Relinquish Default Test		
	Test Conditionality	If no object can be made to meet the configuration requirements, this test shall be skipped.
	Test Directives	
	Testing Hints	
135.1-2013 - 7.3.1.3 - Command Prioritization Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 3.27.4 Contains a Writable Present\_Value Property, or Can Be Placed Out\_Of\_Service

The IUT contains, or can be made to contain, a Date Value object that contains a writable Present\_Value property.

BTL - 7.2.X4 - Date Non-Pattern Properties Test		
	Test Conditionality	Must be executed if the IUT claims Protocol_Revision 11 or greater.
	Test Directives	Apply to the Present_Value property in a Date Value object.
	Testing Hints	
BTL - 7.3.X4 - Date 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 Present_Value property in a Date Value object.
	Testing Hints	

### 3.27.5 Contains a Writable Relinquish\_Default Property

The IUT contains, or can be made to contain, a Date Value object that contains a writable Relinquish\_Default property.

BTL - 7.2.X4 - Date Non-Pattern Properties Test		
	Test Conditionality	Must be executed if the IUT claims Protocol_Revision 11 or greater.

	<b>Test Directives</b>	Apply to the Relinquish_Default property in a Date Value object.
	<b>Testing Hints</b>	
<b>BTL - 9.23.2.X9 - Date Non-Pattern Properties Test using WritePropertyMultiple Service</b>		
	<b>Test Conditionality</b>	This test shall only be applied to devices claiming Protocol_Revision 11 or higher and which supports execution of WritePropertyMultiple.
	<b>Test Directives</b>	Apply to the Relinquish_Default property in a Date Value object.
	<b>Testing Hints</b>	

### 3.27.6 Contains an object with Reliability\_Evaluation\_Inhibit Property

The IUT contains, or can be made to contain, a Reliability\_Evaluation\_Inhibit property that is configurable to a value of TRUE.

<b>BTL - 7.3.1.X8.1 - Reliability_Evaluation_Inhibit Test</b>		
	<b>Test Conditionality</b>	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.X8.2 - Reliability_Evaluation_Inhibit Summarization Test</b>		
	<b>Test Conditionality</b>	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	



## 3.28 DateTime Pattern Value Object

### 3.28.1 Base Requirements

Base requirements must be met by any IUT that can contain DateTime Pattern Value objects. There are no base requirements tests for this section.

### 3.28.2 Supports Writable Out\_Of\_Service Property

The Out\_Of\_Service property in DateTime Pattern Value objects contained in the IUT are writable.

135.1-2013 - 7.3.1.1 - Out_Of_Service, Status_Flags, and Reliability Tests		
	Test Conditionality	If this property is writable, this test must be executed.
	Test Directives	
	Testing Hints	

### 3.28.3 Supports Command Prioritization

The objects contain a priority array and support command prioritization.

135.1-2013 - 7.3.1.2 - Relinquish Default Test		
	Test Conditionality	If no object can be made to meet the configuration requirements, this test shall be skipped.
	Test Directives	
	Testing Hints	
135.1-2013 - 7.3.1.3 - Command Prioritization Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 3.28.4 Contains a writable Present\_Value Property, or Can be Placed Out\_Of\_Service

The IUT contains, or can be made to contain, a DateTime Pattern Value Object that contains a writable Present\_Value property, or can have its Out\_Of\_Service property set to TRUE.

BTL - 7.2.X3 - DateTime Pattern Properties Test		
	Test Conditionality	Must be executed.
	Test Directives	Apply to the Present_Value property.
	Testing Hints	

### 3.28.5 Contains a writable Relinquish\_Default

The IUT contains, or can be made to contain, a DateTime Pattern Value Object that contains a writable Relinquish\_Default property.

BTL - 7.2.X3 - DateTime Pattern Properties Test		
	Test Conditionality	Must be executed.
	Test Directives	Apply to the Relinquish_Default property.
	Testing Hints	

### 3.28.6 Contains an object with Reliability\_Evaluation\_Inhibit Property

The IUT contains, or can be made to contain, a Reliability\_Evaluation\_Inhibit property that is configurable to a value of TRUE.

<b>BTL - 7.3.1.X8.1 - Reliability_Evaluation_Inhibit Test</b>		
	<b>Test Conditionality</b>	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.X8.2 - Reliability_Evaluation_Inhibit Summarization Test</b>		
	<b>Test Conditionality</b>	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

## 3.29 DateTime Value Object

### 3.29.1 Base Requirements

Base requirements must be met by any IUT that can contain DateTime Value objects. There are no base requirements tests for this section.

### 3.29.2 Supports Writable Out\_Of\_Service Property

The Out\_Of\_Service property in DateTime Value objects contained in the IUT are writable.

135.1-2013 - 7.3.1.1 - Out_Of_Service, Status_Flags, and Reliability Tests		
	Test Conditionality	If this property is writable, this test must be executed.
	Test Directives	
	Testing Hints	

### 3.29.3 Supports Command Prioritization

The objects contain a priority array and support command prioritization.

135.1-2013 - 7.3.1.2 - Relinquish Default Test		
	Test Conditionality	If no object can be made to meet the configuration requirements, this test shall be skipped.
	Test Directives	
	Testing Hints	
135.1-2013 - 7.3.1.3 - Command Prioritization Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 3.29.4 Contains a Writable Present\_Value Property, or Can be Placed Out\_Of\_Service

The IUT contains, or can be made to contain, a DateTime Value object that contains a writable Present\_Value property.

BTL - 7.2.X6 - DateTime Non-Pattern Properties Test		
	Test Conditionality	Must be executed if the IUT claims Protocol_Revision 11 or greater.
	Test Directives	Apply to the Present_Value property in a DateTime Value object.
	Testing Hints	
BTL - 9.23.2.X11 - DateTime 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 Present_Value property in a DateTime Value object.
	Testing Hints	

### 3.29.5 Contains a Writable Relinquish\_Default Property

The IUT contains, or can be made to contain, a DateTime Value object that contains a writable Relinquish\_Default property.

BTL - 7.2.X6 - DateTime Non-Pattern Properties Test		
	Test Conditionality	Must be executed if the IUT claims Protocol_Revision 11 or greater.
	Test Directives	Apply to the Relinquish_Default property in a DateTime Value object.

	<b>Testing Hints</b>	
<b>BTL - 9.23.2.X11 - DateTime Non-Pattern Properties Test using WritePropertyMultiple Service</b>		
	<b>Test Conditionality</b>	This test shall only be applied to devices claiming Protocol_Revision 11 or higher and which supports execution of WritePropertyMultiple.
	<b>Test Directives</b>	Apply to the Relinquish_Default property in a DateTime Value object.
	<b>Testing Hints</b>	

### 3.29.6 Contains an object with Reliability\_Evaluation\_Inhibit Property

The IUT contains, or can be made to contain, a Reliability\_Evaluation\_Inhibit property that is configurable to a value of TRUE.

<b>BTL - 7.3.1.X8.1 - Reliability Evaluation_Inhibit Test</b>		
	<b>Test Conditionality</b>	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.X8.2 - Reliability Evaluation_Inhibit Summarization Test</b>		
	<b>Test Conditionality</b>	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 3.30 Integer Value Object

#### 3.30.1 Base Requirements

Base requirements must be met by any IUT that can contain Integer Value objects. There are no base requirements tests for this section.

#### 3.30.2 Supports Writable Out\_Of\_Service Property

The Out\_Of\_Service property in Integer Value objects contained in the IUT are writable.

135.1-2013 - 7.3.1.1 - Out_Of_Service, Status_Flags, and Reliability Tests		
	Test Conditionality	If this property is writable, this test must be executed.
	Test Directives	
	Testing Hints	

#### 3.30.3 Supports Command Prioritization

The objects contain a priority array and support command prioritization.

135.1-2013 - 7.3.1.2 - Relinquish Default Test		
	Test Conditionality	If no object can be made to meet the configuration requirements, this test shall be skipped.
	Test Directives	
	Testing Hints	
135.1-2013 - 7.3.1.3 - Command Prioritization Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

#### 3.30.4 Contains an object with Reliability\_Evaluation\_Inhibit Property

The IUT contains, or can be made to contain, a Reliability\_Evaluation\_Inhibit property that is configurable to a value of TRUE.

BTL - 7.3.1.X8.1 - Reliability_Evaluation_Inhibit Test		
	Test Conditionality	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	Test Directives	
	Testing Hints	
BTL - 7.3.1.X8.2 - Reliability_Evaluation_Inhibit Summarization Test		
	Test Conditionality	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	Test Directives	
	Testing Hints	

### 3.31 Large Analog Value Object

#### 3.31.1 Base Requirements

Base requirements must be met by any IUT that can contain Large Analog Value objects. There are no base requirements tests for this section.

#### 3.31.2 Supports Writable Out\_Of\_Service Property

The Out\_Of\_Service property in Large Analog Value objects contained in the IUT are writable.

135.1-2013 - 7.3.1.1 - Out_Of_Service, Status_Flags, and Reliability Tests		
	Test Conditionality	If this property is writable, this test must be executed.
	Test Directives	
	Testing Hints	

#### 3.31.3 Supports Command Prioritization

The objects contain a priority array and support command prioritization.

135.1-2013 - 7.3.1.2 - Relinquish Default Test		
	Test Conditionality	If no object can be made to meet the configuration requirements, this test shall be skipped.
	Test Directives	
	Testing Hints	
135.1-2013 - 7.3.1.3 - Command Prioritization Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

#### 3.31.4 Contains an object with Reliability\_Evaluation\_Inhibit Property

The IUT contains, or can be made to contain, a Reliability\_Evaluation\_Inhibit property that is configurable to a value of TRUE.

BTL - 7.3.1.X8.1 - Reliability_Evaluation_Inhibit Test		
	Test Conditionality	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	Test Directives	
	Testing Hints	
BTL - 7.3.1.X8.2 - Reliability_Evaluation_Inhibit Summarization Test		
	Test Conditionality	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	Test Directives	
	Testing Hints	

## 3.32 OctetString Value Object

### 3.32.1 Base Requirements

Base requirements must be met by any IUT that can contain OctetString Value objects. There are no base requirements tests for this section.

### 3.32.2 Supports Writable Out\_Of\_Service Property

The Out\_Of\_Service property in OctetString Value objects contained in the IUT are writable.

135.1-2013 - 7.3.1.1 - Out_Of_Service, Status_Flags, and Reliability Tests		
	Test Conditionality	If this property is writable, this test must be executed.
	Test Directives	
	Testing Hints	

### 3.32.3 Supports Command Prioritization

The objects contain a priority array and support command prioritization.

135.1-2013 - 7.3.1.2 - Relinquish Default Test		
	Test Conditionality	If no object can be made to meet the configuration requirements, this test shall be skipped.
	Test Directives	
	Testing Hints	
135.1-2013 - 7.3.1.3 - Command Prioritization Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 3.32.4 Contains an object with Reliability\_Evaluation\_Inhibit Property

The IUT contains, or can be made to contain, a Reliability\_Evaluation\_Inhibit property that is configurable to a value of TRUE.

BTL - 7.3.1.X8.1 - Reliability_Evaluation_Inhibit Test		
	Test Conditionality	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	Test Directives	
	Testing Hints	
BTL - 7.3.1.X8.2 - Reliability_Evaluation_Inhibit Summarization Test		
	Test Conditionality	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	Test Directives	
	Testing Hints	

### 3.33 Positive Integer Value Object

#### 3.33.1 Base Requirements

Base requirements must be met by any IUT that can contain Positive Integer Value objects. There are no base requirements tests for this section.

#### 3.33.2 Supports Writable Out\_Of\_Service Property

The Out\_Of\_Service property in Positive Integer Value objects contained in the IUT are writable.

135.1-2013 - 7.3.1.1 - Out_Of_Service, Status_Flags, and Reliability Tests		
	Test Conditionality	If this property is writable, this test must be executed.
	Test Directives	
	Testing Hints	

#### 3.33.3 Supports Command Prioritization

The objects contain a priority array and support command prioritization.

135.1-2013 - 7.3.1.2 - Relinquish Default Test		
	Test Conditionality	If no object can be made to meet the configuration requirements, this test shall be skipped.
	Test Directives	
	Testing Hints	
135.1-2013 - 7.3.1.3 - Command Prioritization Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

#### 3.33.4 Contains an object with Reliability\_Evaluation\_Inhibit Property

The IUT contains, or can be made to contain, a Reliability\_Evaluation\_Inhibit property that is configurable to a value of TRUE.

BTL - 7.3.1.X8.1 - Reliability_Evaluation_Inhibit Test		
	Test Conditionality	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	Test Directives	
	Testing Hints	
BTL - 7.3.1.X8.2 - Reliability_Evaluation_Inhibit Summarization Test		
	Test Conditionality	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	Test Directives	
	Testing Hints	



### 3.34 Time Pattern Value Object

#### 3.34.1 Base Requirements

Base requirements must be met by any IUT that can contain Time Pattern Value objects. There are no base requirements tests for this section.

#### 3.34.2 Supports Writable Out\_Of\_Service Property

The Out\_Of\_Service property in Time Pattern Value objects contained in the IUT are writable.

135.1-2013 - 7.3.1.1 - Out_Of_Service, Status_Flags, and Reliability Tests		
	Test Conditionality	If this property is writable, this test must be executed.
	Test Directives	
	Testing Hints	

#### 3.34.3 Supports Command Prioritization

The objects contain a priority array and support command prioritization.

135.1-2013 - 7.3.1.2 - Relinquish Default Test		
	Test Conditionality	If no object can be made to meet the configuration requirements, this test shall be skipped.
	Test Directives	
	Testing Hints	
135.1-2013 - 7.3.1.3 - Command Prioritization Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

#### 3.34.4 Contains a writable Present\_Value Property, or Can be Placed Out\_Of\_Service

The IUT contains, or can be made to contain, a Time Pattern Value Object that contains a writable Present\_Value property, or can have its Out\_Of\_Service property set to TRUE.

BTL - 7.2.X2 - Time Pattern Properties Test		
	Configuration	If Present_Value is not writable, then Out_Of_Service shall be set to TRUE.
	Test Conditionality	Must be executed.
	Test Directives	Apply to the Present_Value property.
	Testing Hints	

#### 3.34.5 Contains a writable Relinquish\_Default

The IUT contains, or can be made to contain, a Time Pattern Value Object that contains a writable Relinquish\_Default property.

BTL - 7.2.X2 - Time Pattern Properties Test		
	Test Conditionality	Must be executed.
	Test Directives	Apply to the Relinquish_Default property.
	Testing Hints	

### 3.34.6 Contains an object with Reliability\_Evaluation\_Inhibit Property

The IUT contains, or can be made to contain, a Reliability\_Evaluation\_Inhibit property that is configurable to a value of TRUE.

<b>BTL - 7.3.1.X8.1 - Reliability_Evaluation_Inhibit Test</b>		
	<b>Test Conditionality</b>	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.X8.2 - Reliability_Evaluation_Inhibit Summarization Test</b>		
	<b>Test Conditionality</b>	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 3.35 Time Value Object

#### 3.35.1 Base Requirements

Base requirements must be met by any IUT that can contain Time Value objects. There are no base requirements tests for this section.

#### 3.35.2 Supports Writable Out\_Of\_Service Property

The Out\_Of\_Service property in Time Value objects contained in the IUT are writable.

135.1-2013 - 7.3.1.1 - Out_Of_Service, Status_Flags, and Reliability Tests		
	<b>Test Conditionality</b>	If this property is writable, this test must be executed.
	<b>Test Directives</b>	The test shall be executed using a Time Value object
	<b>Testing Hints</b>	

#### 3.35.3 Supports Command Prioritization

The objects contain a priority array and support command prioritization.

135.1-2013 - 7.3.1.2 - Relinquish Default Test		
	<b>Test Conditionality</b>	If no object can be made to meet the configuration requirements, this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
135.1-2013 - 7.3.1.3 - Command Prioritization Test		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 3.35.4 Contains a Writable Present\_Value Property

The IUT contains, or can be made to contain, a Time Value object that contains a writable Present\_Value property.

BTL - 7.2.X5 - Time Non-Pattern Properties Test		
	<b>Test Conditionality</b>	Must be executed if the IUT claims Protocol_Revision 11 or greater.
	<b>Test Directives</b>	Apply to the Present_Value property in a Time Value object.
	<b>Testing Hints</b>	
BTL - 9.23.2.X10 - Time Non-Pattern Properties Test using WritePropertyMultiple Service		
	<b>Test Conditionality</b>	This test shall only be applied to devices claiming Protocol_Revision 11 or higher and which supports execution of WritePropertyMultiple.
	<b>Test Directives</b>	Apply to the Present_Value property in a Time Value object.
	<b>Testing Hints</b>	

#### 3.35.5 Contains a Writable Relinquish\_Default Property

The IUT contains, or can be made to contain, a Time Value object that contains a writable Relinquish\_Default property.

BTL - 7.2.X5 - Time Non-Pattern Properties Test		
	<b>Test Conditionality</b>	Must be executed if the IUT claim Protocol_Revision 11 or greater.
	<b>Test Directives</b>	Apply to the Relinquish_Default property in a Time Value object.

	<b>Testing Hints</b>	
<b>BTL - 9.23.2.X10 - Time Non-Pattern Properties Test using WritePropertyMultiple Service</b>		
	<b>Test Conditionality</b>	This test shall only be applied to devices claiming Protocol_Revision 11 or higher and which supports execution of WritePropertyMultiple.
	<b>Test Directives</b>	Apply to the Relinquish_Default property in a Time Value object.
	<b>Testing Hints</b>	

### 3.35.6 Contains an object with Reliability\_Evaluation\_Inhibit Property

The IUT contains, or can be made to contain, a Reliability\_Evaluation\_Inhibit property that is configurable to a value of TRUE.

<b>BTL - 7.3.1.X8.1 - Reliability_Evaluation_Inhibit Test</b>		
	<b>Test Conditionality</b>	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.X8.2 - Reliability_Evaluation_Inhibit Summarization Test</b>		
	<b>Test Conditionality</b>	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 3.36 Global Group Object

#### 3.36.1 Base Requirements

Base requirements must be met by any IUT that can contain Global Group objects.

<b>BTL - 7.2.3 - Read-only Property Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Test the Present_Value property of each Global Group object.
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.13.X2 - Reliability MEMBER_FAULT Test</b>		
	<b>Test Conditionality</b>	If no object pointed to by the Group_Members property can be made to contain Status_Flags FAULT flag equal to TRUE, then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.13.X3 - Reliability COMMUNICATION_FAILURE Test</b>		
	<b>Test Conditionality</b>	If the Groups_Members property is not writable and can not be made to contain references to external objects, this test shall be skipped.
	<b>Test Directives</b>	Repeat this test with a member of the Group_Members property pointing to a device not communicating and with a member of the Group_Members property pointing to a device responding with a BACnet-Error-PDU.
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.13.X4 - Present_Value Tracking and Reliability Test</b>		
	<b>Test Conditionality</b>	If the Reliability property is not present or can not be made to not equal NO_FAULT_DETECTED, this test shall be skipped.
	<b>Test Directives</b>	The test shall be executed using a Global Group object.
	<b>Testing Hints</b>	

#### 3.36.2 Supports Writable Out\_Of\_Service Property

The Out\_Of\_Service property in the Global Group object contained in the IUT are writable.

<b>BTL - 7.3.2.13.X1 - Global Group Present_Value, Out_Of_Service and Status_Flags Test</b>		
	<b>Test Conditionality</b>	If Out_Of_Service can be made TRUE, this test must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 3.36.3 Supports Writable Group\_Members Properties

The property in the Global Group object contained in the IUT is writable.

<b>135.1-2013 - 7.3.2.13.1 - Resizing Group_Member_Names by Writing Group_Members Property Test</b>		
	<b>Test Conditionality</b>	If this property is writable, this test must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.13.2 - Resizing Group_Members by Writing Group_Member_Names Property Test</b>		
	<b>Test Conditionality</b>	If this property is writable, this test must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 3.36.4 Supports DS-RP-A for retrieving member values

The IUT supports DS-RP-A.

Verify Checklist		
	Test Conditionality	Must be executed.
	Test Directives	Verify that the IUT claims support for DS-RP-A.
	Testing Hints	

### 3.36.5 Is able to read REAL values

The IUT supports a Global Group object that is able to read REAL values.

BTL - 7.3.2.13.X5 - Present Value Tracking Test		
	Test Conditionality	Must be executed.
	Test Directives	Set the Group_Members property to make the Present_Value read a REAL value.
	Testing Hints	

### 3.36.6 Is able to read Unsigned values

The IUT supports a Global Group object that is able to read Unsigned values.

BTL - 7.3.2.13.X5 - Present Value Tracking Test		
	Test Conditionality	Must be executed.
	Test Directives	Set the Group_Members property to make the Present_Value read an UNSIGNED value.
	Testing Hints	

### 3.36.7 Is able to read INTEGER (Signed) values

The IUT supports a Global Group object that is able to read INTEGER values.

BTL - 7.3.2.13.X5 - Present Value Tracking Test		
	Test Conditionality	Must be executed.
	Test Directives	Set the Group_Members property to make the Present_Value read an INTEGER (Signed) value.
	Testing Hints	

### 3.36.8 Is able to read BOOLEAN values

The IUT supports a Global Group object that is able to read BOOLEAN values.

BTL - 7.3.2.13.X5 - Present Value Tracking Test		
	Test Conditionality	Must be executed.
	Test Directives	Set the Group_Members property to make the Present_Value read a BOOLEAN value.
	Testing Hints	

### 3.36.9 Is able to read Bit String values

The IUT supports a Global Group object that is able to read Bit String values.

BTL - 7.3.2.13.X5 - Present Value Tracking Test		
	Test Conditionality	Must be executed.
	Test Directives	Set the Group_Members property to make the Present_Value read a Bit String value.
	Testing Hints	

### 3.36.10 Is able to read Enumerated values

The IUT supports a Global Group object that is able to read Enumerated values.

BTL - 7.3.2.13.X5 - Present Value Tracking Test		
	Test Conditionality	Must be executed.
	Test Directives	Set the Group_Members property to make the Present_Value read an Enumerated value.
	Testing Hints	

### 3.36.11 Is able to read NULL values

The IUT supports a Global Group object that is able to read NULL values.

BTL - 7.3.2.13.X5 - Present Value Tracking Test		
	Test Conditionality	Must be executed.
	Test Directives	Set the Group_Members property to make the Present_Value read a NULL value.
	Testing Hints	

### 3.36.12 Is able to read Double values

The IUT supports a Global Group object that is able to read Double values.

BTL - 7.3.2.13.X5 - Present Value Tracking Test		
	Test Conditionality	Must be executed.
	Test Directives	Set the Group_Members property to make the Present_Value read a Double value.
	Testing Hints	

### 3.36.13 Is able to read Character String values

The IUT supports a Global Group object that is able to read Character String values.

BTL - 7.3.2.13.X5 - Present Value Tracking Test		
	Test Conditionality	Must be executed.
	Test Directives	Set the Group_Members property to make the Present_Value read a Character String value.
	Testing Hints	

### 3.36.14 Is able to read Octet String values

The IUT supports a Global Group object that is able to read Octet String values.

BTL - 7.3.2.13.X5 - Present Value Tracking Test		
	Test Conditionality	Must be executed.
	Test Directives	Set the Group_Members property to make the Present_Value read an Octet String value.
	Testing Hints	

### 3.36.15 Is able to read Date values

The IUT supports a Global Group object that is able to read Date values.

BTL - 7.3.2.13.X5 - Present Value Tracking Test		
	Test Conditionality	Must be executed.

<b>Test Directives</b>	Set the Group_Members property to make the Present_Value read a Date value.
<b>Testing Hints</b>	

### 3.36.16 Is able to read Time values

The IUT supports a Global Group object that is able to read Time values.

BTL - 7.3.2.13.X5 - Present Value Tracking Test		
	Test Conditionality	Must be executed.
	Test Directives	Set the Group_Members property to make the Present_Value read a Time value.
	Testing Hints	

### 3.36.17 Is able to read BACnetObjectIdentifier values

The IUT supports a Global Group object that is able to read BACnetObjectIdentifier values.

BTL - 7.3.2.13.X5 - Present Value Tracking Test		
	Test Conditionality	Must be executed.
	Test Directives	Set the Group_Members property to make the Present_Value read a BACnetObjectIdentifier value.
	Testing Hints	

### 3.36.18 Supports COV to retrieve remote property values

The IUT can be made to use COV to retrieve values for the Global Group object.

Requires that COV\_Resubscription\_Interval be present.

BTL - 7.3.1.7.X1 - COV Resubscription Interval Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 3.36.19 Supports COVU\_Period and COVU\_Recipients properties

The IUT contains, or can be made to contain, a Global Group object that can generate UnconfirmedCOVNotifications using the COVU\_Period and COVU\_Recipients properties.

Verify Checklist		
	Test Conditionality	Must be executed.
	Test Directives	Verify that the IUT claims support for DS-COVU-B in the Checklist.
	Testing Hints	
Verify Checklist		
	Test Conditionality	Must be executed.
	Test Directives	Verify that the IUT claims support for DM-DDB-A in the Checklist.
	Testing Hints	
BTL - 9.22.1.X2 - Writing to Properties Based on Data Type		
	Test Conditionality	Must be executed.
	Test Directives	The property that is written for this instance of this test shall be the COVU_Recipients property.
	Testing Hints	Verify that any legal value of BACnetRecipient (as a list element) can be written.
BTL - 9.22.1.X2 - Writing to Properties Based on Data Type		
	Test Conditionality	Must be executed.



	<b>Test Directives</b>	The property that is written for this instance of this test shall be the COVU Period property.
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.13.X6 - COVU Period and COVU Recipients Zero Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 8.3.X1 - COVU Recipients Notifications</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 3.36.20 Supports Event Reporting

The IUT supports or can be configured to support, event reporting in the Global Group Object.

<b>Verify Checklist</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support for AE-N-I-B or AE-N-E-B in the Checklist with option "Implements the CHANGE_OF_RELIABILITY - FAULT_STATUS_FLAGS Algorithm".
	<b>Testing Hints</b>	
<b>BTL - 8.5.X9.11 - CHANGE OF RELIABILITY with First Stage Object Fault</b>		
	<b>Test Conditionality</b>	This test shall be executed if the object's Reliability property can be made to equal COMMUNICATION_FAILURE otherwise this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 3.37 Accumulator Object

#### 3.37.1 Base Requirements

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

BTL - 7.3.2.X37.1.1 - Present Value Remains In-Range Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
BTL - 7.2.X6 - Datetime Non-Pattern Properties Test		
	Test Conditionality	Must be executed if the IUT claims Protocol_Revision 11 or greater, if the Value_Change_Time property is writable.
	Test Directives	Apply to the Value_Change_Time property, if writable.
	Testing Hints	

#### 3.37.2 Supports Prescale Property

The Prescale property in at least one Accumulator object is present.

BTL - 7.3.2.X37.1.2 - Prescale in Accumulator Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

#### 3.37.3 Supports Logging\_Record Property

The Logging\_Record property in at least one Accumulator object is present.

BTL - 7.3.2.X37.1.3 - Logging_Record in Accumulator Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
BTL - 7.3.2.X37.1.4 - Logging_Record in Accumulator RECOVERED Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
BTL - 7.3.2.X37.1.5 - Logging_Record in Accumulator STARTING Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

#### 3.37.4 Supports Writable Out\_Of\_Service Property

The Out\_Of\_Service property in Accumulator objects is writable.

BTL - 7.3.2.X37.1.6 - Out Of Service in Accumulator Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

#### 3.37.5 Supports Writable Value\_Set Property

The Value\_Set property in Accumulator objects is writable.

BTL - 7.3.2.X37.1.7 - Value_Set Writing Test		
	Test Conditionality	Must be executed.

	<b>Test Directives</b>	Verify in each object where test 7.3.2.X37.1.7 could be executed, that Value_Before_Change in that object is read-only.
	<b>Testing Hints</b>	

### 3.37.6 Supports Writable Value\_Before\_Change Property

The Value\_Before\_Change property in Accumulator objects is writable.

BTL - 7.3.2.X37.1.8 - Value_Before_Change Writing Test		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify in each object where test 7.3.2.X37.1.8 could be executed, that Value_Set in that object is read-only.
	<b>Testing Hints</b>	

---

## 3.38 Program Object

---

### 3.38.1 Base Requirements

All BACnet devices must meet these base requirements.

BTL - 7.3.2.22.1 - Program_Change property test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	Test only Program_Change values RUN and HALT.

### 3.39 Life Safety Point Object

#### 3.39.1 Base Requirements

Base requirements must be met by any IUT that can contain Life Safety Point objects

<b>BTL - 7.3.2.15.X6 - Supports Writable Mode Property</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	The test shall be executed using a Life Safety Point object.
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.15.X5 - Support Writable Tracking Value</b>		
	<b>Test Conditionality</b>	If Out Of Service can be made TRUE, this test must be executed.
	<b>Test Directives</b>	The test shall be executed using a Life Safety Point object.
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.15.X9 - Silenced Property Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	The test shall be executed using a Life Safety Point object.
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.15.X7 - Support Operation Expected Property</b>		
	<b>Test Conditionality</b>	If IUT is capable of generating event notifications then, it Must be executed.
	<b>Test Directives</b>	The test shall be executed using a Life Safety Point object.
	<b>Testing Hints</b>	

#### 3.39.2 Supports Writable Out\_Of\_Service Property

The Out\_Of\_Service property in Life Safety objects contained in the IUT are writable.

<b>135.1-2013 - 7.3.1.1 - Out Of Service, Status Flags, and Reliability Tests</b>		
	<b>Test Conditionality</b>	If Out Of Service can be made TRUE, this test must be executed.
	<b>Test Directives</b>	The test shall be executed using a Life Safety Point object.
	<b>Testing Hints</b>	

#### 3.39.3 Supports Writable Member\_Of Property

<b>BTL - 7.3.2.15.X8 - Support Writable Member_Of Property</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 3.39.4 Contains an Object with Reliability\_Evaluation\_Inhibit Property

The IUT contains, or can be made to contain, a Reliability\_Evaluation\_Inhibit property that is configurable to a value of TRUE.

<b>BTL - 7.3.1.X8.1 - Reliability Evaluation Inhibit Test</b>		
	<b>Test Conditionality</b>	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.X8.2 - Reliability Evaluation Inhibit Summarization Test</b>		
	<b>Test Conditionality</b>	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	



### 3.40 Life Safety Zone Object

#### 3.40.1 Base Requirements

Base requirements must be met by any IUT that can contain Life Safety Zone objects

<b>BTL - 7.3.2.15.X6 - Supports Writable Mode Property</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	The test shall be executed using a Life Safety Zone object.
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.15.X5 - Support Writable Tracking Value</b>		
	<b>Test Conditionality</b>	If Out Of Service can be made TRUE, this test must be executed.
	<b>Test Directives</b>	The test shall be executed using a Life Safety Zone object.
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.15.X9 - Silenced Property Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	The test shall be executed using a Life Safety Zone object.
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.15.X7 - Support Operation Expected Property</b>		
	<b>Test Conditionality</b>	If IUT is capable of generating event notifications then, it Must be executed.
	<b>Test Directives</b>	The test shall be executed using a Life Safety Zone object.
	<b>Testing Hints</b>	

#### 3.40.2 Supports Writable Out\_Of\_Service Property

The Out\_Of\_Service property in Life Safety objects contained in the IUT are writable.

<b>135.1-2013 - 7.3.1.1 - Out Of Service, Status Flags, and Reliability Tests</b>		
	<b>Test Conditionality</b>	If Out Of Service can be made TRUE, this test must be executed.
	<b>Test Directives</b>	The test shall be executed using a Life Safety Zone object.
	<b>Testing Hints</b>	

#### 3.40.3 Supports Writable Member\_Of Property

<b>BTL - 7.3.2.15.X8 - Support Writable Member_Of Property</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 3.40.4 Contains an Object with Reliability\_Evaluation\_Inhibit Property

The IUT contains, or can be made to contain, a Reliability\_Evaluation\_Inhibit property that is configurable to a value of TRUE.

<b>BTL - 7.3.1.X8.1 - Reliability Evaluation Inhibit Test</b>		
	<b>Test Conditionality</b>	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.X8.2 - Reliability Evaluation Inhibit Summarization Test</b>		
	<b>Test Conditionality</b>	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	





## 3.41 Pulse Converter Object

### 3.41.1 Base Requirements

Base requirements must be met by any IUT that can contain Pulse Converter objects.

<b>BTL - 7.3.2.X38.1.1 - Adjust Value Write Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.X38.1.2 - Scale Factor Pulse Converter Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.X38.1.5 - Update Time Reflects Change to the Count and is Updated Atomically Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.X38.2.1 - Adjust Value Out-of-Range WriteProperty Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify in the EPICS that Value_Before_Change in the object is read-only.
	<b>Testing Hints</b>	
<b>Verify EPICS</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify in the EPICS that Update_Time and Count_Change_Time in the object are read-only.
	<b>Testing Hints</b>	

### 3.41.2 Supports Writable Out\_Of\_Service Property

The Out\_Of\_Service property in Pulse Converter objects is writable.

<b>BTL - 7.3.2.X38.1.3 - Out_Of_Service Pulse Converter Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

## 3.42 Access Door Object

### 3.42.1 Base Requirements

Base requirements must be met by any IUT that supports Access Door objects

BTL - 7.3.2.X55.1.X1 - Commandable Present_Value Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 3.42.2 Supports Command Prioritization

135.1-2013 - 7.3.1.2 - Relinquish Default Test		
	Test Conditionality	If no object can be made to meet the configuration requirements, this test shall be skipped.
	Test Directives	
	Testing Hints	
135.1-2013 - 7.3.1.3 - Command Prioritization Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 3.42.3 Supports Writable Out\_Of\_Service Property

The IUT contains or can be made to contain writable Out\_Of\_Service property.

135.1-2013 - 7.3.1.1 - Out Of Service, Status Flags, and Reliability Tests		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
BTL - 7.3.2.X55.1.X2 - Door_Status, Lock_Status and Door_Alarm_State Tests		
	Test Conditionality	If neither Door_Status, Lock_Status nor Door_Alarm_State is supported, this test shall be skipped.
	Test Directives	
	Testing Hints	

### 3.42.4 Supports Door\_Status Property

The IUT contains or can be made to contain Door\_Status property which is writable when Out\_Of\_Service is True.

BTL - 7.3.2.X55.1.X3 - Door_Status with Physical Door_Status Tests		
	Test Conditionality	If the Door_Status property is permanently configured to have the value UNUSED then this test shall be skipped.
	Test Directives	
	Testing Hints	

### 3.42.5 Supports Lock\_Status Property

The IUT contains or can be made to contain Lock\_Status property which is writable when Out\_Of\_Service is True.

BTL - 7.3.2.X55.1.X4 - Lock_Status Tests		
	<b>Test Conditionality</b>	If the physical lock cannot be manipulated without writing to Present_Value of the associated Access Door objet then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 3.42.6 Supports Secured\_Status Property

The IUT contains or can be made to contain Secured\_Status property.

BTL - 7.3.2.X55.1.X5 - Secured_Status Tests		
	<b>Test Conditionality</b>	If the Secured_Status property is permanently configured to have the value UNKNOWN then this test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 3.42.7 Supports Door\_Unlock\_Delay\_Time Property

The IUT contains or can be made to contain a writable or read-only Door\_Unlock\_Delay\_Time property

BTL - 7.3.2.X55.1.X6 - Door_Unlock_Delay_Time Test		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 3.42.8 Supports Masked\_Alarm\_Values Property

The IUT contains or can be made to contain Masked\_Alarm\_Value property.

BTL - 7.3.2.X55.1.X7 - Masked_Alarm_Values Test		
	<b>Test Conditionality</b>	If Out_Of_Service is not writeable and cannot be set to TRUE by any other means, this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 3.42.9 Supports Intrinsic Reporting

The IUT supports intrinsic reporting.

BTL - 7.3.2.X55.1.X8 - Door_Open_Too_Long Test		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 3.42.10 Contains an Object with Reliability\_Evaluation\_Inhibit Property

The IUT contains or can be made to contain a Reliability\_Evaluation\_Inhibit property that is configurable to a value of TRUE.

<b>BTL - 7.3.1.X8.1 - Reliability_Evaluation_Inhibit Test</b>		
	<b>Test Conditionality</b>	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.X8.2 - Reliability_Evaluation_Inhibit Summarization Test</b>		
	<b>Test Conditionality</b>	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 3.43 Load Control Object

#### 3.43.1 Base Requirements

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

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

#### 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	

### 3.44 Access Point Object

#### 3.44.1 Base Requirements

Base requirements must be met by any IUT that can contain Access Point objects

<b>BTL - 7.3.2.X56.1 - Authentication_Status and Access_Event Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.X56.2 - Allowed Access Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.X56.3 - Denied Access Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.X56.4 - Authorization Mode Test</b>		
	<b>Test Conditionality</b>	If the IUT only supports the authorization mode AUTHORIZE, this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.X56.5 - Access Rights Exemptions Test</b>		
	<b>Test Conditionality</b>	If the Authorization_Exemption property is not supported or the ACCESS_RIGHTS exemption is not supported this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.X56.10 - Denied Access Disabled Credential Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 3.44.2 Supports Writable Out\_Of\_Service Property

The Out\_Of\_Service property in Access Point objects contained in the IUT are writable.

<b>135.1-2013 - 7.3.1.1 - Out_Of_Service, Status Flags, and Reliability Tests</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	The test shall be executed using an Access Point object
	<b>Testing Hints</b>	

#### 3.44.3 Supports Writable Active\_Authentication\_Policy Property

IUT supports Access Point objects with writable Active\_Authentication\_Policy property.

<b>BTL - 7.3.2.X56.6 - Change Authentication Policy Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 3.44.4 Supports Writable Lockout Property

IUT supports Access Point objects with writable Lockout Property.

<b>BTL - 7.3.2.X56.7 - Lockout State Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 3.44.5 Supports Writable Threat\_Level Property

IUT supports Access Point objects with writable Threat\_Level property.

<b>BTL - 7.3.2.X56.8 - Threat Level Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 3.44.6 Supports Occupancy\_Upper\_Limit\_Enforced and Occupancy\_Count\_Adjust Properties

IUT supports Access Point objects with modifiable Occupancy\_Upper\_Limit\_Enforced and Occupancy\_Count\_Adjust properties.

<b>BTL - 7.3.2.X56.9 - Denied Access Occupancy Upper Limit Test</b>		
	<b>Test Conditionality</b>	If the IUT does not support the Access Zone object type, this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 3.44.7 Contains an Object with Reliability\_Evaluation\_Inhibit Property

The IUT contains, or can be made to contain, a Reliability\_Evaluation\_Inhibit property that is configurable to a value of TRUE.

<b>BTL - 7.3.1.X8.1 - Reliability Evaluation Inhibit Test</b>		
	<b>Test Conditionality</b>	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.X8.2 - Reliability Evaluation Inhibit Summarization Test</b>		
	<b>Test Conditionality</b>	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 3.45 Access Zone Object

#### 3.45.1 Base Requirements

There are no base requirements tests for this section.

#### 3.45.2 Supports Writable Out\_Of\_Service Property

The Out\_Of\_Service property in Access Zone objects contained in the IUT are writable.

135.1-2013 - 7.3.1.1 - Out Of Service, Status Flags, and Reliability Tests		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	The test shall be executed using an Access Zone object
	<b>Testing Hints</b>	

#### 3.45.3 Supports Occupancy\_Count Property

The IUT supports Occupancy\_Count property

BTL - 7.3.2.X57.1 - Occupancy State Test		
	<b>Test Conditionality</b>	If the Occupancy_Lower_Limit is not supported then the steps of the test which test this functionality shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
BTL - 7.3.2.X57.2 - Occupancy Counting Test		
	<b>Test Conditionality</b>	If the Occupancy_Lower_Limit is not supported then the steps of the test which test this functionality shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 3.45.4 Supports Credentials\_In\_Zone Property

The IUT supports Credentials\_In\_Zone property.

BTL - 7.3.2.X57.3 - Keeping Track of Credentials Test		
	<b>Test Conditionality</b>	
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 3.45.5 Supports Passback\_Mode Property

The IUT supports Passback\_Mode property.

BTL - 7.3.2.X57.4 - Passback Mode Test		
	<b>Test Conditionality</b>	If the IUT does not support soft passback or hard passback mode then those tests shall be skipped. If Passback_Timeout is not supported then those steps shall be skipped. If the passback exemption of the Access Credential object type is not supported then those steps shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	



### 3.45.6 Contains an object with Reliability\_Evaluation\_Inhibit Property

The IUT contains, or can be made to contain, a Reliability\_Evaluation\_Inhibit property that is configurable to a value of TRUE.

<b>BTL - 7.3.1.X8.1 - Reliability_Evaluation_Inhibit Test</b>		
	<b>Test Conditionality</b>	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.X8.2 - Reliability_Evaluation_Inhibit Summarization Test</b>		
	<b>Test Conditionality</b>	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

---

## 3.46 Access User Object

---

### 3.46.1 Base Requirements

There are no base requirements tests for this section.

### 3.46.2 Contains an Object with Reliability\_Evaluation\_Inhibit Property

The IUT contains, or can be made to contain, a Reliability\_Evaluation\_Inhibit property that is configurable to a value of TRUE.

BTL - 7.3.1.X8.1 - Reliability_Evaluation_Inhibit Test		
	Test Conditionality	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	Test Directives	
	Testing Hints	
BTL - 7.3.1.X8.2 - Reliability_Evaluation_Inhibit Summarization Test		
	Test Conditionality	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	Test Directives	
	Testing Hints	

## 3.47 Access Rights Object

### 3.47.1 Base Requirements

Contact BTL for interim tests for this object

Base requirements must be met by any IUT that can contain Access Rights objects

<b>BTL - 7.3.2.X59.1 - Enable Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.X59.2 - Negative Rules Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.X59.3 - Positive Access Rules Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.X59.4 - Accompaniment Test</b>		
	<b>Test Conditionality</b>	If the IUT does not support the Accompaniment_Time in Access Point Object Type, this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 3.47.2 Contains an Object with Reliability\_Evaluation\_Inhibit Property

The IUT contains, or can be made to contain, a Reliability\_Evaluation\_Inhibit property that is configurable to a value of TRUE.

<b>BTL - 7.3.1.X8.1 - Reliability_Evaluation_Inhibit Test</b>		
	<b>Test Conditionality</b>	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.X8.2 - Reliability_Evaluation_Inhibit Summarization Test</b>		
	<b>Test Conditionality</b>	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

## 3.48 Access Credential Object

### 3.48.1 Base Requirements

Contact BTL for interim tests for this object

Base requirements must be met by any IUT that can contain Access Credential objects

BTL - 7.3.2.X60.1 - Credential Status, Credential Disable and Reason for Disable Test		
	Test Conditionality	If the IUT does not support certain reasons for disable then the corresponding steps shall be skipped.
	Test Directives	
	Testing Hints	
BTL - 7.3.2.X60.2 - Activation Time and Expiration Time Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
BTL - 7.3.2.X60.3 - Disabled Access Rights Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 3.48.2 Supports Days\_Remaining Property

The IUT supports the Access Credential Objects with Days\_Remaining property

BTL - 7.3.2.X60.4 - Days Remaining and Uses Remaining Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 3.48.3 Supports Absentee\_Limit Property

The IUT supports the Access Credential Objects with Absentee\_Limit property

BTL - 7.3.2.X60.5 - Absentee Limit Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 3.48.4 Supports Last\_Access\_Point, Last\_Use\_Time and Last\_Access\_Event Properties

The IUT supports the Access Credential Objects with Last\_Access\_Point, Last\_Use\_Time and Last\_Access\_Event properties

BTL - 7.3.2.X60.6 - Last Access Point, Last Use Time and Last Access Event Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 3.48.5 Supports Writable Extended\_Time\_Enable Property

The IUT supports the Access Credential Objects with writable Extended\_Time\_Enable property

BTL - 7.3.2.X60.7 - Extended Time Enable Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 3.48.6 Contains an Object with Reliability\_Evaluation\_Inhibit Property

The IUT contains, or can be made to contain, a Reliability\_Evaluation\_Inhibit property that is configurable to a value of TRUE.

BTL - 7.3.1.X8.1 - Reliability Evaluation Inhibit Test		
	Test Conditionality	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	Test Directives	
	Testing Hints	
BTL - 7.3.1.X8.2 - Reliability Evaluation Inhibit Summarization Test		
	Test Conditionality	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	Test Directives	
	Testing Hints	

## 3.49 Credential Data Input Object

### 3.49.1 Base Requirements

Base requirements must be met by any IUT that can contain Credential Data Input objects

BTL - 7.3.2.X61.1 - Return From Out Of Service Undefined Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
BTL - 7.3.2.X61.2 - Read Valid Authentication Factor Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 3.49.2 Supports Writable Out\_Of\_Service Property

The Out\_Of\_Service property in Credential Data Input objects contained in the IUT are writable.

135.1-2013 - 7.3.1.1 - Out Of Service, Status Flags, and Reliability Tests		
	Test Conditionality	Must be executed.
	Test Directives	The test shall be executed using a Credential Data Input object
	Testing Hints	

### 3.49.3 Contains an Object with Reliability\_Evaluation\_Inhibit Property

The IUT contains, or can be made to contain, a Reliability\_Evaluation\_Inhibit property that is configurable to a value of TRUE.

BTL - 7.3.1.X8.1 - Reliability_Evaluation_Inhibit Test		
	Test Conditionality	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	Test Directives	
	Testing Hints	
BTL - 7.3.1.X8.2 - Reliability_Evaluation_Inhibit Summarization Test		
	Test Conditionality	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	Test Directives	
	Testing Hints	

---

## **3.50 Network Security Object**

---

### **3.50.1 Base Requirements**

Contact BTL for interim tests for this object

## 3.51 Notification Forwarder Object

### 3.51.1 Base Requirements

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

<b>Verify Checklist</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify the IUT claims Protocol_Revision 13 or higher.
	<b>Testing Hints</b>	
<b>BTL - 7.2.X5 - Time Non-Pattern Properties Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	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.
	<b>Testing Hints</b>	
<b>135.1-2013q - 7.3.2.30.8.1 - Time Count Down Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013q - 7.3.2.30.8.2 - Expiration Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013q - 7.3.2.30.8.3 - Time Renewal Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013q - 7.3.2.30.8.4 - Resubscription Update Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013q - 7.3.2.30.8.5 - Delete Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Perform this using base Setup 1.
	<b>Testing Hints</b>	
<b>135.1-2013q - 7.3.2.30.8.6 - Subscription Of Similar Entries Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Perform this using base Setup 1.
	<b>Testing Hints</b>	
<b>135.1-2013q - 7.3.2.30.13.1 - Recipient_List Persistence Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013q - 7.3.2.21.3.7 - Recipient_List non-volatility Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013q - 7.3.2.30.13.2 - Subscribed_Recipients Persistence Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013q - 7.3.2.30.14.1 - Time Remaining Range Test</b>		
	<b>Test Conditionality</b>	Must be executed.



	<b>Test Directives</b>	
	<b>Testing Hints</b>	
	<b>135.1-2013q - 7.3.2.30.14.2 - Recipient Capacity Test</b>	
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.23.2.X10 - Time Non-Pattern Properties Test using WritePropertyMultiple Service</b>		
	<b>Test Conditionality</b>	This test shall only be applied to devices claiming Protocol_Revision 11 or higher and which supports execution of WritePropertyMultiple.
	<b>Test Directives</b>	Apply to the fromTime and again to the toTime element in a Recipient_List property in a Notification Forwarder Object.
	<b>Testing Hints</b>	

### 3.51.2 Supports DM-LM-B

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

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

### 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.

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

### 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.

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

### 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.

<b>135.1-2013q - 7.3.2.30.10 - Port_Filter Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Test each port value, if the IUT is a router.
	<b>Testing Hints</b>	
<b>Verify EPICS</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	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.
	<b>Testing Hints</b>	

### 3.51.6 Supports Writable Out\_Of\_Service Property

The IUT contains, or can be made to contain, an Out\_Of\_Service property that is writable.

<b>BTL - 7.3.2.30.6 - Out_Of_Service Property Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 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.

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

## 3.52 Alert Enrollment Object

### 3.52.1 Base Requirements

Base requirements must be met by any IUT that can contain Alert Enrollment Objects.

135.1-2013q - 7.3.1.X1.1 - Alert Enrollment Reports The Source Object		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
135.1-2013q - 7.3.1.X1.2 - Alert Enrollment Does Not Generate Acknowledgeable Transitions		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 3.53 Channel Object

#### 3.53.1 Base Requirements

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

<b>BTL - 7.3.2.X40.2 - Last_Priority Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.X40.3 - WriteGroup Service Support Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.X40.4 - Propagation Entirety Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.X40.5 - Write_Status Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Repeat the test by sending a write group request to both BROADCAST and to the IUT.
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.X40.12 - Write Priority Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Repeat the test by writing to the Present_Value of the Channel object using WritePropertyMultiple and WriteGroup service if supported by the IUT.
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.X40.13 - Writing with a NULL value Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 3.53.2 Supports Writable Out\_Of\_Service Property

The Out\_Of\_Service property in Channel objects contained in the IUT are writable.

<b>135.1-2013 - 7.3.1.1 - Out_Of_Service, Status_Flags, and Reliability Tests</b>		
	<b>Test Conditionality</b>	If this property is writable, this test must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 3.53.3 Supports an Object with Reliability\_Evaluation\_Inhibit Property

The IUT contains, or can be made to contain, a Reliability\_Evaluation\_Inhibit property that is configurable to a value of TRUE.

<b>BTL - 7.3.1.X8.1 - Reliability_Evaluation_Inhibit Test</b>		
	<b>Test Conditionality</b>	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.X8.2 - Reliability_Evaluation_Inhibit Summarization Test</b>		

<b>Test Conditionality</b>	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
<b>Test Directives</b>	
<b>Testing Hints</b>	

### 3.53.4 Supports Allow\_Group\_Delay\_Inhibit

The object contains Allow\_Group\_Delay\_Inhibit property.

<b>BTL - 7.3.2.X40.6 - Allow_Group_Delay_Inhibit Test</b>		
<b>Test Conditionality</b>	Must be executed.	
<b>Test Directives</b>		
<b>Testing Hints</b>		

### 3.53.5 Is Able to Correctly Handle Targets of BOOLEAN Data Type

The channel object on the IUT is able to correctly handle targets of BOOLEAN data type.

BTL - 7.3.2.X40.7 - Numeric to BOOLEAN Coercion Rule Test		
	Test Conditionality	Test with both local and remote target. If DS-WG-E-B is not supported, test only with local target. If the IUT does not contain or cannot be made to contain any writable BOOLEAN properties, skip the test with local target.
	Test Directives	Repeat the test by writing all numeric data type (Unsigned, INTEGER, REAL, Double and ENUMERATED) to a Present_Value of a Channel object containing a BOOLEAN target object property reference.
	Testing Hints	
BTL - 7.3.2.X40.10 - Invalid Datatype Coercion Test		
	Test Conditionality	Test with both local and remote target. If DS-WG-E-B is not supported, test only with local target. Skip the test for (an) invalid data type(s) not supported by the device containing a target object property.
	Test Directives	Repeat the test by writing all invalid data type (OCTET STRING, CharacterString, BIT STRING, Date, Time, BACnetObjectIdentifier and BACnetLightingCommand) to a Present_Value of a Channel object containing a BOOLEAN target object property reference.
	Testing Hints	
BTL - 7.3.2.X40.11 - No Coercion Test		
	Test Conditionality	Test with both local and remote target. If DS-WG-E-B is not supported, test only with local target. If the IUT does not contain or cannot be made to contain any writable object properties of the written value type, skip the test with local target. Select a value within an acceptable range of the target object property.
	Test Directives	Repeat the test by writing all data type that does not require coercion (NULL, BOOLEAN) to a Present_Value of a Channel object containing a BOOLEAN target object property reference.
	Testing Hints	

### 3.53.6 Is Able to Correctly Handle Targets of Unsigned Data Type

The channel object on the IUT is able to correctly handle targets of Unsigned data type.

<b>BTL - 7.3.2.X40.8 - BOOLEAN to Numeric Coercion Rule Test</b>
--

	<b>Test Conditionality</b>	Test with both local and remote target. If DS-WG-E-B is not supported, test only with local target. If the IUT does not contain or cannot be made to contain any writable Numeric properties, skip the test with local target.
	<b>Test Directives</b>	Test with Unsigned target object reference.
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.X40.9 - Unsigned/INTEGER/REAL/Double to Numeric Coercion Rule Test</b>		
	<b>Test Conditionality</b>	Test with both local and remote target. If DS-WG-E-B is not supported, test only with local target. If the IUT does not contain or cannot be made to contain any writable Numeric properties, skip the test with local target. Select a value within an acceptable range of the target object property.
	<b>Test Directives</b>	Repeat the test by writing all numeric data type (Unsigned, INTEGER, REAL, Double and ENUMERATED) to a Present_Value of a Channel object containing an Unsigned target object property reference. When writing either unsigned or INTEGER data type, use a value less than 2147483647 as par Coercion Rule 3-6 defined in ASHRAE 135. When writing either REAL or Double data type, use a value less than 2147483000 as par Coercion Rule 3-6 defined in ASHRAE 135.
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.X40.10 - Invalid Datatype Coercion Test</b>		
	<b>Test Conditionality</b>	Test with both local and remote target. If DS-WG-E-B is not supported, test only with local target. If the IUT does not contain or cannot be made to contain any writable Numeric properties, skip the test with local target
	<b>Test Directives</b>	Repeat the test by writing all invalid data type (OCTET STRING, CharacterString, BIT STRING, Date, Time and BACnetLightingCommand) to a Present_Value of a Channel object containing an Unsigned target object property reference.
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.X40.11 - No Coercion Test</b>		
	<b>Test Conditionality</b>	Test with both local and remote target. If DS-WG-E-B is not supported, test only with local target. If the IUT does not contain or cannot be made to contain any writable object properties of the written value type, skip the test with local target. Select a value within an acceptable range of the target object property.
	<b>Test Directives</b>	Repeat the test by writing all data type that does not require coercion (NULL, Unsigned, ENUMERATED, and BACnetObjectIdentifier) to a Present_Value of a Channel object containing an Unsigned target object property reference.
	<b>Testing Hints</b>	

### 3.53.7Is Able to Correctly Handle Targets of INTEGER Data Type

The channel object on the IUT is able to correctly handle targets of INTEGER data type.

<b>BTL - 7.3.2.X40.8 - BOOLEAN to Numeric Coercion Rule Test</b>		
	<b>Test Conditionality</b>	Test with both local and remote target. If DS-WG-E-B is not supported, test only with local target. If the IUT does not contain or cannot be made to contain any writable INTEGER properties, skip the test with local target.
	<b>Test Directives</b>	Test with INTEGER target object reference.
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.X40.9 - Unsigned/INTEGER/REAL/Double to Numeric Coercion Rule Test</b>		

	<b>Test Conditionality</b>	Test with both local and remote target. If DS-WG-E-B is not supported, test only with local target. If the IUT does not contain or cannot be made to contain any writable Numeric properties, skip the test with local target. Select a value within an acceptable range of the target object property.
	<b>Test Directives</b>	Repeat the test by writing all numeric data type (Unsigned, INTEGER, REAL, Double and ENUMERATED) to a Present_Value of a Channel object containing an INTEGER target object property reference.  When writing either REAL or DOUBLE data type, use a value more than - 2147483000 and less than 214783000 as per Coercion Rule 3-6 defined in ASHRAE 135.
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.X40.10 - Invalid Datatype Coercion Test</b>		
	<b>Test Conditionality</b>	Test with both local and remote target. If DS-WG-E-B is not supported, test only with local target. Skip the test for (an) invalid data type(s) not supported by the device containing a target object property.
	<b>Test Directives</b>	Repeat the test by writing all invalid data type (OCTET STRING, CharacterString, BIT STRING, Date, Time, BACnetObjectIdentifier and BACnetLightingCommand) to a Present_Value of a Channel object containing an INTEGER target object property reference.
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.X40.11 - No Coercion Test</b>		
	<b>Test Conditionality</b>	Test with both local and remote target. If DS-WG-E-B is not supported, test only with local target. If the IUT does not contain or cannot be made to contain any writable object properties of the written value type, skip the test with local target. Select a value within an acceptable range of the target object property.
	<b>Test Directives</b>	Repeat the test by writing all data type that does not require coercion (NULL and INTEGER) to a Present_Value of a Channel object containing an INTEGER target object property reference.
	<b>Testing Hints</b>	

### 3.53.8 Is Able to Correctly Handle Targets of REAL Data Type

The channel object on the IUT is able to correctly handle targets of REAL data type.

<b>BTL - 7.3.2.X40.8 - BOOLEAN to Numeric Coercion Rule Test</b>		
	<b>Test Conditionality</b>	Test with both local and remote target. If DS-WG-E-B is not supported, test only with local target. If the IUT does not contain or cannot be made to contain any writable BOOLEAN properties, skip the test with local target.
	<b>Test Directives</b>	Test with REAL target object property reference.
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.X40.9 - Unsigned/INTEGER/REAL/Double to Numeric Coercion Rule Test</b>		
	<b>Test Conditionality</b>	Test with both local and remote target. If DS-WG-E-B is not supported, test only with local target. If the IUT does not contain or cannot be made to contain any writable Numeric properties, skip the test with local target. Select a value within an acceptable range of the target object property.

	<b>Test Directives</b>	<p>Repeat the test by writing all numeric datatype (Unsigned, INTEGER, REAL, Double and ENUMERATED) to a Present_Value of a Channel object containing a REAL target object property reference.</p> <p>When writing either Unsigned or INTEGER data type, use a value in precision up to seven significant digits as par Coercion Rules defined in ASHRAE 135.</p> <p>When writing DOUBLE data type, use a value less than <math>3.4 \times 10^{+38}</math> and more than <math>3.4 \times 10^{-38}</math> as par Coercion Rules defined in ASHRAE 135.</p>
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.X40.10 - Invalid Datatype Coercion Test</b>		
	<b>Test Conditionality</b>	<p>Test with both local and remote target.</p> <p>If DS-WG-E-B is not supported, test only with local target.</p> <p>Skip the test for (an) invalid data type(s) not supported by the device containing a target object property.</p>
	<b>Test Directives</b>	Repeat the test by writing all invalid data type (OCTET STRING, CharacterString, BIT STRING, Date, Time, BACnetObjectIdentifier and BACnetLightingCommand) to a Present_Value of a Channel object containing an REAL target object property reference.
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.X40.11 - No Coercion Test</b>		
	<b>Test Conditionality</b>	<p>Test with both local and remote target.</p> <p>If DS-WG-E-B is not supported, test only with local target.</p> <p>If the IUT does not contain or cannot be made to contain any writable object properties of the written value type, skip the test with local target.</p> <p>Select a value within an acceptable range of the target object property.</p>
	<b>Test Directives</b>	Repeat the test by writing all data type that does not require coercion (NULL and REAL) to a Present_Value of a Channel object containing a REAL target object property reference.
	<b>Testing Hints</b>	

### 3.53.9 Is Able to Correctly Handle Targets of Double Data Type

The channel object on the IUT is able to correctly handle targets of DOUBLE data type.

<b>BTL - 7.3.2.X40.8 - BOOLEAN to Numeric Coercion Rule Test</b>		
	<b>Test Conditionality</b>	<p>Test with both local and remote target.</p> <p>If DS-WG-E-B is not supported, test only with local target.</p> <p>If the IUT does not contain or cannot be made to contain any writable BOOLEAN properties, skip the test with local target.</p>
	<b>Test Directives</b>	Test with Double target object property reference.
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.X40.9 - Unsigned/INTEGER/REAL/Double to Numeric Coercion Rule Test</b>		
	<b>Test Conditionality</b>	<p>Test with both local and remote target.</p> <p>If DS-WG-E-B is not supported, test only with local target.</p> <p>If the IUT does not contain or cannot be made to contain any writable Numeric properties, skip the test with local target.</p> <p>Select a value within an acceptable range of the target object property.</p>
	<b>Test Directives</b>	Repeat the test by writing all numeric datatype (Unsigned, INTEGER, REAL, Double and ENUMERATED) to a Present_Value of a Channel object containing a Double target object property reference.
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.X40.10 - Invalid Datatype Coercion Test</b>		



	<b>Test Conditionality</b>	Test with both local and remote target. If DS-WG-E-B is not supported, test only with local target. Skip the test for (an) invalid data type(s) not supported by the device containing a target object property.
	<b>Test Directives</b>	Repeat the test by writing all invalid data type (OCTET STRING, CharacterString, BIT STRING, Date, Time, BACnetObjectIdentifier and BACnetLightingCommand) to a Present_Value of a Channel object containing a Double target object property reference.
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.X40.11 - No Coercion Test</b>		
	<b>Test Conditionality</b>	Test with both local and remote target. If DS-WG-E-B is not supported, test only with local target. If the IUT does not contain or cannot be made to contain any writable object properties of the written value type, skip the test with local target. Select a value within an acceptable range of the target object property.
	<b>Test Directives</b>	Repeat the test by writing all data type that does not require coercion (NULL and Double) to a Present_Value of a Channel object containing a Double target object property reference.
	<b>Testing Hints</b>	

### 3.53.10 Is Able to Correctly Handle Targets of ENUMERATED Data Type

The channel object on the IUT is able to correctly handle targets of Enumerated data type.

<b>BTL - 7.3.2.X40.8 - BOOLEAN to Numeric Coercion Rule Test</b>		
	<b>Test Conditionality</b>	Test with both local and remote target. If DS-WG-E-B is not supported, test only with local target. If the IUT does not contain or cannot be made to contain any writable Numeric properties, skip the test with local target. Select a value within an acceptable range of the target object property.
	<b>Test Directives</b>	Test with Enumerated target object property reference.
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.X40.9 - Unsigned/INTEGER/REAL/Double to Numeric Coercion Rule Test</b>		
	<b>Test Conditionality</b>	Test with both local and remote target. If DS-WG-E-B is not supported, test only with local target. If the IUT does not contain or cannot be made to contain any writable Numeric properties, skip the test with local target. Select a value within an acceptable range of the target object property.
	<b>Test Directives</b>	Repeat the test by writing all numeric datatype (Unsigned, INTEGER, REAL, Double and Enumerated) to a Present_Value of a Channel object containing an Enumerated target object property reference.
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.X40.10 - Invalid Datatype Coercion Test</b>		
	<b>Test Conditionality</b>	Test with both local and remote target. If DS-WG-E-B is not supported, test only with local target. Skip the test for (an) invalid data type(s) not supported by the device containing a target object property.
	<b>Test Directives</b>	Repeat the test by writing all invalid data type (OCTET STRING, CharacterString, BitString, Date, Time, BACnetObjectIdentifier and BACnetLightingCommand) to a Present_Value of a Channel object containing an Enumerated target object property reference.
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.X40.11 - No Coercion Test</b>		

<b>Test Conditionality</b>	Test with both local and remote target. If DS-WG-E-B is not supported, test only with local target. If the IUT does not contain or cannot be made to contain any writable object properties of the written value type, skip the test with local target. Select a value within an acceptable range of the target object property.
<b>Test Directives</b>	Repeat the test by writing all data type that does not require coercion (NULL, Unsigned and Enumerated) to a Present_Value of a Channel object containing an Enumerated target object property reference.
<b>Testing Hints</b>	

### 3.53.11 Is Able to Correctly Handle Targets of NULL Data Type

The channel object on the IUT is able to correctly handle targets of NULL data type

<b>BTL - 7.3.2.X40.10 - Invalid Datatype Coercion Test</b>	
<b>Test Conditionality</b>	Test with both local and remote target. If DS-WG-E-B is not supported, test only with local target. Skip the test for (an) invalid data type(s) not supported by the device containing a target object property.
<b>Test Directives</b>	Repeat the test by writing all invalid data type (BOOLEAN, Unsigned, INTEGER, REAL, Double, CharacterString, BitString, Enumerated, Date, Time, BACnetObjectIdentifier and BACnetLightingCommand) to a Present_Value of a Channel object containing an OCTET STRING target object property reference.
<b>Testing Hints</b>	
<b>BTL - 7.3.2.X40.11 - No Coercion Test</b>	
<b>Test Conditionality</b>	Test with both local and remote target. If DS-WG-E-B is not supported, test only with local target. If the IUT does not contain or cannot be made to contain any writable object properties of the written value type, skip the test with local target. Select a value within an acceptable range of the target object property.
<b>Test Directives</b>	Repeat the test by writing all data type that does not require coercion (NULL and OCTET STRING) to a Present_Value of a Channel object containing an OCTET STRING target object property reference.
<b>Testing Hints</b>	

### 3.53.12 Is Able to Correctly Handle Targets of OCTET STRING Data Type

The channel object on the IUT is able to correctly handle targets of OCTET STRING data type

<b>BTL - 7.3.2.X40.10 - Invalid Datatype Coercion Test</b>	
<b>Test Conditionality</b>	Test with both local and remote target. If DS-WG-E-B is not supported, test only with local target. Skip the test for (an) invalid data type(s) not supported by the device containing a target object property.
<b>Test Directives</b>	Repeat the test by writing all invalid data type (BOOLEAN, Unsigned, INTEGER, REAL, Double, CharacterString, BitString, Enumerated, Date, Time, BACnetObjectIdentifier and BACnetLightingCommand) to a Present_Value of a Channel object containing an OCTET STRING target object property reference.
<b>Testing Hints</b>	
<b>BTL - 7.3.2.X40.11 - No Coercion Test</b>	

<b>Test Conditionality</b>	Test with both local and remote target. If DS-WG-E-B is not supported, test only with local target. If the IUT does not contain or cannot be made to contain any writable object properties of the written value type, skip the test with local target. Select a value within an acceptable range of the target object property.
<b>Test Directives</b>	Repeat the test by writing all data type that does not require coercion (NULL and OCTET STRING) to a Present_Value of a Channel object containing an OCTET STRING target object property reference.
<b>Testing Hints</b>	

### 3.53.13 Is Able to Correctly Handle Targets of CharacterString Data Type

The channel object on the IUT is able to correctly handle targets of Character String data type.

<b>BTL - 7.3.2.X40.10 - Invalid Datatype Coercion Test</b>	
<b>Test Conditionality</b>	Test with both local and remote target. If DS-WG-E-B is not supported, test only with local target. Skip the test for (an) invalid data type(s) not supported by the device containing a target object property.
<b>Test Directives</b>	Repeat the test by writing all invalid data type (BOOLEAN, Unsigned, INTEGER, REAL, Double, OCTET STRING, BitString, Enumerated, Date, Time, BACnetObjectIdentifier and BACnetLightingCommand) to a Present_Value of a Channel object containing a Character String target object property reference.
<b>Testing Hints</b>	
<b>BTL - 7.3.2.X40.11 - No Coercion Test</b>	
<b>Test Conditionality</b>	Test with both local and remote target. If DS-WG-E-B is not supported, test only with local target. If the IUT does not contain or cannot be made to contain any writable object properties of the written value type, skip the test with local target. Select a value within an acceptable range of the target object property.
<b>Test Directives</b>	Repeat the test by writing all data type that does not require coercion (NULL and Character String) to a Present_Value of a Channel object containing a Character String target object property reference.
<b>Testing Hints</b>	

### 3.53.14 Is Able to Correctly Handle Targets of BIT STRING Data Type

The channel object on the IUT is able to correctly handle targets of BITSTRING data type.

<b>BTL - 7.3.2.X40.10 - Invalid Datatype Coercion Test</b>	
<b>Test Conditionality</b>	Test with both local and remote target. If DS-WG-E-B is not supported, test only with local target. Skip the test for (an) invalid data type(s) not supported by the device containing a target object property.
<b>Test Directives</b>	Repeat the test by writing all invalid data type (BOOLEAN, Unsigned, INTEGER, REAL, Double, OCTET STRING, Character String, Enumerated, Date, Time, BACnetObjectIdentifier and BACnetLightingCommand) to a Present_Value of a Channel object containing a BIT STRING target object property reference.
<b>Testing Hints</b>	
<b>BTL - 7.3.2.X40.11 - No Coercion Test</b>	

<b>Test Conditionality</b>	Test with both local and remote target. If DS-WG-E-B is not supported, test only with local target. If the IUT does not contain or cannot be made to contain any writable object properties of the written value type, skip the test with local target. Select a value within an acceptable range of the target object property.
<b>Test Directives</b>	Repeat the test by writing all data type that does not require coercion (NULL and BitString) to a Present_Value of a Channel object containing a BitString target object property reference.
<b>Testing Hints</b>	

### 3.53.15 Is Able to Correctly Handle Targets of Date Data Type

The channel object on the IUT is able to correctly handle targets of Date data type

<b>BTL - 7.3.2.X40.10 - Invalid Datatype Coercion Test</b>	
<b>Test Conditionality</b>	Test with both local and remote target. If DS-WG-E-B is not supported, test only with local target. Skip the test for (an) invalid data type(s) not supported by the device containing a target object property.
<b>Test Directives</b>	Repeat the test by writing all invalid data type (BOOLEAN, Unsigned, INTEGER, REAL, Double, OCTET STRING, Character String, BitString, Enumerated, Time, BACnetObjectIdentifier and BACnetLightingCommand) to a Present_Value of a Channel object containing a Date target object property reference.
<b>Testing Hints</b>	
<b>BTL - 7.3.2.X40.11 - No Coercion Test</b>	
<b>Test Conditionality</b>	
<b>Test Directives</b>	Repeat the test by writing all data type that does not require coercion (NULL and Date) to a Present_Value of a Channel object containing a Date target object property reference.
<b>Testing Hints</b>	

### 3.53.16 Is Able to Correctly Handle Targets of Time Data Type

The channel object on the IUT is able to correctly handle targets of Time data type.

<b>BTL - 7.3.2.X40.10 - Invalid Datatype Coercion Test</b>	
<b>Test Conditionality</b>	Test with both local and remote target. If DS-WG-E-B is not supported, test only with local target. Skip the test for (an) invalid data type(s) not supported by the device containing a target object property.
<b>Test Directives</b>	Repeat the test by writing all invalid data type (BOOLEAN, Unsigned, INTEGER, REAL, Double, OCTET STRING, Character String, BitString, Enumerated, Date, BACnetObjectIdentifier and BACnetLightingCommand) to a Present_Value of a Channel object containing a Time target object property reference.
<b>Testing Hints</b>	
<b>BTL - 7.3.2.X40.11 - No Coercion Test</b>	
<b>Test Conditionality</b>	
<b>Test Directives</b>	Repeat the test by writing all data type that does not require coercion (NULL and Time) to a Present_Value of a Channel object containing an OCTET STRING target object property reference.
<b>Testing Hints</b>	

### 3.53.17 Is Able to Correctly Handle Targets of BACnetObjectIdentifier Data Type

The channel object on the IUT is able to correctly handle targets of BACnetObjectIdentifier data type.

<b>BTL - 7.3.2.X40.10 - Invalid Datatype Coercion Test</b>		
	<b>Test Conditionality</b>	Test with both local and remote target. If DS-WG-E-B is not supported, test only with local target. Skip the test for (an) invalid data type(s) not supported by the device containing a target object property.
	<b>Test Directives</b>	Repeat the test by writing all invalid data type (BOOLEAN, INTEGER, REAL, Double, OCTET STRING, Character String, BitString, Enumerated, Date, Time and BACnetLightingCommand) to a Present_Value of a Channel object containing a BACnetObjectIdentifier target object property reference.
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.X40.11 - No Coercion Test</b>		
	<b>Test Conditionality</b>	Test with both local and remote target. If DS-WG-E-B is not supported, test only with local target. If the IUT does not contain or cannot be made to contain any writable object properties of the written value type, skip the test with local target. Select a value within an acceptable range of the target object property.
	<b>Test Directives</b>	Repeat the test by writing data type that does not require coercion (NULL, Unsigned and BACnetObjectIdentifier) to a Present_Value of a Channel object containing a BACnetObjectIdentifier target object property reference.
	<b>Testing Hints</b>	

### 3.53.18 Is Able to Correctly Handle Targets of BACnetLightingCommand Data Type

The channel object on the IUT is able to correctly handle targets of BACnetLighting Command data type.

<b>BTL - 7.3.2.X40.10 - Invalid Datatype Coercion Test</b>		
	<b>Test Conditionality</b>	Test with both local and remote target. If DS-WG-E-B is not supported, test only with local target. Skip the test for (an) invalid data type(s) not supported by the device containing a target object property.
	<b>Test Directives</b>	Repeat the test by writing all invalid data type (NULL, BOOLEAN, Unsigned, INTEGER, REAL, Double, OCTET STRING, Character String, BitString, Enumerated, Date, Time and BACnetObjectIdentifier) to a Present_Value of a Channel object containing a BACnetLightingCommand target object property reference.
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.X40.11 - No Coercion Test</b>		
	<b>Test Conditionality</b>	Test with both local and remote target. If DS-WG-E-B is not supported, test only with local target. If the IUT does not contain or cannot be made to contain any writable object properties of the written value type, skip the test with local target. Select a value within an acceptable range of the target object property.
	<b>Test Directives</b>	Repeat the test by writing BACnetLightingCommand value to a Present_Value of a Channel object containing a BACnetObjectIdentifier target object property reference.

	<b>Testing Hints</b>	
--	----------------------	--

### 3.53.19 Supports DS-WG-I-B

The IUT supports DS-WG-I-B in order to execute a WriteGroup service

Verify Checklist		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support for DS-WG-I-B in the checklist
	<b>Testing Hints</b>	

### 3.53.20 Supports DS-WG-E-B

The IUT supports DS-WG-E-B in order to execute a WriteGroup service and then to initiate a write service to propagate a value to one or more target object on one or more remote device.

Verify Checklist		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support for DS-WG-E-B in the checklist
	<b>Testing Hints</b>	

## 3.54 Lighting Output Object

### 3.54.1 Base Requirements

Base requirements must be met by any IUT that can contain Lighting Output objects.

BTL - 7.3.2.X54.21 - Lighting Output Tracking Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
BTL - 7.3.2.X54.22 - Lighting Output Present Value between 0.0 and 1.0 Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 3.54.2 Supports Command Prioritization

The objects contain a priority array and support command prioritization.

135.1-2013 - 7.3.1.2 - Relinquish Default Test		
	Test Conditionality	If no object can be made to meet the configuration requirements, this test shall be skipped.
	Test Directives	
	Testing Hints	
135.1-2013 - 7.3.1.3 - Command Prioritization Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 3.54.3 Supports all BACnetLightingOperations

The objects can perform all operations defined by the BACnetLightingOperations enumeration.

BTL - 7.3.2.X54.31 - Lighting Command Operation NONE Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
BTL - 7.3.2.X54.32 - Lighting Command Operation FADE_TO Test		
	Test Conditionality	Must be executed.
	Test Directives	Repeat the test by using the BACnetLightingCommand without the optional fields (priority and fade-time) and check that PTY1= Lighting_Command_Default_Priority and fade-time = Default_Fade_Time
	Testing Hints	
BTL - 7.3.2.X54.33 - Lighting Command Operation RAMP_TO Test		
	Test Conditionality	Must be executed.
	Test Directives	Repeat the test by using the BACnetLightingCommand without the optional fields (priority and ramp-rate) and check that PTY1= Lighting_Command_Default_Priority and ramp-rate = Default_Ramp_Rate
	Testing Hints	
BTL - 7.3.2.X54.34 - Lighting Command Operation STEP_UP Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

	<b>Test Directives</b>	Repeat the test by using the BACnetLightingCommand without the optional fields (priority and ramp-rate) and check that PTY1= Lighting_Command_Default_Priority and step-increment = Default_Step_Increment
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.X54.35 - Lighting Command Operation STEP_DOWN Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Repeat the test by using the BACnetLightingCommand without the optional fields (priority and ramp-rate) and check that PTY1= Lighting_Command_Default_Priority and step-increment = Default_Step_increment
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.X54.36 - Lighting Command Operation STEP_ON Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Repeat the test by using the BACnetLightingCommand without the optional fields (priority and ramp-rate) and check that PTY1= Lighting_Command_Default_Priority and step-increment = Default_step-increment
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.X54.37 - Lighting Command Operation STEP_OFF Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Repeat the test by using the BACnetLightingCommand without the optional fields (priority and ramp-rate) and check that PTY1= Lighting_Command_Default_Priority and step-increment = Default_step-increment
	<b>Testing Hints</b>	

### 3.54.4 Supports Writable Out\_Of\_Service Property

The Out\_Of\_Service property in Lighting Output objects contained in the IUT are writable.

<b>135.1-2013 - 7.3.1.1 - Out Of Service, Status Flags, and Reliability Tests</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	The test shall be executed using an Lighting Output object
	<b>Testing Hints</b>	

### 3.54.5 Supports Blink-Warn

The Blink\_Warn\_Enable property in Lighting Output is writable or can be changed to TRUE by other means.

<b>BTL - 7.3.1.X41.Y1 - Blink-Warn WARN Command Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Must be executed using both the Present_Value and Lighting_Command properties.
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.X41.Y2 - Blink-Warn WARN OFF Command Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Must be executed using both the Present_Value and Lighting_Command properties.
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.X41.Y3 - Blink-Warn WARN_RELINQUISH Command Test</b>		
	<b>Test Conditionality</b>	Must be executed.



	<b>Test Directives</b>	Must be executed using both the Present_Value and Lighting_Command properties.
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.X41.Y4 - Blink-Warn STOP Command Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Repeat the test with WARN_OFF and WARN_RELINQUISH commands
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.X41.Y5 - Blink-Warn WARN Command Failure Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.X41.Y6 - Blink-Warn WARN_OFF Command Failure Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.X41.Y7 - Blink-Warn WARN_RELINQUISH Command Failure Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.X41.Y8 - Blink-Warn WARN_OFF Command Halted Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.X41.Y9 - Blink Warn WARN_RELINQUISH Command Halted Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 3.54.6 Supports Transition Property

The IUT contains Lighting Output Objects in which the Transition property is supported.

<b>BTL - 7.3.2.X54.41 - Transition None Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.X54.42 - Transition Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 3.54.7 Supports Feedback\_Value Property

The IUT contains Lighting Output Objects in which the Feedback\_Value property is supported.

<b>BTL - 7.3.2.X54.51 - Feedback_Value Clamping Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 3.54.8 Supports Min\_Actual\_Value and Max\_Actual\_Value Properties

The IUT contains Lighting Output Objects in which the the Min\_Actual\_Value and Max\_Actual\_Value properties are supported.

<b>BTL - 7.3.2.X54.61 - Min_Actual_Value and Max_Actual_Value Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.X54.62 - Min_Actual_Value and Max_Actual_Value ScalingTest</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 3.54.9 Contains an Object with Reliability\_Evaluation\_Inhibit Property

The IUT contains, or can be made to contain, a Reliability\_Evaluation\_Inhibit property that is configurable to a value of TRUE.

<b>BTL - 7.3.1.X8.1 - Reliability_Evaluation_Inhibit Test</b>		
	<b>Test Conditionality</b>	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.X8.2 - Reliability_Evaluation_Inhibit Summarization Test</b>		
	<b>Test Conditionality</b>	If no object exists in the IUT for which fault conditions can be generated then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

## 3.55 Binary Lighting Output Object

### 3.55.1 Base Requirements

Base requirements must be met by any IUT that can contain Binary Lighting Output objects. There are no base requirements for this object.

### 3.55.2 Supports Command Prioritization

The objects contain a priority array and support command prioritization.

135.1-2013 - 7.3.1.2 - Relinquish Default Test		
	Test Conditionality	If no object can be made to meet the configuration requirements, this test shall be skipped.
	Test Directives	
	Testing Hints	
135.1-2013 - 7.3.1.3 - Command Prioritization Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 3.55.3 Supports Writable Out\_Of\_Service Property

The Out\_Of\_Service property in Binary Lighting Output objects contained in the IUT are writable.

135.1-2013 - 7.3.1.1 - Out_Of_Service, Status_Flags, and Reliability Tests		
	Test Conditionality	If Out_Of_Service is writable, this test must be executed.
	Test Directives	
	Testing Hints	

### 3.55.4 Supports Blink-Warn

The IUT supports blink-warn in the Binary Lighting Output object. The Blink\_Warn\_Enable property is true or can be set to true.

BTL - 7.3.1.X41.Y1 - Blink-Warn WARN Command Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
BTL - 7.3.1.X41.Y2 - Blink-Warn WARN_OFF Command Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
BTL - 7.3.1.X41.Y3 - Blink-Warn WARN_RELINQUISH Command Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
BTL - 7.3.1.X41.Y4 - Blink-Warn STOP Command Test		
	Test Conditionality	Must be executed.
	Test Directives	Repeat the test with WARN_OFF and WARN_RELINQUISH commands
	Testing Hints	
BTL - 7.3.1.X41.Y5 - Blink-Warn WARN Command Failure Test		
	Test Conditionality	Must be executed.
	Test Directives	Repeat the test with WARN_OFF and WARN_RELINQUISH commands
	Testing Hints	
BTL - 7.3.1.X41.Y6 - Blink-Warn WARN_OFF Command Failure Test		

	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.X41.Y7 - Blink-Warn WARN_RELINQUISH Command Failure Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.X41.Y8 - Blink-Warn WARN_OFF Command Halted Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.X41.Y9 - Blink-Warn WARN_RELINQUISH Command Halted Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 3.55.5 Supports Writable Polarity Property

The IUT supports a writable Polarity property in the Binary Output object.

<b>135.1-2013 - 7.3.2.6.3 - Polarity Property Tests</b>		
	<b>Test Conditionality</b>	Must be executed
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 3.55.6 Supports Strike Count Tracking

The IUT contains a Binary Lighting Output object that has the Strike\_Count and Time\_Of\_Strike\_Count\_Reset properties.

<b>BTL - 7.3.2.X41.Y10 - Strike Count Tests</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.X20 - Non-zero Writable Strike Count Test</b>		
	<b>Test Conditionality</b>	If no Binary Lighting Output object contains a writable Strike_Count that accepts writes of non-zero values then this test shall be executed. If IUT claims Protocol_Revision less than 16, then this test shall be skipped.
	<b>Test Directives</b>	This test shall be performed using a Binary Lighting Output object.
	<b>Testing Hints</b>	

### 3.55.7 Supports Elapsed Active Time Tracking

The IUT contains or can be made to contain an object with the Elapsed\_Active\_Time and Time\_Of\_Active\_Time\_Reset properties.

<b>BTL - 7.3.1.9 - Elapsed Active Time Test</b>		
	<b>Test Conditionality</b>	If no Binary Lighting Output object contains a writable Elapsed_Active_Time then this test shall be skipped.

	<b>Test Directives</b>	This test shall be performed using a Binary Lighting Output object.
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.X19 - Non-zero Writable Elapsed Active Time Test</b>		
	<b>Test Conditionality</b>	If no Binary Lighting Output object contains a writable Elapsed_Active_Time that accepts writes of non-zero values then this test shall be skipped. If IUT claims Protocol_Revision less than 16, then this test shall be skipped.
	<b>Test Directives</b>	This test shall be performed using a Binary Lighting Output object.
	<b>Testing Hints</b>	

### 3.55.8 Contains an object with Reliability\_Evaluation\_Inhibit Property

The IUT contains, or can be made to contain, a Reliability\_Evaluation\_Inhibit property that is configurable to a value of TRUE.

<b>BTL - 7.3.1.X8.1 - Reliability_Evaluation_Inhibit Test</b>		
	<b>Test Conditionality</b>	If no object exists in the IUT for which fault conditions can be generated, then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.X8.2 - Reliability_Evaluation_Inhibit Summarization Test</b>		
	<b>Test Conditionality</b>	If no object exists in the IUT for which fault conditions can be generated, then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

---

## **3.56 Network Port Object**

---

### **3.56.1 Base Requirements**

Contact BTL for interim tests for this object

---

## **3.57 Timer Object**

---

### **3.57.1 Base Requirements**

Contact BTL for interim tests for this object

---

## **3.58 Elevator Group Object**

---

### **3.58.1 Base Requirements**

Contact BTL for interim tests for this object



---

## **3.59 Lift Object**

---

### **3.59.1 Base Requirements**

Contact BTL for interim tests for this object

---

## **3.60 Escalator Object**

---

### **3.60.1 Base Requirements**

Contact BTL for interim tests for this object

## 3.61 File Object

### 3.61.1 Base Requirements

For File object, there are no base requirements.

### 3.61.2 Supports DM-BR-B

The IUT supports a data File that is readable and writable during Backup and Restore using AtomicReadFile and AtomicWriteFile requests.

Verify Checklist		
	Test Conditionality	Must be executed.
	Test Directives	Verify that the IUT claims support for Device Management - Backup and Restore - B in the Checklist.
	Testing Hints	

### 3.61.3 Supports a Record-Based File Object for a Purpose Other Than Backup and Restore

For a device which contains a record-based File object for a purpose other than Backup and Restore, there are no testing requirements.

### 3.61.4 Supports a Stream-Based File Object for a Purpose Other Than Backup and Restore

The IUT supports a data stream-based File that is not accessed during Backup and Restore.

BTL - 9.12.1.2.1 - Reading an Entire Stream-based File		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
135.1-2013 - 9.12.2.2.2 - Attempting to Read Data from a Nonexistent File		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
135.1-2013 - 9.12.2.2.3 - Attempting to Read Data Using the Wrong File Access Type		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 3.61.5 Contains a Writable Stream-Based File for a Purpose Other Than Backup and Restore

The IUT supports a data stream-based File that is not accessed during Backup and Restore.

BTL - 9.13.1.2.1 - Writing an Entire Stream-based File		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

<b>BTL - 9.13.1.2.3 - Appending Data to the End of a File</b>		
	<b>Test Conditionality</b>	If the file size cannot be changed or if the IUT does not support files that cannot be modified except by replacing the entire file, then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.13.1.2.4 - Truncating a File</b>		
	<b>Test Conditionality</b>	If the file size cannot be changed, then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.13.1.2.5 - Deleting a File</b>		
	<b>Test Conditionality</b>	If the file size cannot be changed, then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.13.2.2.1 - Writing to a Stream Access File using Record Access</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.13.2.2.2 - Writing to a File with an Invalid Starting Position</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Test Directives</b>	
<b>135.1-2013 - 9.13.2.2.4 - Writing to a Nonexistent File</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

---

## **3.62 Staging Object**

---

### **3.62.1 Base Requirements**

Contact BTL for interim tests for this object

---

## **3.63 Audit Reporter Object**

---

### **3.63.1 Base Requirements**

Contact BTL for interim tests for this object

---

## **3.64 Audit Log Object**

---

### **3.64.1 Base Requirements**

Contact BTL for interim tests for this object

---

## **4 Data Sharing BIBBs**

---



## 4.1 Data Sharing - ReadProperty - A

### 4.1.1 Base Requirements

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

For all of the DS-RP-A tests, the vendor must supply information on how to make the IUT generate the specified requests.

As with all A-side BIBBs, the IUT must be able to initiate the service against any BACnet device.

There are no base requirements tests for this section.

### 4.1.2 Can Read Non-Array Properties

The IUT is able to read non-array properties from another BACnet device.

135.1-2013 - 8.18.1 - Reading Non-Array Properties		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 4.1.3 Can Read Array Elements

The IUT is able to read array elements from another BACnet device.

135.1-2013 - 8.18.2 - Reading an Array Element		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 4.1.4 Can Read the Size of an Array

The IUT is able to read the 0<sup>th</sup> element of an array property to determine the number of elements in the array.

135.1-2013 - 8.18.5 - Reading an Array Size		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 4.1.5 Can Read Whole Arrays

The IUT is able to read complete array properties without the use of array indices.

135.1-2013 - 8.18.4 - Reading Whole Array Properties		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 4.1.6 Can Read List Properties

The IUT is able to read list properties.

135.1-2013 - 8.18.1 - Reading Non-Array Properties		
	Test Conditionality	This test can be skipped if 8.18.2 is executed against a property that is an array of lists.
	Test Directives	The property that the IUT reads shall be a list property.
	Testing Hints	
135.1-2013 - 8.18.2 - Reading an Array Element		
	Test Conditionality	This test can be skipped if 8.18.1 is executed against a list property.

<b>Test Directives</b>	The property that the IUT reads shall be an array of lists.
<b>Testing Hints</b>	

#### 4.1.7 Can Read NULL Property Values

The IUT is able to read NULL values.

135.1-2013 - 8.18.1 - Reading Non-Array Properties		
	<b>Test Conditionality</b>	This test can be skipped if 8.18.2 is executed against a property that contains a NULL value.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	A non-array properties that might contain a NULL value are the Present_Value and Schedule_Default properties of a Schedule Object, and Low_Diff_Limit in the Loop Object.
135.1-2013 - 8.18.2 - Reading an Array Element		
	<b>Test Conditionality</b>	This test can be skipped if 8.18.1 is executed against a property that contains a NULL value.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	Array properties that might contain a NULL value are the Alarm_Values and Fault_Values of the CharacterString Value Object.

#### 4.1.8 Can Read BOOLEAN Property Values

The IUT is able to read BOOLEAN values.

135.1-2013 - 8.18.1 - Reading Non-Array Properties		
	<b>Test Conditionality</b>	This test can be skipped if 8.18.2 is executed against a property that contains a BOOLEAN value.
	<b>Test Directives</b>	The property that the IUT reads shall contain a BOOLEAN value.
	<b>Testing Hints</b>	
135.1-2013 - 8.18.2 - Reading an Array Element		
	<b>Test Conditionality</b>	This test can be skipped if 8.18.1 is executed against a property that contains a BOOLEAN value.
	<b>Test Directives</b>	The property that the IUT reads shall contain a BOOLEAN value.
	<b>Testing Hints</b>	

#### 4.1.9 Can Read Enumerated Property Values

The IUT is able to read Enumerated values.

135.1-2013 - 8.18.1 - Reading Non-Array Properties		
	<b>Test Conditionality</b>	This test can be skipped if 8.18.2 is executed against a property that contains an Enumerated value.
	<b>Test Directives</b>	The property that the IUT reads shall contain an Enumerated value.
	<b>Testing Hints</b>	
135.1-2013 - 8.18.2 - Reading an Array Element		
	<b>Test Conditionality</b>	This test can be skipped if 8.18.1 is executed against a property that contains an Enumerated value.
	<b>Test Directives</b>	The property that the IUT reads shall contain an Enumerated value.
	<b>Testing Hints</b>	

#### 4.1.10 Can Read INTEGER Property Values

The IUT is able to read INTEGER values.

135.1-2013 - 8.18.1 - Reading Non-Array Properties
--

	<b>Test Conditionality</b>	This test can be skipped if 8.18.2 is executed against a property that contains an INTEGER value.
	<b>Test Directives</b>	The property that the IUT reads shall contain an INTEGER value.
	<b>Testing Hints</b>	
<b>135.1-2013 - 8.18.2 - Reading an Array Element</b>		
	<b>Test Conditionality</b>	This test can be skipped if 8.18.1 is executed against a property that contains an INTEGER value.
	<b>Test Directives</b>	The property that the IUT reads shall contain an INTEGER value.
	<b>Testing Hints</b>	

#### 4.1.11 Can Read Unsigned Property Values

The IUT is able to read Unsigned values.

<b>135.1-2013 - 8.18.1 - Reading Non-Array Properties</b>		
	<b>Test Conditionality</b>	This test can be skipped if 8.18.2 is executed against a property that contains an Unsigned value.
	<b>Test Directives</b>	The property that the IUT reads shall contain an Unsigned value.
	<b>Testing Hints</b>	
<b>135.1-2013 - 8.18.2 - Reading an Array Element</b>		
	<b>Test Conditionality</b>	This test can be skipped if 8.18.1 is executed against a property that contains an Unsigned value.
	<b>Test Directives</b>	The property that the IUT reads shall contain an Unsigned value.
	<b>Testing Hints</b>	

#### 4.1.12 Can Read REAL Property Values

The IUT is able to read REAL values.

<b>135.1-2013 - 8.18.1 - Reading Non-Array Properties</b>		
	<b>Test Conditionality</b>	This test can be skipped if 8.18.2 is executed against a property that contains a REAL value.
	<b>Test Directives</b>	The property that the IUT reads shall contain a REAL value.
	<b>Testing Hints</b>	
<b>135.1-2013 - 8.18.2 - Reading an Array Element</b>		
	<b>Test Conditionality</b>	This test can be skipped if 8.18.1 is executed against a property that contains a REAL value.
	<b>Test Directives</b>	The property that the IUT reads shall contain a REAL value.
	<b>Testing Hints</b>	

#### 4.1.13 Can Read Double Property Values

The IUT is able to read Double values.

<b>135.1-2013 - 8.18.1 - Reading Non-Array Properties</b>		
	<b>Test Conditionality</b>	This test can be skipped if 8.18.2 is executed against a property that contains a Double value.
	<b>Test Directives</b>	The property that the IUT reads shall contain a Double value.
	<b>Testing Hints</b>	
<b>135.1-2013 - 8.18.2 - Reading an Array Element</b>		
	<b>Test Conditionality</b>	This test can be skipped if 8.18.1 is executed against a property that contains a Double value.
	<b>Test Directives</b>	The property that the IUT reads shall contain a Double value.
	<b>Testing Hints</b>	

#### 4.1.14 Can Read Time Property Values

The IUT is able to read Time values.

135.1-2013 - 8.18.1 - Reading Non-Array Properties		
	Test Conditionality	This test can be skipped if 8.18.2 is executed against a property that contains a Time value.
	Test Directives	The property that the IUT reads shall contain a Time value.
	Testing Hints	
135.1-2013 - 8.18.2 - Reading an Array Element		
	Test Conditionality	This test can be skipped if 8.18.1 is executed against a property that contains a Time value.
	Test Directives	The property that the IUT reads shall contain a Time value.
	Testing Hints	

#### 4.1.15 Can Read Date Property Values

The IUT is able to read Date values.

135.1-2013 - 8.18.1 - Reading Non-Array Properties		
	Test Conditionality	This test can be skipped if 8.18.2 is executed against a property that contains a Date value.
	Test Directives	The property that the IUT reads shall contain a Date value.
	Testing Hints	
135.1-2013 - 8.18.2 - Reading an Array Element		
	Test Conditionality	This test can be skipped if 8.18.1 is executed against a property that contains a Date value.
	Test Directives	The property that the IUT reads shall contain a Date value.
	Testing Hints	

#### 4.1.16 Can Read Character String Property Values

The IUT is able to read Character String values.

135.1-2013 - 8.18.1 - Reading Non-Array Properties		
	Test Conditionality	This test can be skipped if 8.18.2 is executed against a property that contains a Character String value.
	Test Directives	The property that the IUT reads shall contain a Character String value.
	Testing Hints	
135.1-2013 - 8.18.2 - Reading an Array Element		
	Test Conditionality	This test can be skipped if 8.18.1 is executed against a property that contains a Character String value.
	Test Directives	The property that the IUT reads shall contain a Character String value.
	Testing Hints	

#### 4.1.17 Can Read Octet String Property Values

The IUT is able to read Octet String values.

135.1-2013 - 8.18.1 - Reading Non-Array Properties		
	Test Conditionality	This test can be skipped if 8.18.2 is executed against a property that contains an Octet String value.
	Test Directives	The property that the IUT reads shall contain an Octet String value.
	Testing Hints	
135.1-2013 - 8.18.2 - Reading an Array Element		
	Test Conditionality	This test can be skipped if 8.18.1 is executed against a property that contains an Octet String value.
	Test Directives	
	Testing Hints	

	<b>Test Directives</b>	The property that the IUT reads shall contain an Octet String value.
	<b>Testing Hints</b>	

#### 4.1.18 Can Read Bit String Property Values

The IUT is able to read Bit String values.

135.1-2013 - 8.18.1 - Reading Non-Array Properties		
	<b>Test Conditionality</b>	This test can be skipped if 8.18.2 is executed against a property that contains a Bit String value.
	<b>Test Directives</b>	The property that the IUT reads shall contain a Bit String value.
	<b>Testing Hints</b>	
135.1-2013 - 8.18.2 - Reading an Array Element		
	<b>Test Conditionality</b>	This test can be skipped if 8.18.1 is executed against a property that contains a Bit String value.
	<b>Test Directives</b>	The property that the IUT reads shall contain a Bit String value.
	<b>Testing Hints</b>	

#### 4.1.19 Can Read BACnetObjectIdentifier Property Values

The IUT is able to read BACnetObjectIdentifier values.

135.1-2013 - 8.18.1 - Reading Non-Array Properties		
	<b>Test Conditionality</b>	This test can be skipped if 8.18.2 is executed against a property that contains a BACnetObjectIdentifier value.
	<b>Test Directives</b>	The property that the IUT reads shall contain a BACnetObjectIdentifier value.
	<b>Testing Hints</b>	
135.1-2013 - 8.18.2 - Reading an Array Element		
	<b>Test Conditionality</b>	This test can be skipped if 8.18.1 is executed against a property that contains a BACnetObjectIdentifier value.
	<b>Test Directives</b>	The property that the IUT reads shall contain a BACnetObjectIdentifier value.
	<b>Testing Hints</b>	

#### 4.1.20 Can Read Constructed Property Values

The IUT is able to read constructed property values.

135.1-2013 - 8.18.1 - Reading Non-Array Properties		
	<b>Test Conditionality</b>	This test can be skipped if 8.18.2 is executed against a property that contains a constructed value. This test shall be repeated for each standard constructed value that the IUT is able to read.
	<b>Test Directives</b>	The property that the IUT reads shall contain a constructed value.
	<b>Testing Hints</b>	
135.1-2013 - 8.18.2 - Reading an Array Element		
	<b>Test Conditionality</b>	This test can be skipped if 8.18.1 is executed against a property that contains a constructed value. This test shall be repeated for each standard constructed value that the IUT is able to read.
	<b>Test Directives</b>	The property that the IUT reads shall contain a constructed value.
	<b>Testing Hints</b>	

#### 4.1.21 Can Read Proprietary Property Values Of Basic Data Types

The IUT is able to read arbitrary proprietary properties with a data type of NULL, BOOLEAN, INTEGER, Unsigned, REAL, Double, Character String, Octet String, Time, Date, or BACnetObjectIdentifier.

135.1-2013 - 8.18.1 - Reading Non-Array Properties
--

# BTL Test Plan

	<b>Test Conditionality</b>	This test can be skipped if 8.18.2 is executed against a proprietary property.
	<b>Test Directives</b>	The property that the IUT reads shall be proprietary and not in a device manufactured by the same vendor.
	<b>Testing Hints</b>	
<b>135.1-2013 - 8.18.2 - Reading an Array Element</b>		
	<b>Test Conditionality</b>	This test can be skipped if 8.18.1 is executed against a proprietary property.
	<b>Test Directives</b>	The property that the IUT reads shall be proprietary and not in a device manufactured by the same vendor.
	<b>Testing Hints</b>	

## 4.2 Data Sharing - ReadProperty - B

### 4.2.1 Base Requirements

All devices must support this BIBB.

<b>BTL - 7.1.1 - Read Support Test Procedure</b>		
	<b>Test Conditionality</b>	Must be executed. To satisfy this test item, this test needs only be executed using ReadProperty.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.18.2.1 - Reading Non-Array Properties with an Array Index</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.18.2.3 - Reading an Unknown Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.18.2.4 - Reading an Unknown Property</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Be sure to test at least one property identifier that is within the ASHRAE allocated range for standard property identifiers, but that has not yet been defined.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.18.1.3 - Reading a Property From the Device Object using the Unknown Instance</b>		
	<b>Test Conditionality</b>	If the device implements protocol revision 4 or higher, this test must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.1.X3 - Verifying Property List against the EPICS</b>		
	<b>Test Conditionality</b>	Must be executed if the IUT claims Protocol Revision 14 or greater.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.18.1.X4 - Reading Array Properties at different Array Indexes</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Repeat for all supported BACnetARRAY properties
	<b>Testing Hints</b>	

### 4.2.2 Contains Enumerated Property Values

The IUT contains, or can be made to contain, a property with the value of Enumerated.

<b>BTL - 9.18.1.X1 - Reading Properties Based on Data Type</b>		
	<b>Test Conditionality</b>	If the IUT contains a property with the value Enumerated when test BTL - 7.1 is executed and the EPICS does not contain a ? for this property, this test can be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 4.2.3 Contains Unsigned Property Values

The IUT contains, or can be made to contain, a property with a data type of Unsigned.

BTL - 9.18.1.X1 - Reading Properties Based on Data Type		
	<b>Test Conditionality</b>	If the IUT contains a property with a Unsigned data type when test BTL - 7.1 is executed and the EPICS does not contain a ? for this property, this test can be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 4.2.4 Contains BACnetObjectIdentifier Property Values

The IUT contains, or can be made to contain, a property with a data type of BACnetObjectIdentifier.

BTL - 9.18.1.X1 - Reading Properties Based on Data Type		
	<b>Test Conditionality</b>	If the IUT contains a property with an BACnetObjectIdentifier data type when test BTL - 7.1 is executed and the EPICS does not contain a ? for this property, this test can be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 4.2.5 Contains Character String Property Values

The IUT contains, or can be made to contain, a property with a data type of Character String.

BTL - 9.18.1.X1 - Reading Properties Based on Data Type		
	<b>Test Conditionality</b>	If the IUT contains a property with a Character String data type when test BTL - 7.1 is executed and the EPICS does not contain a ? for this property, this test can be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 4.2.6 Contains Bit String Property Values

The IUT contains, or can be made to contain, a property with a data type of Bit String.

BTL - 9.18.1.X1 - Reading Properties Based on Data Type		
	<b>Test Conditionality</b>	If the IUT contains a property with a Bit String data type when test BTL - 7.1 is executed and the EPICS does not contain a ? for this property, this test can be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 4.2.7 Contains NULL Property Values

The IUT contains, or can be made to contain, a property with the value of NULL.

BTL - 9.18.1.X1 - Reading Properties Based on Data Type		
	<b>Test Conditionality</b>	If the IUT contains a property with the value NULL when test BTL - 7.1 is executed and the EPICS does not contain a ? for this property, this test can be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	Schedule_Default and Present_Value of Schedule Object, Alarm_Values and Fault_Values of the CharacterString Value Object and Low_Diff_Limit in the Loop Object are standard properties that contain NULL values.



### 4.2.8 Contains BOOLEAN Property Values

The IUT contains, or can be made to contain, a property with a data type of BOOLEAN.

BTL - 9.18.1.X1 - Reading Properties Based on Data Type		
	<b>Test Conditionality</b>	If the IUT contains a property with a BOOLEAN data type when test BTL - 7.1 is executed and the EPICS does not contain a ? for this property, this test can be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 4.2.9 Contains INTEGER Property Values

The IUT contains, or can be made to contain, a property with a data type of INTEGER.

BTL - 9.18.1.X1 - Reading Properties Based on Data Type		
	<b>Test Conditionality</b>	If the IUT contains a property with an INTEGER data type when test BTL - 7.1 is executed and the EPICS does not contain a ? for this property, this test can be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 4.2.10 Contains REAL Property Values

The IUT contains, or can be made to contain, a property with a data type of REAL.

BTL - 9.18.1.X1 - Reading Properties Based on Data Type		
	<b>Test Conditionality</b>	If the IUT contains a property with a REAL data type when test BTL - 7.1 is executed and the EPICS does not contain a ? for this property, this test can be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 4.2.11 Contains Double Property Values

The IUT contains, or can be made to contain, a property with a data type of Double.

BTL - 9.18.1.X1 - Reading Properties Based on Data Type		
	<b>Test Conditionality</b>	If the IUT contains a property with a Double data type when test BTL - 7.1 is executed and the EPICS does not contain a ? for this property, this test can be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 4.2.12 Contains Time Property Values

The IUT contains, or can be made to contain, a property with a data type of Time.

BTL - 9.18.1.X1 - Reading Properties Based on Data Type		
	<b>Test Conditionality</b>	If the IUT contains a property with a Time data type when test BTL - 7.1 is executed and the EPICS does not contain a ? for this property, this test can be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 4.2.13 Contains Date Property Values

The IUT contains, or can be made to contain, a property with a data type of Date.

<b>BTL - 9.18.1.X1 - Reading Properties Based on Data Type</b>		
	<b>Test Conditionality</b>	If the IUT contains a property with a Date data type when test BTL - 7.1 is executed and the EPICS does not contain a ? for this property, this test can be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.2.14 Contains Octet String Property Values

The IUT contains, or can be made to contain, a property with a data type of Octet String.

<b>BTL - 9.18.1.X1 - Reading Properties Based on Data Type</b>		
	<b>Test Conditionality</b>	If the IUT contains a property with an Octet String data type when test BTL - 7.1 is executed and the EPICS does not contain a ? for this property, this test can be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	A property that contains an Octet String is the Present_Value of an OctetString Value object.

#### 4.2.15 Contains Proprietary Properties with Basic Data Types

The IUT contains, or can be made to contain, a proprietary property with a data type of NULL, BOOLEAN, INTEGER, Unsigned, REAL, Double, Character String, Octet String, Time, Date, or BACnetObjectIdentifier.

<b>BTL - 9.18.1.X1 - Reading Properties Based on Data Type</b>		
	<b>Test Conditionality</b>	This test shall be skipped until such time as the test tools support interaction with proprietary properties.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

## 4.3 Data Sharing - ReadPropertyMultiple - A

### 4.3.1 Base Requirements

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

For all of the DS-RPM-A tests, the vendor must supply information on how to make the IUT generate the specified requests.

As with all A side tests, the IUT must be able to initiate the service against any BACnet device.

There are no base requirements tests for this section.

### 4.3.2 Supports DS-RP-A

The IUT supports DS-RP-A.

Verify Checklist		
	Test Conditionality	Must be executed.
	Test Directives	Verify that the IUT claims conformance to DS-RP-A.
	Testing Hints	

### 4.3.3 Can Read Multiple Properties from a Single Object

The IUT is able to read multiple properties from a single object in a single ReadPropertyMultiple request.

135.1-2013 - 8.20.2 - Reading Multiple Properties of a Single Object		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 4.3.4 Can Read a Single Property from Multiple Objects

The IUT is able to read a single property from multiple objects in a single ReadPropertyMultiple request. The same property need not be read from each object.

135.1-2013 - 8.20.3 - Reading Multiple Objects, One Property Each		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 4.3.5 Can Read Multiple Properties from Multiple Objects

The IUT is able to read multiple properties from multiple objects in a single ReadPropertyMultiple request.

135.1-2013 - 8.20.4 - Reading Multiple Objects, Multiple Properties for Each		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 4.3.6 Can Read Using the Special ALL Property Identifier

The IUT is able to read the special ALL property.

135.1-2013 - 8.20.1 - Reading a Single Property of a Single Object, 135.1-2013 - 8.20.2 - Reading Multiple Properties of a Single Object, 135.1-2013 - 8.20.3 - Reading Multiple Objects, One Property Each, or 135.1-2013 - 8.20.4 - Reading Multiple Objects, Multiple Properties for Each		
	Test Conditionality	At least one of the tests (8.20.1..8.20.4) shall be executed with property ALL.

<b>Test Directives</b>	At least one of the properties read by the selected test shall be the special ALL property.
<b>Testing Hints</b>	

### 4.3.7 Can Read Using the Special OPTIONAL Property Identifier

The IUT is able to read the special OPTIONAL property.

<b>135.1-2013 - 8.20.1 - Reading a Single Property of a Single Object,  135.1-2013 - 8.20.2 - Reading Multiple Properties of a Single Object,  135.1-2013 - 8.20.3 - Reading Multiple Objects, One Property Each, or  135.1-2013 - 8.20.4 - Reading Multiple Objects, Multiple Properties for Each</b>	
<b>Test Conditionality</b>	At least one of the tests (8.20.1..8.20.4) shall be executed with property OPTIONAL.
<b>Test Directives</b>	At least one of the properties read by the selected test shall be the special OPTIONAL property.
<b>Testing Hints</b>	

### 4.3.8 Can Read Using the Special REQUIRED Property Identifier

The IUT is able to read the special REQUIRED property.

<b>135.1-2013 - 8.20.1 - Reading a Single Property of a Single Object,  135.1-2013 - 8.20.2 - Reading Multiple Properties of a Single Object,  135.1-2013 - 8.20.3 - Reading Multiple Objects, One Property Each, or  135.1-2013 - 8.20.4 - Reading Multiple Objects, Multiple Properties for Each</b>	
<b>Test Conditionality</b>	At least one of the tests (8.20.1..8.20.4) shall be executed with property REQUIRED.
<b>Test Directives</b>	At least one of the properties read by the selected test shall be the special REQUIRED property.
<b>Testing Hints</b>	

### 4.3.9 Can Fallback to ReadProperty Based on Protocol\_Services\_Supported

The IUT is able to select between initiating ReadPropertyMultiple and ReadProperty requests based on whether or not support for ReadPropertyMultiple is advertised by a peer device's Protocol\_Services\_Supported property.

<b>BTL - 8.20.5.1 - The IUT Determines the TD does not Support the ReadPropertyMultiple Service</b>	
<b>Test Conditionality</b>	Must be executed.
<b>Test Directives</b>	
<b>Testing Hints</b>	

### 4.3.10 Can fallback to ReadProperty upon receipt of UNRECOGNIZED\_SERVICE

The IUT is able to read property values with the ReadProperty service when it receives a Reject-PDU with a Reject Reason of UNRECOGNIZED\_SERVICE in response to a ReadPropertyMultiple request.

<b>135.1-2013 - 8.20.5.2 - The IUT Automatically Sends ReadProperty Requests when the TD Returns a Reject - UNRECOGNIZED_SERVICE Response</b>	
<b>Test Conditionality</b>	Must be executed.
<b>Test Directives</b>	
<b>Testing Hints</b>	

### 4.3.11 Can Read a Single Property from a Single Object

The IUT is able to read a single property from a single object.

135.1-2013 - 8.20.1 - Reading a Single Property of a Single Object		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 4.3.12 Can Read Non-Array Properties

The IUT is able to read non-array properties from another BACnet device.

135.1-2013 - 8.20.1 - Reading a Single Property of a Single Object, 135.1-2013 - 8.20.2 - Reading Multiple Properties of a Single Object, 135.1-2013 - 8.20.3 - Reading Multiple Objects, One Property Each, or 135.1-2013 - 8.20.4 - Reading Multiple Objects, Multiple Properties for Each		
	Test Conditionality	At least one of the tests (8.20.1..8.20.4) shall be executed against a non-array property.
	Test Directives	At least one of the properties read by the selected test shall be a non-array property
	Testing Hints	

### 4.3.13 Can Read Array Elements

The IUT is able to read array elements from another BACnet device.

135.1-2013 - 8.20.1 - Reading a Single Property of a Single Object, 135.1-2013 - 8.20.2 - Reading Multiple Properties of a Single Object, 135.1-2013 - 8.20.3 - Reading Multiple Objects, One Property Each, or 135.1-2013 - 8.20.4 - Reading Multiple Objects, Multiple Properties for Each		
	Test Conditionality	At least one of the tests (8.20.1..8.20.4) shall be executed against an array element.
	Test Directives	At least one of the properties read by the selected test shall be an array element. This will require the inclusion of an array index in the property specification even though the test definition does not show one.
	Testing Hints	

### 4.3.14 Can Read the Size of an Array

The IUT is able to read the 0<sup>th</sup> element of an array property to determine the number of elements in the array.

135.1-2013 - 8.20.1 - Reading a Single Property of a Single Object, 135.1-2013 - 8.20.2 - Reading Multiple Properties of a Single Object, 135.1-2013 - 8.20.3 - Reading Multiple Objects, One Property Each, or 135.1-2013 - 8.20.4 - Reading Multiple Objects, Multiple Properties for Each		
	Test Conditionality	At least one of the tests (8.20.1..8.20.4) shall be executed against the 0 <sup>th</sup> element of an array.
	Test Directives	At least one of the properties read by the selected test shall be the 0 <sup>th</sup> element of an array.
	Testing Hints	

### 4.3.15 Can Read Whole Arrays

The IUT is able to read complete array properties without the use of array indices.

<b>135.1-2013 - 8.20.1 - Reading a Single Property of a Single Object,  135.1-2013 - 8.20.2 - Reading Multiple Properties of a Single Object,  135.1-2013 - 8.20.3 - Reading Multiple Objects, One Property Each, or  135.1-2013 - 8.20.4 - Reading Multiple Objects, Multiple Properties for Each</b>		
	<b>Test Conditionality</b>	At least one of the tests (8.20.1..8.20.4) shall be executed against an array property without the use of an array index.
	<b>Test Directives</b>	At least one of the properties read by the selected test shall be array property without an array element specified.
	<b>Testing Hints</b>	

### 4.3.16 Can Read List Properties

The IUT is able to read list properties.

<b>135.1-2013 - 8.20.1 - Reading a Single Property of a Single Object,  135.1-2013 - 8.20.2 - Reading Multiple Properties of a Single Object,  135.1-2013 - 8.20.3 - Reading Multiple Objects, One Property Each, or  135.1-2013 - 8.20.4 - Reading Multiple Objects, Multiple Properties for Each</b>		
	<b>Test Conditionality</b>	At least one of the tests (8.20.1..8.20.4) shall be executed against a list property.
	<b>Test Directives</b>	At least one of the properties read by the selected test shall be a list property.
	<b>Testing Hints</b>	

### 4.3.17 Can Read NULL Property Values

The IUT is able to read NULL values.

<b>135.1-2013 - 8.20.1 - Reading a Single Property of a Single Object,  135.1-2013 - 8.20.2 - Reading Multiple Properties of a Single Object,  135.1-2013 - 8.20.3 - Reading Multiple Objects, One Property Each, or  135.1-2013 - 8.20.4 - Reading Multiple Objects, Multiple Properties for Each</b>		
	<b>Test Conditionality</b>	At least one of the tests (8.20.1..8.20.4) shall be executed against a property with a NULL value.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	Schedule_Default and Present_Value of the Schedule Object, Alarm_Values and Fault_Values of the CharacterString Value Object and Low_Diff_Limit in the Loop Object are standard properties that contain NULL Values.

### 4.3.18 Can Read BOOLEAN Property Values

The IUT is able to read BOOLEAN values.

<b>135.1-2013 - 8.20.1 - Reading a Single Property of a Single Object,  135.1-2013 - 8.20.2 - Reading Multiple Properties of a Single Object,  135.1-2013 - 8.20.3 - Reading Multiple Objects, One Property Each, or  135.1-2013 - 8.20.4 - Reading Multiple Objects, Multiple Properties for Each</b>		
	<b>Test Conditionality</b>	At least one of the tests (8.20.1..8.20.4) shall be executed against a property with a BOOLEAN value.
	<b>Test Directives</b>	At least one of the properties read by the selected test shall contain a BOOLEAN value.
	<b>Testing Hints</b>	

### 4.3.19 Can Read Enumerated Property Values

The IUT is able to read Enumerated values.

<b>135.1-2013 - 8.20.1 - Reading a Single Property of a Single Object,  135.1-2013 - 8.20.2 - Reading Multiple Properties of a Single Object,  135.1-2013 - 8.20.3 - Reading Multiple Objects, One Property Each, or  135.1-2013 - 8.20.4 - Reading Multiple Objects, Multiple Properties for Each</b>		
	<b>Test Conditionality</b>	At least one of the tests (8.20.1..8.20.4) shall be executed against a property with an Enumerated value.
	<b>Test Directives</b>	At least one of the properties read by the selected test shall contain an Enumerated value.
	<b>Testing Hints</b>	

### 4.3.20 Can Read INTEGER Property Values

The IUT is able to read INTEGER values.

<b>135.1-2013 - 8.20.1 - Reading a Single Property of a Single Object,  135.1-2013 - 8.20.2 - Reading Multiple Properties of a Single Object,  135.1-2013 - 8.20.3 - Reading Multiple Objects, One Property Each, or  135.1-2013 - 8.20.4 - Reading Multiple Objects, Multiple Properties for Each</b>		
	<b>Test Conditionality</b>	At least one of the tests (8.20.1..8.20.4) shall be executed against a property with an INTEGER value.
	<b>Test Directives</b>	At least one of the properties read by the selected test shall contain an INTEGER value.
	<b>Testing Hints</b>	

### 4.3.21 Can Read Unsigned Property Values

The IUT is able to read Unsigned values.

<b>135.1-2013 - 8.20.1 - Reading a Single Property of a Single Object,  135.1-2013 - 8.20.2 - Reading Multiple Properties of a Single Object,  135.1-2013 - 8.20.3 - Reading Multiple Objects, One Property Each, or  135.1-2013 - 8.20.4 - Reading Multiple Objects, Multiple Properties for Each</b>		
	<b>Test Conditionality</b>	At least one of the tests (8.20.1..8.20.4) shall be executed against a property with an Unsigned value.
	<b>Test Directives</b>	At least one of the properties read by the selected test shall contain an Unsigned value.
	<b>Testing Hints</b>	

### 4.3.22 Can Read REAL Property Values

The IUT is able to read REAL values.

<b>135.1-2013 - 8.20.1 - Reading a Single Property of a Single Object,  135.1-2013 - 8.20.2 - Reading Multiple Properties of a Single Object,  135.1-2013 - 8.20.3 - Reading Multiple Objects, One Property Each, or  135.1-2013 - 8.20.4 - Reading Multiple Objects, Multiple Properties for Each</b>		
	<b>Test Conditionality</b>	At least one of the tests (8.20.1..8.20.4) shall be executed against a property with a REAL value.
	<b>Test Directives</b>	At least one of the properties read by the selected test shall contain a REAL value.
	<b>Testing Hints</b>	

### 4.3.23 Can Read Double Property Values

The IUT is able to read Double values.

<b>135.1-2013 - 8.20.1 - Reading a Single Property of a Single Object,  135.1-2013 - 8.20.2 - Reading Multiple Properties of a Single Object,  135.1-2013 - 8.20.3 - Reading Multiple Objects, One Property Each, or  135.1-2013 - 8.20.4 - Reading Multiple Objects, Multiple Properties for Each</b>		
	<b>Test Conditionality</b>	At least one of the tests (8.20.1..8.20.4) shall be executed against a property with a Double value.
	<b>Test Directives</b>	At least one of the properties read by the selected test shall contain a Double value.
	<b>Testing Hints</b>	

#### 4.3.24 Can Read Time Property Values

The IUT is able to read Time values.

<b>135.1-2013 - 8.20.1 - Reading a Single Property of a Single Object,  135.1-2013 - 8.20.2 - Reading Multiple Properties of a Single Object,  135.1-2013 - 8.20.3 - Reading Multiple Objects, One Property Each, or  135.1-2013 - 8.20.4 - Reading Multiple Objects, Multiple Properties for Each</b>		
	<b>Test Conditionality</b>	At least one of the tests (8.20.1..8.20.4) shall be executed against a property with a Time value.
	<b>Test Directives</b>	At least one of the properties read by the selected test shall contain a Time value.
	<b>Testing Hints</b>	

#### 4.3.25 Can Read Date Property Values

The IUT is able to read Date values.

<b>135.1-2013 - 8.20.1 - Reading a Single Property of a Single Object,  135.1-2013 - 8.20.2 - Reading Multiple Properties of a Single Object,  135.1-2013 - 8.20.3 - Reading Multiple Objects, One Property Each, or  135.1-2013 - 8.20.4 - Reading Multiple Objects, Multiple Properties for Each</b>		
	<b>Test Conditionality</b>	At least one of the tests (8.20.1..8.20.4) shall be executed against a property with a Date value.
	<b>Test Directives</b>	At least one of the properties read by the selected test shall contain a Date value.
	<b>Testing Hints</b>	

#### 4.3.26 Can Read Character String Property Values

The IUT is able to read Character String values.

<b>135.1-2013 - 8.20.1 - Reading a Single Property of a Single Object,  135.1-2013 - 8.20.2 - Reading Multiple Properties of a Single Object,  135.1-2013 - 8.20.3 - Reading Multiple Objects, One Property Each, or  135.1-2013 - 8.20.4 - Reading Multiple Objects, Multiple Properties for Each</b>		
	<b>Test Conditionality</b>	At least one of the tests (8.20.1..8.20.4) shall be executed against a property with a Character String value.
	<b>Test Directives</b>	At least one of the properties read by the selected test shall contain a Character String value.
	<b>Testing Hints</b>	

#### 4.3.27 Can Read Octet String Property Values

The IUT is able to read Octet String values.



<b>135.1-2013 - 8.20.1 - Reading a Single Property of a Single Object,  135.1-2013 - 8.20.2 - Reading Multiple Properties of a Single Object,  135.1-2013 - 8.20.3 - Reading Multiple Objects, One Property Each, or  135.1-2013 - 8.20.4 - Reading Multiple Objects, Multiple Properties for Each</b>		
	<b>Test Conditionality</b>	At least one of the tests (8.20.1..8.20.4) shall be executed against a property with an Octet String value.
	<b>Test Directives</b>	At least one of the properties read by the selected test shall contain an Octet String value.
	<b>Testing Hints</b>	

### 4.3.28 Can Read Bit String Property Values

The IUT is able to read Bit String values.

<b>135.1-2013 - 8.20.1 - Reading a Single Property of a Single Object,  135.1-2013 - 8.20.2 - Reading Multiple Properties of a Single Object,  135.1-2013 - 8.20.3 - Reading Multiple Objects, One Property Each, or  135.1-2013 - 8.20.4 - Reading Multiple Objects, Multiple Properties for Each</b>		
	<b>Test Conditionality</b>	At least one of the tests (8.20.1..8.20.4) shall be executed against a property with a Bit String value.
	<b>Test Directives</b>	At least one of the properties read by the selected test shall contain a Bit String value.
	<b>Testing Hints</b>	

### 4.3.29 Can Read BACnetObjectIdentifier Property Values

The IUT is able to read BACnetObjectIdentifier values.

<b>135.1-2013 - 8.20.1 - Reading a Single Property of a Single Object,  135.1-2013 - 8.20.2 - Reading Multiple Properties of a Single Object,  135.1-2013 - 8.20.3 - Reading Multiple Objects, One Property Each, or  135.1-2013 - 8.20.4 - Reading Multiple Objects, Multiple Properties for Each</b>		
	<b>Test Conditionality</b>	At least one of the tests (8.20.1..8.20.4) shall be executed against a property with a BACnetObjectIdentifier value.
	<b>Test Directives</b>	At least one of the properties read by the selected test shall contain a BACnetObjectIdentifier value.
	<b>Testing Hints</b>	

### 4.3.30 Can Read Constructed Property Values

The IUT is able to read constructed property values.

<b>135.1-2013 - 8.20.1 - Reading a Single Property of a Single Object,  135.1-2013 - 8.20.2 - Reading Multiple Properties of a Single Object,  135.1-2013 - 8.20.3 - Reading Multiple Objects, One Property Each, or  135.1-2013 - 8.20.4 - Reading Multiple Objects, Multiple Properties for Each</b>		
	<b>Test Conditionality</b>	At least one of the tests (8.20.1..8.20.4) shall be executed against a property with a constructed value. This test shall be repeated for each standard constructed value that the IUT is able to read.
	<b>Test Directives</b>	At least one of the properties read by the selected test shall contain a constructed value.
	<b>Testing Hints</b>	

### 4.3.31 Can Read Proprietary Property Values Of Basic Data Types

The IUT is able to read arbitrary proprietary properties with a data type of NULL, BOOLEAN, INTEGER, Unsigned, REAL, Double, Character String, Octet String, Time, Date, or BACnetObjectIdentifier.

<b>135.1-2013 - 8.20.1 - Reading a Single Property of a Single Object,  135.1-2013 - 8.20.2 - Reading Multiple Properties of a Single Object,  135.1-2013 - 8.20.3 - Reading Multiple Objects, One Property Each, or  135.1-2013 - 8.20.4 - Reading Multiple Objects, Multiple Properties for Each</b>		
	<b>Test Conditionality</b>	At least one of the tests (8.20.1..8.20.4) shall be executed against a proprietary property.
	<b>Test Directives</b>	At least one of the properties read by the selected test shall be proprietary.
	<b>Testing Hints</b>	

## 4.4 Data Sharing - ReadPropertyMultiple - B

### 4.4.1 Base Requirements

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

<b>BTL - 7.1.1 - Read Support Test Procedure</b>		
	<b>Test Conditionality</b>	Must be executed. To satisfy this test item, test 7.1 need only be executed using ReadPropertyMultiple.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.20.1.1 - Reading a Single Property from a Single Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.20.1.2 - Reading Multiple properties from a Single Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.20.1.3 - Reading a Single Property from Multiple Objects</b>		
	<b>Test Conditionality</b>	This test can be skipped if the IUT cannot be made to contain more than 1 object.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.20.1.4 - Reading Multiple Properties from Multiple Objects</b>		
	<b>Test Conditionality</b>	This test can be skipped if the IUT cannot be made to contain more than 1 object.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.20.1.5 - Reading Multiple Properties with a Single Embedded Access Error</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.20.1.6 - Reading Multiple Properties with Multiple Embedded Access Errors</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.20.1.7 - Reading ALL Properties</b>		
	<b>Test Conditionality</b>	Must be executed. This test shall be skipped for any object type whose set of properties cannot be transmitted in the largest supported response message based on the IUT's APDU and segmentation limitations.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	The pre-tester should apply this test to every object type. If the set of properties differs between instances of the same object type in the IUT, each form of the object type should be tested.
<b>BTL - 9.20.1.8 - Reading OPTIONAL Properties</b>		
	<b>Test Conditionality</b>	Must be executed. This test shall be skipped for any object type whose set of optional properties cannot be transmitted in the largest supported response message based on the IUT's APDU and segmentation limitations.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	The pre-tester should apply this test to every object type.

		If the set of properties differs between instances of the same object type in the IUT, each form of the object type should be tested.
<b>BTL - 9.20.1.9 - Reading REQUIRED Properties</b>		
	<b>Test Conditionality</b>	Must be executed. This test shall be skipped for any object type whose set of optional properties cannot be transmitted in the largest supported response message based on the IUT's APDU and segmentation limitations.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	The pre-tester should apply this test to every object type. If the set of properties differs between instances of the same object type in the IUT, each form of the object type should be tested.
<b>135.1-2013 - 9.20.1.10 - Reading the Size of an Array</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.20.2.1 - Reading a Single, Unsupported Property from a Single Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.20.2.2 - Reading Multiple Properties with Access Errors for Every Property</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.20.2.3 - Reading Non-Array Properties with an Array Index</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.20.1.11 - Reading a Property From the Device Object using the Unknown Instance</b>		
	<b>Test Conditionality</b>	If the device implements protocol revision 4 or higher, this test must be executed. If the device does not support ReadPropertyMultiple, this test may be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.20.1.X2 - ReadPropertyMultiple Array Properties</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Repeat for all supported BACnetARRAY properties
	<b>Testing Hints</b>	

#### 4.4.2 Contains Enumerated Property Values

The IUT contains, or can be made to contain, a property with the value of Enumerated.

<b>BTL - 9.20.1.X1 - Reading Properties Based on Data Type</b>		
	<b>Test Conditionality</b>	If the IUT contains a property with the value Enumerated when test BTL - 7.1 is executed and the EPICS does not contain a ? for this property, this test can be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.4.3 Contains Unsigned Property Values

The IUT contains, or can be made to contain, a property with a data type of Unsigned.

<b>BTL - 9.20.1.X1 - Reading Properties Based on Data Type</b>		
--	--	--

	<b>Test Conditionality</b>	If the IUT contains a property with a Unsigned data type when test BTL - 7.1 is executed and the EPICS does not contain a ? for this property, this test can be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.4.4 Contains BACnetObjectIdentifier Property Values

The IUT contains, or can be made to contain, a property with a data type of BACnetObjectIdentifier.

<b>BTL - 9.20.1.X1 - Reading Properties Based on Data Type</b>		
	<b>Test Conditionality</b>	If the IUT contains a property with an BACnetObjectIdentifier data type when test BTL - 7.1 is executed and the EPICS does not contain a ? for this property, this test can be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.4.5 Contains Character String Property Values

The IUT contains, or can be made to contain, a property with a data type of Character String.

<b>BTL - 9.20.1.X1 - Reading Properties Based on Data Type</b>		
	<b>Test Conditionality</b>	If the IUT contains a property with a Character String data type when test BTL - 7.1 is executed and the EPICS does not contain a ? for this property, this test can be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.4.6 Contains Bit String Property Values

The IUT contains, or can be made to contain, a property with a data type of Bit String.

<b>BTL - 9.20.1.X1 - Reading Properties Based on Data Type</b>		
	<b>Test Conditionality</b>	If the IUT contains a property with a Bit String data type when test BTL - 7.1 is executed and the EPICS does not contain a ? for this property, this test can be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.4.7 Contains NULL Property Values

The IUT contains, or can be made to contain, a property with the value of NULL.

<b>BTL - 9.20.1.X1 - Reading Properties Based on Data Type</b>		
	<b>Test Conditionality</b>	If the IUT contains a property with a NULL data type when test BTL - 7.1 is executed this test can be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	Schedule_Default and Present_Value of the Schedule Object, Alarm_Values and Fault_Values of the CharacterString Value Object and Low_Diff_Limit in the Loop Object are standard properties that contain NULL Values.

#### 4.4.8 Contains BOOLEAN Property Values

The IUT contains, or can be made to contain, a property with a data type of BOOLEAN.

<b>BTL - 9.20.1.X1 - Reading Properties Based on Data Type</b>		
	<b>Test Conditionality</b>	If the IUT contains a property with a BOOLEAN data type when test BTL - 7.1 is executed this test can be skipped.

	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.4.9 Contains INTEGER Property Values

The IUT contains, or can be made to contain, a property with a data type of INTEGER.

<b>BTL - 9.20.1.X1 - Reading Properties Based on Data Type</b>		
	<b>Test Conditionality</b>	If the IUT contains a property with an INTEGER data type when test BTL - 7.1 is executed this test can be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.4.10 Contains REAL Property Values

The IUT contains, or can be made to contain, a property with a data type of REAL.

<b>BTL - 9.20.1.X1 - Reading Properties Based on Data Type</b>		
	<b>Test Conditionality</b>	If the IUT contains a property with a REAL data type when test BTL - 7.1 is executed this test can be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.4.11 Contains Double Property Values

The IUT contains, or can be made to contain, a property with a data type of Double.

<b>BTL - 9.20.1.X1 - Reading Properties Based on Data Type</b>		
	<b>Test Conditionality</b>	If the IUT contains a property with a Double data type when test BTL - 7.1 is executed this test can be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.4.12 Contains Time Property Values

The IUT contains, or can be made to contain, a property with a data type of Time.

<b>BTL - 9.20.1.X1 - Reading Properties Based on Data Type</b>		
	<b>Test Conditionality</b>	If the IUT contains a property with a Time data type when test BTL - 7.1 is executed this test can be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.4.13 Contains Date Property Values

The IUT contains, or can be made to contain, a property with a data type of Date.

<b>BTL - 9.20.1.X1 - Reading Properties Based on Data Type</b>		
	<b>Test Conditionality</b>	If the IUT contains a property with a Date data type when test BTL - 7.1 is executed this test can be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.4.14 Contains Octet String Property Values

The IUT contains, or can be made to contain, a property with a data type of Octet String.

<b>BTL - 9.20.1.X1 - Reading Properties Based on Data Type</b>		
--	--	--

	<b>Test Conditionality</b>	If the IUT contains a property with an Octet String data type when test BTL - 7.1 is executed this test can be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.4.15 Contains Proprietary Properties with Basic Data Types

The IUT contains, or can be made to contain, a proprietary property with a data type of NULL, BOOLEAN, INTEGER, Unsigned, REAL, Double, Character String, Bit String, Octet String, Time, Date, or BACnetObjectIdentifier.

<b>BTL - 9.20.1.X1 - Reading Properties Based on Data Type</b>		
	<b>Test Conditionality</b>	This test shall be skipped until such time as the test tools support interaction with proprietary properties.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

## 4.5 Data Sharing - WriteProperty - A

### 4.5.1 Base Requirements

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

For all of the DS-WP-A tests, the vendor must supply information on how to make the IUT generate the specified requests.

As with all A-side tests, the IUT must be able to initiate the service against any BACnet device.

There are no base requirements tests for this section.

### 4.5.2 Can Write Non-Array Properties

The IUT is able to write values to properties that are not arrays.

135.1-2013 - 8.22.1 - Writing Non-Array Properties		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 4.5.3 Can Write Array Elements

The IUT is able to write values to the elements of an array individually.

135.1-2013 - 8.22.2 - Writing Array Properties		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 4.5.4 Can Write Whole Arrays

The IUT is able to write whole array properties without an array index.

BTL - 8.22.X4 - Writing Array Properties as a Whole Array		
	Test Conditionality	Must be executed.
	Test Directives	If possible, for resizable array propertier, repeat this test for 0, 1, and $n$ elements.
	Testing Hints	

### 4.5.5 Can Write Whole Lists

The IUT is able to write list values using WriteProperty.

135.1-2013 - 8.22.1 - Writing Non-Array Properties, or 135.1-2013 - 8.22.2 - Writing Array Properties		
	Test Conditionality	At least one of these tests must be applied with the IUT writing a list.
	Test Directives	The property chosen must be a list or an array of lists depending on the test used.
	Testing Hints	

### 4.5.6 Can Command and Relinquish With a Priority

The IUT is able to supply a priority parameter when writing to a property with WriteProperty as is required in order to command commandable properties at a priority other than 16.

135.1-2013 - 8.22.3 - Writing Commandable Properties		
	Test Conditionality	Must be applied twice, once with a NULL value and once with a non-NULL value.



	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 4.5.7 Can Write NULL Property Values to non-commandable Properties

The IUT is able to write NULL values to non-commandable properties.

135.1-2013 - 8.22.1 - Writing Non-Array Properties, or 135.1-2013 - 8.22.2 - Writing Array Properties		
	<b>Test Conditionality</b>	At least one of these tests must be applied with the IUT writing a NULL value.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	Schedule_Default and Present_Value of the Schedule Object, Alarm_Values and Fault_Values of the CharacterString Value Object and Low_Diff_Limit in the Loop Object are standard properties that should accept a written NULL.

### 4.5.8 Can Write the Size of an Array

The IUT is able to change the size of array properties by writing to the 0<sup>th</sup> element.

135.1-2013 - 8.22.6 - Writing An Array Size		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 4.5.9 Can Write BOOLEAN Property Values

The IUT is able to write BOOLEAN values to properties.

135.1-2013 - 8.22.1 - Writing Non-Array Properties, or 135.1-2013 - 8.22.2 - Writing Array Properties		
	<b>Test Conditionality</b>	At least one of these tests must be applied with the IUT writing a BOOLEAN value.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 4.5.10 Can Write Enumerated Property Values

The IUT is able to write Enumerated values to properties.

135.1-2013 - 8.22.1 - Writing Non-Array Properties, or 135.1-2013 - 8.22.2 - Writing Array Properties		
	<b>Test Conditionality</b>	At least one of these tests must be applied with the IUT writing an Enumerated value.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 4.5.11 Can Write INTEGER Property Values

The IUT is able to write INTEGER values to properties.

135.1-2013 - 8.22.1 - Writing Non-Array Properties, or 135.1-2013 - 8.22.2 - Writing Array Properties		
	<b>Test Conditionality</b>	At least one of these tests must be applied with the IUT writing an INTEGER value.

	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.5.12 Can Write Unsigned Property Values

The IUT is able to write Unsigned values to properties.

<b>135.1-2013 - 8.22.1 - Writing Non-Array Properties, or 135.1-2013 - 8.22.2 - Writing Array Properties</b>		
	<b>Test Conditionality</b>	At least one of these tests must be applied with the IUT writing an Unsigned value.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.5.13 Can Write REAL Property Values

The IUT is able to write REAL values to properties.

<b>135.1-2013 - 8.22.1 - Writing Non-Array Properties, or 135.1-2013 - 8.22.2 - Writing Array Properties</b>		
	<b>Test Conditionality</b>	At least one of these tests must be applied with the IUT writing a REAL value.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.5.14 Can Write Double Property Values

The IUT is able to write Double values to properties.

<b>135.1-2013 - 8.22.1 - Writing Non-Array Properties, or 135.1-2013 - 8.22.2 - Writing Array Properties</b>		
	<b>Test Conditionality</b>	At least one of these tests must be applied with the IUT writing a Double value.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.5.15 Can Write Time Property Values

The IUT is able to write Time values to properties.

<b>135.1-2013 - 8.22.1 - Writing Non-Array Properties, or 135.1-2013 - 8.22.2 - Writing Array Properties</b>		
	<b>Test Conditionality</b>	At least one of these tests must be applied with the IUT writing a Time value.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.5.16 Can Write Date Property Values

The IUT is able to write Date values to properties.

<b>135.1-2013 - 8.22.1 - Writing Non-Array Properties, or 135.1-2013 - 8.22.2 - Writing Array Properties</b>		
	<b>Test Conditionality</b>	At least one of these tests must be applied with the IUT writing a Date value.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.5.17 Can Write Character String Property Values

The IUT is able to write Character String values to properties.

135.1-2013 - 8.22.1 - Writing Non-Array Properties, or 135.1-2013 - 8.22.2 - Writing Array Properties		
	<b>Test Conditionality</b>	At least one of these tests must be applied with the IUT writing a Character String value.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.5.18 Can Write Octet String Property Values

The IUT is able to write Octet String values to properties.

135.1-2013 - 8.22.1 - Writing Non-Array Properties, or 135.1-2013 - 8.22.2 - Writing Array Properties		
	<b>Test Conditionality</b>	At least one of these tests must be applied with the IUT writing an Octet String value.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.5.19 Can Write Bit String Property Values

The IUT is able to write Bit String values to properties.

135.1-2013 - 8.22.1 - Writing Non-Array Properties, or 135.1-2013 - 8.22.2 - Writing Array Properties		
	<b>Test Conditionality</b>	At least one of these tests must be applied with the IUT writing a Bit String value.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.5.20 Can Write BACnetObjectIdentifier Property Values

The IUT is able to write BACnetObjectIdentifier values to properties.

135.1-2013 - 8.22.1 - Writing Non-Array Properties, or 135.1-2013 - 8.22.2 - Writing Array Properties		
	<b>Test Conditionality</b>	At least one of these tests must be applied with the IUT writing a BACnetObjectIdentifier value.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.5.21 Can Write Constructed Property Values

The IUT is able to write constructed values to properties. Constructed values are SEQUENCEs or CHOICEs of basic or other constructed datatypes.

135.1-2013 - 8.22.1 - Writing Non-Array Properties, or 135.1-2013 - 8.22.2 - Writing Array Properties		
	<b>Test Conditionality</b>	For each type of standard constructed datatypes that the IUT can write. At least one of these tests must be applied.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.5.22 Can Write Proprietary Property Values of Basic Data Types

The IUT is able to write basic datatype values to properties. The basic datatypes are NULL, BOOLEAN, Enumerated, INTEGER, Unsigned, REAL, Double, Date, Time, Character String, Octet String, Bit String, and BACnetObjectIdentifier.

135.1-2013 - 8.22.1 - Writing Non-Array Properties, or 135.1-2013 - 8.22.2 - Writing Array Properties		
	<b>Test Conditionality</b>	At least one of these tests must be applied with the IUT writing any basic datatype value to a proprietary property.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

## 4.6 Data Sharing - WriteProperty - B

### 4.6.1 Base Requirements

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

<b>BTL - 7.2.2 - Write Support Test Procedure</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	For each property on which 9.22.1.X2 is executed on each writable datatype, that property can be skipped in the performance of 7.2.2. To satisfy this test item, test 7.2.2, need only be executed using WriteProperty.
	<b>Testing Hints</b>	
<b>BTL - 9.22.1.3 - Writing a Non-Commandable Property With a Priority</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.22.2.3 - Writing with a Property Value Having the Wrong Datatype</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	The pre-tester may want to consider running this test on a number of different properties with differing datatypes as this is a commonly failed test at the BTL.
<b>BTL - 9.22.2.4 - Writing with a Property Value that is Out of Range</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.22.2.X1 - Writing Non-Array Read-only Property with an Array Index</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>Verify Checklist</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that if the IUT is capable of executing AddListElement and/or RemoveListElement, that the DS-WP-B option “Contains Writable List Properties” is claimed.
	<b>Testing Hints</b>	
<b>BTL - 9.22.2.X2 - Resizing a writable fixed size array property</b>		
	<b>Test Conditionality</b>	If IUT does not contain a writable fixed size array property, then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	For example, Weekly_Schedule.

### 4.6.2 Contains Writable Non-Array Properties

The IUT contains, or can be made to contain, a writable property that is not an array.

<b>BTL - 9.22.2.1 - Writing Non-Array Properties with an Array Index</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	The IUT must contain a writable non-array property.
	<b>Testing Hints</b>	

### 4.6.3 Contains Writable Array Properties

The IUT contains, or can be made to contain, a writeable array property.

<b>BTL - 9.22.1.1 - Writing a Single Element of an Array</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.22.2.2 - Writing Array Properties with an Array Index that is Out of Range</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.6.4 Contains Resizable Array Properties

The IUT contains, or can be made to contain, an array property that is resizable by writing to the 0<sup>th</sup> element.

<b>BTL - 9.22.1.X1 - Writing an Array Size</b>		
	<b>Test Conditionality</b>	This test shall be executed on a single instance of each resizable property, both standard and proprietary, that do not have specific tests for those properties.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.1.16 - Array Sizing Test</b>		
	<b>Test Conditionality</b>	This test shall be executed if the IUT is protocol revision 4 or higher on a single instance of each resizable property, both standard and proprietary, that do not have specific tests for those properties.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.6.5 Contains Writable List Properties

The IUT contains, or can be made to contain, a writable list property. A writable list property is one that is modifiable via any of the BACnet services: WriteProperty, WritePropertyMultiple, AddListElement, or RemoveListElement.

<b>BTL - 9.22.1.X2 - Writing to Properties Based on Data Type</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	The property that is written for this instance of this test must be a list property.
	<b>Testing Hints</b>	

#### 4.6.6 Contains Commandable Properties

The IUT contains, or can be made to contain, a commandable property. Commandable properties are ones that are controlled by the command prioritization mechanism described in Clause 19.2 of BACnet-2001.

<b>135.1-2013 - 7.3.1.3 - Command Prioritization Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.22.1.2 - Writing a Commandable Property Without a Priority</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.6.7 Contains non-commandable Properties which Accept a Written NULL Value

The IUT contains, or can be made to contain, a writable property that accepts a written NULL value.

<b>BTL - 9.22.1.X2 - Writing to Properties Based on Data Type</b>		
	<b>Test Conditionality</b>	If the IUT cannot be configured to contain a property that will accept and retain a NULL value, then this test shall be omitted. Note that commandable properties do not retain the NULL value and as such devices that accept NULL values only for the relinquishing of commanded values will not be subjected to this test.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	Schedule_Default and Present_Value of the Schedule Object, Alarm_Values and Fault_Values of the CharacterString Value Object and Low_Diff_Limit in the Loop Object are standard properties that should accept a written NULL.

#### 4.6.8 Contains Writable BOOLEAN Properties

The IUT contains, or can be made to contain, a writable property with a data type of BOOLEAN.

<b>BTL - 9.22.1.X2 - Writing to Properties Based on Data Type</b>		
	<b>Test Conditionality</b>	If the IUT contains a property with a BOOLEAN data type when test 135.1-2013 - 7.2.2 is executed, this test can be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.6.9 Contains Writable Enumerated Properties

The IUT contains, or can be made to contain, a writable property with a data type of Enumerated.

<b>BTL - 9.22.1.X2 - Writing to Properties Based on Data Type</b>		
	<b>Test Conditionality</b>	If the IUT contains a property with an Enumerated data type when test 135.1-2013 - 7.2.2 is executed, this test can be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.6.10 Contains Writable INTEGER Properties

The IUT contains, or can be made to contain, a writable property with a data type of INTEGER.

<b>BTL - 9.22.1.X2 - Writing to Properties Based on Data Type</b>		
	<b>Test Conditionality</b>	If the IUT contains a property with an INTEGER data type when test 135.1-2013 - 7.2.2 is executed, this test can be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.6.11 Contains Writable Unsigned Properties

The IUT contains, or can be made to contain, a writable property with a data type of Unsigned.

<b>BTL - 9.22.1.X2 - Writing to Properties Based on Data Type</b>		
	<b>Test Conditionality</b>	If the IUT contains a property with an Unsigned data type when test 135.1-2013 - 7.2.2 is executed, this test can be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.6.12 Contains Writable REAL Properties

The IUT contains, or can be made to contain, a writable property with a data type of REAL.

<b>BTL - 9.22.1.X2 - Writing to Properties Based on Data Type</b>		
	<b>Test Conditionality</b>	If the IUT contains a property with a REAL data type when test 135.1-2013 - 7.2.2 is executed, this test can be skipped.

	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.6.13 Contains Writable Double Properties

The IUT contains, or can be made to contain, a writable property with a data type of Double.

<b>BTL - 9.22.1.X2 - Writing to Properties Based on Data Type</b>		
	<b>Test Conditionality</b>	If the IUT contains a property with a Double data type when test 135.1-2013 - 7.2.2 is executed, this test can be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.6.14 Contains Writable Time Properties

The IUT contains, or can be made to contain, a writable property with a data type of Time.

<b>BTL - 9.22.1.X2 - Writing to Properties Based on Data Type</b>		
	<b>Test Conditionality</b>	If the IUT contains a property with a Time data type when test 135.1-2013 - 7.2.2 is executed, this test can be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.6.15 Contains Writable Date Properties

The IUT contains, or can be made to contain, a writable property with a data type of Date.

<b>BTL - 9.22.1.X2 - Writing to Properties Based on Data Type</b>		
	<b>Test Conditionality</b>	If the IUT contains a property with a Date data type when test 135.1-2013 - 7.2.2 is executed, this test can be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.6.16 Contains Writable Character String Properties

The IUT contains, or can be made to contain, a writable property with a data type of Character String.

<b>BTL - 9.22.1.X2 - Writing to Properties Based on Data Type</b>		
	<b>Test Conditionality</b>	If the IUT contains a property with a Character String data type when test 135.1-2013 - 7.2.2 is executed, this test can be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.6.17 Contains Writable Octet String Properties

The IUT contains, or can be made to contain, a writable property with a data type of Octet String.

<b>BTL - 9.22.1.X2 - Writing to Properties Based on Data Type</b>		
	<b>Test Conditionality</b>	If the IUT contains a property with a Octet String data type when test 135.1-2013 - 7.2.2 is executed, this test can be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.6.18 Contains Writable Bit String Properties

The IUT contains, or can be made to contain, a writable property with a data type of Bit String.

<b>BTL - 9.22.1.X2 - Writing to Properties Based on Data Type</b>		
---	--	--



	<b>Test Conditionality</b>	If the IUT contains a property with a Bit String data type when test 135.1-2013 - 7.2.2 is executed, this test can be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.6.19 Contains Writable BACnetObjectIdentifier Properties

The IUT contains, or can be made to contain, a writable property with a data type of BACnetObjectIdentifier.

<b>BTL - 9.22.1.X2 - Writing to Properties Based on Data Type</b>		
	<b>Test Conditionality</b>	If the IUT contains a property with a BACnetObjectIdentifier data type when test 135.1-2013 - 7.2.2 is executed, this test can be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.6.20 Contains Writable Properties with Non-Basic Data Types

The IUT contains, or can be made to contain, a writable property with a non-basic data type. A non-basic data type is one that is represented by a SEQUENCE or CHOICE construct when described in ASN.1.

<b>BTL - 9.22.1.X2 - Writing to Properties Based on Data Type</b>		
	<b>Test Conditionality</b>	If properties of each of the writable non-basic data types that the IUT supports are present in the IUT when test 135.1-2013 - 7.2.2 is executed, this test can be skipped. Otherwise this test shall be repeated for non-basic writable data type that the IUT supports that was not tested by the execution of 135.1-2013 - 7.2.2.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.6.21 Contains Writable Proprietary Properties with Basic Data Types

The IUT contains, or can be made to contain, at least one writable proprietary property with a data type of NULL, BOOLEAN, INTEGER, Unsigned, REAL, Double, Character String, Octet String, Time, Date, or BACnetObjectIdentifier.

<b>BTL - 9.22.1.X2 - Writing to Properties Based on Data Type</b>		
	<b>Test Conditionality</b>	This test shall be skipped until such time as the test tools support interaction with proprietary properties.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

## 4.7 Data Sharing - WritePropertyMultiple - A

### 4.7.1 Base Requirements

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

For all of the DS-WPM-A tests, the vendor must supply information on how to make the IUT generate the specified requests.

As with all A side tests, the IUT must be able to initiate the service against any BACnet device that advertises that it executes WritePropertyMultiple.

### 4.7.2 Supports DS-WP-A

The IUT supports DS-WP-A.

Verify Checklist		
	Test Conditionality	Must be executed.
	Test Directives	Verify that the IUT claims conformance to DS-WP-A.
	Testing Hints	

### 4.7.3 Can Write Multiple Properties to a Single Object

The IUT is able to write multiple properties to a single object in a single WritePropertyMultiple request.

135.1-2013 - 8.23.2 - Writing Multiple Properties of a Single Object		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 4.7.4 Can Write A Single Property to Multiple Objects

The IUT is able to write a single property to multiple objects in a single WritePropertyMultiple request. The same property need not be written to each object.

135.1-2013 - 8.23.3 - Writing Multiple Objects, One Property Each		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 4.7.5 Can Write Multiple Properties to Multiple Objects

The IUT is able to write multiple properties to multiple objects in a single WritePropertyMultiple request.

135.1-2013 - 8.23.4 - Writing Multiple Objects, Multiple Properties for Each		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 4.7.6 Can Write A Single Property to a Single Object

The IUT is able to write a single property to a single object.

135.1-2013 - 8.23.1 - Writing a Single Property of a Single Object		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 4.7.7 Can Write Non-Array Properties

The IUT is able to write non-array properties in another BACnet device.

<b>135.1-2013 - 8.23.1 - Writing a Single Property of a Single Object, 135.1-2013 - 8.23.2 - Writing Multiple Properties of a Single Object, 135.1-2013 - 8.23.3 - Writing Multiple Objects, One Property Each, or 135.1-2013 - 8.23.4 - Writing Multiple Objects, Multiple Properties for Each</b>		
	<b>Test Conditionality</b>	At least one of the tests (8.23.1..8.23.4) shall be executed against a non-array property.
	<b>Test Directives</b>	At least one of the properties written by the selected test shall be a non-array property.
	<b>Testing Hints</b>	

### 4.7.8 Can Write Array Elements

The IUT is able to write array elements in another BACnet device.

<b>135.1-2013 - 8.23.1 - Writing a Single Property of a Single Object, 135.1-2013 - 8.23.2 - Writing Multiple Properties of a Single Object, 135.1-2013 - 8.23.3 - Writing Multiple Objects, One Property Each, or 135.1-2013 - 8.23.4 - Writing Multiple Objects, Multiple Properties for Each</b>		
	<b>Test Conditionality</b>	At least one of the tests (8.23.1..8.23.4) shall be executed against an array element.
	<b>Test Directives</b>	At least one of the properties written by the selected test shall be an array element. This will require the inclusion of an array index in the property specification even though the test definition does not show one.
	<b>Testing Hints</b>	

### 4.7.9 Can Write the Size of an Array

The IUT is able to write the 0<sup>th</sup> element of an array property to change the number of elements in the array.

<b>135.1-2013 - 8.23.1 - Writing a Single Property of a Single Object, 135.1-2013 - 8.23.2 - Writing Multiple Properties of a Single Object, 135.1-2013 - 8.23.3 - Writing Multiple Objects, One Property Each, or 135.1-2013 - 8.23.4 - Writing Multiple Objects, Multiple Properties for Each</b>		
	<b>Test Conditionality</b>	At least one of the tests (8.23.1..8.23.4) shall be executed against the 0 <sup>th</sup> element of an array.
	<b>Test Directives</b>	At least one of the properties written by the selected test shall be the 0 <sup>th</sup> element of an array.
	<b>Testing Hints</b>	

### 4.7.10 Can Write Whole Arrays

The IUT is able to write complete array properties without the use of array indices.

<b>135.1-2013 - 8.23.1 - Writing a Single Property of a Single Object, 135.1-2013 - 8.23.2 - Writing Multiple Properties of a Single Object, 135.1-2013 - 8.23.3 - Writing Multiple Objects, One Property Each, or 135.1-2013 - 8.23.4 - Writing Multiple Objects, Multiple Properties for Each</b>		
	<b>Test Conditionality</b>	At least one of the tests (8.23.1..8.23.4) shall be executed against an array property without the use of an array index.
	<b>Test Directives</b>	At least one of the properties written by the selected test shall be array property without an array element specified.
	<b>Testing Hints</b>	

#### 4.7.11 Can Write Whole Lists

The IUT is able to write list properties.

135.1-2013 - 8.23.1 - Writing a Single Property of a Single Object, 135.1-2013 - 8.23.2 - Writing Multiple Properties of a Single Object, 135.1-2013 - 8.23.3 - Writing Multiple Objects, One Property Each, or 135.1-2013 - 8.23.4 - Writing Multiple Objects, Multiple Properties for Each		
	Test Conditionality	At least one of the tests (8.23.1..8.23.4) shall be executed against a list property.
	Test Directives	At least one of the properties written by the selected test shall be a list property.
	Testing Hints	

#### 4.7.12 Can Command and Relinquish With a Priority

The IUT is able to include a priority when writing.

135.1-2013 - 8.23.6 - Writing Commandable Properties		
	Test Conditionality	Note that the test calls for 2 properties to be written, and only 1 contains a priority. The tester shall accept any well formed WritePropertyMultiple request, regardless of the number of properties contained in it, regardless of the number of properties for which a priority is supplied (as long as there is at least 1), and regardless of the position of the property(s) within the request that are accompanied by a priority.
	Test Directives	At least one of the properties written by the selected test shall be written with a priority.
	Testing Hints	

#### 4.7.13 Can Write NULL Property Values to non-commandable Properties

The IUT is able to write NULL values to non-commandable properties.

135.1-2013 - 8.23.1 - Writing a Single Property of a Single Object, 135.1-2013 - 8.23.2 - Writing Multiple Properties of a Single Object, 135.1-2013 - 8.23.3 - Writing Multiple Objects, One Property Each, or 135.1-2013 - 8.23.4 - Writing Multiple Objects, Multiple Properties for Each		
	Test Conditionality	At least one of the tests (8.23.1..8.23.4) shall be executed against a property with a NULL value.
	Test Directives	
	Testing Hints	Schedule_Default and Present_Value of the Schedule Object, Alarm_Values and Fault_Values of the CharacterString Value Object and Low_Diff_Limit in the Loop Object, are standard properties that should accept a written NULL.

#### 4.7.14 Can Write BOOLEAN Property Values

The IUT is able to write BOOLEAN values.

135.1-2013 - 8.23.1 - Writing a Single Property of a Single Object, 135.1-2013 - 8.23.2 - Writing Multiple Properties of a Single Object, 135.1-2013 - 8.23.3 - Writing Multiple Objects, One Property Each, or 135.1-2013 - 8.23.4 - Writing Multiple Objects, Multiple Properties for Each		
	Test Conditionality	At least one of the tests (8.23.1..8.23.4) shall be executed against a property with a BOOLEAN value.

<b>Test Directives</b>	At least one of the properties written by the selected test shall contain a BOOLEAN value.
<b>Testing Hints</b>	

#### 4.7.15 Can Write Enumerated Property Values

The IUT is able to write Enumerated values.

<b>135.1-2013 - 8.23.1 - Writing a Single Property of a Single Object,  135.1-2013 - 8.23.2 - Writing Multiple Properties of a Single Object,  135.1-2013 - 8.23.3 - Writing Multiple Objects, One Property Each, or  135.1-2013 - 8.23.4 - Writing Multiple Objects, Multiple Properties for Each</b>	
<b>Test Conditionality</b>	At least one of the tests (8.23.1..8.23.4) shall be executed against a property with an Enumerated value.
<b>Test Directives</b>	At least one of the properties written by the selected test shall contain an Enumerated value.
<b>Testing Hints</b>	

#### 4.7.16 Can Write INTEGER Property Values

The IUT is able to write INTEGER values.

<b>135.1-2013 - 8.23.1 - Writing a Single Property of a Single Object,  135.1-2013 - 8.23.2 - Writing Multiple Properties of a Single Object,  135.1-2013 - 8.23.3 - Writing Multiple Objects, One Property Each, or  135.1-2013 - 8.23.4 - Writing Multiple Objects, Multiple Properties for Each</b>	
<b>Test Conditionality</b>	At least one of the tests (8.23.1..8.23.4) shall be executed against a property with an INTEGER value.
<b>Test Directives</b>	At least one of the properties written by the selected test shall contain an INTEGER value.
<b>Testing Hints</b>	

#### 4.7.17 Can Write Unsigned Property Values

The IUT is able to write Unsigned values.

<b>135.1-2013 - 8.23.1 - Writing a Single Property of a Single Object,  135.1-2013 - 8.23.2 - Writing Multiple Properties of a Single Object,  135.1-2013 - 8.23.3 - Writing Multiple Objects, One Property Each, or  135.1-2013 - 8.23.4 - Writing Multiple Objects, Multiple Properties for Each</b>	
<b>Test Conditionality</b>	At least one of the tests (8.23.1..8.23.4) shall be executed against a property with an Unsigned value.
<b>Test Directives</b>	At least one of the properties written by the selected test shall contain an Unsigned value.
<b>Testing Hints</b>	

#### 4.7.18 Can Write REAL Property Values

The IUT is able to write REAL values.

<b>135.1-2013 - 8.23.1 - Writing a Single Property of a Single Object,  135.1-2013 - 8.23.2 - Writing Multiple Properties of a Single Object,  135.1-2013 - 8.23.3 - Writing Multiple Objects, One Property Each, or  135.1-2013 - 8.23.4 - Writing Multiple Objects, Multiple Properties for Each</b>	
<b>Test Conditionality</b>	At least one of the tests (8.23.1..8.23.4) shall be executed against a property with a REAL value.
<b>Test Directives</b>	At least one of the properties written by the selected test shall contain a REAL value.

	Testing Hints	
--	---------------	--

#### 4.7.19 Can Write Double Property Values

The IUT is able to write Double values.

<b>135.1-2013 - 8.23.1 - Writing a Single Property of a Single Object,  135.1-2013 - 8.23.2 - Writing Multiple Properties of a Single Object,  135.1-2013 - 8.23.3 - Writing Multiple Objects, One Property Each, or  135.1-2013 - 8.23.4 - Writing Multiple Objects, Multiple Properties for Each</b>		
	Test Conditionality	At least one of the tests (8.23.1..8.23.4) shall be executed against a property with a Double value.
	Test Directives	At least one of the properties written by the selected test shall contain a Double value.
	Testing Hints	

#### 4.7.20 Can Write Time Property Values

The IUT is able to write Time values.

<b>135.1-2013 - 8.23.1 - Writing a Single Property of a Single Object,  135.1-2013 - 8.23.2 - Writing Multiple Properties of a Single Object,  135.1-2013 - 8.23.3 - Writing Multiple Objects, One Property Each, or  135.1-2013 - 8.23.4 - Writing Multiple Objects, Multiple Properties for Each</b>		
	Test Conditionality	At least one of the tests (8.23.1..8.23.4) shall be executed against a property with a Time value.
	Test Directives	At least one of the properties written by the selected test shall contain a Time value.
	Testing Hints	

#### 4.7.21 Can Write Date Property Values

The IUT is able to write Date values.

<b>135.1-2013 - 8.23.1 - Writing a Single Property of a Single Object,  135.1-2013 - 8.23.2 - Writing Multiple Properties of a Single Object,  135.1-2013 - 8.23.3 - Writing Multiple Objects, One Property Each, or  135.1-2013 - 8.23.4 - Writing Multiple Objects, Multiple Properties for Each</b>		
	Test Conditionality	At least one of the tests (8.23.1..8.23.4) shall be executed against a property with a Date value.
	Test Directives	At least one of the properties written by the selected test shall contain a Date value.
	Testing Hints	

#### 4.7.22 Can Write Character String Property Values

The IUT is able to write Character String values.

<b>135.1-2013 - 8.23.1 - Writing a Single Property of a Single Object,  135.1-2013 - 8.23.2 - Writing Multiple Properties of a Single Object,  135.1-2013 - 8.23.3 - Writing Multiple Objects, One Property Each, or  135.1-2013 - 8.23.4 - Writing Multiple Objects, Multiple Properties for Each</b>		
	Test Conditionality	At least one of the tests (8.23.1..8.23.4) shall be executed against a property with a Character String value.
	Test Directives	At least one of the properties written by the selected test shall contain a Character String value.
	Testing Hints	

### 4.7.23 Can Write Octet String Property Values

The IUT is able to write Octet String values.

<b>135.1-2013 - 8.23.1 - Writing a Single Property of a Single Object,  135.1-2013 - 8.23.2 - Writing Multiple Properties of a Single Object,  135.1-2013 - 8.23.3 - Writing Multiple Objects, One Property Each, or  135.1-2013 - 8.23.4 - Writing Multiple Objects, Multiple Properties for Each</b>		
	<b>Test Conditionality</b>	At least one of the tests (8.23.1..8.23.4) shall be executed against a property with an Octet String value.
	<b>Test Directives</b>	At least one of the properties written by the selected test shall contain an Octet String value.
	<b>Testing Hints</b>	

### 4.7.24 Can Write Bit String Property Values

The IUT is able to write Bit String values.

<b>135.1-2013 - 8.23.1 - Writing a Single Property of a Single Object,  135.1-2013 - 8.23.2 - Writing Multiple Properties of a Single Object,  135.1-2013 - 8.23.3 - Writing Multiple Objects, One Property Each, or  135.1-2013 - 8.23.4 - Writing Multiple Objects, Multiple Properties for Each</b>		
	<b>Test Conditionality</b>	At least one of the tests (8.23.1..8.23.4) shall be executed against a property with a Bit String value.
	<b>Test Directives</b>	At least one of the properties written by the selected test shall contain a Bit String value.
	<b>Testing Hints</b>	

### 4.7.25 Can Write BACnetObjectIdentifier Property Values

The IUT is able to write BACnetObjectIdentifier values.

<b>135.1-2013 - 8.23.1 - Writing a Single Property of a Single Object,  135.1-2013 - 8.23.2 - Writing Multiple Properties of a Single Object,  135.1-2013 - 8.23.3 - Writing Multiple Objects, One Property Each, or  135.1-2013 - 8.23.4 - Writing Multiple Objects, Multiple Properties for Each</b>		
	<b>Test Conditionality</b>	At least one of the tests (8.23.1..8.23.4) shall be executed against a property with a BACnetObjectIdentifier value.
	<b>Test Directives</b>	At least one of the properties written by the selected test shall contain a BACnetObjectIdentifier value.
	<b>Testing Hints</b>	

### 4.7.26 Can Write Constructed Property Values

The IUT is able to write constructed property values.

<b>135.1-2013 - 8.23.1 - Writing a Single Property of a Single Object,  135.1-2013 - 8.23.2 - Writing Multiple Properties of a Single Object,  135.1-2013 - 8.23.3 - Writing Multiple Objects, One Property Each, or  135.1-2013 - 8.23.4 - Writing Multiple Objects, Multiple Properties for Each</b>		
	<b>Test Conditionality</b>	At least one of the tests (8.23.1..8.23.4) shall be executed against a property with a constructed value. This test shall be repeated for each standard constructed value that the IUT is able to write.
	<b>Test Directives</b>	At least one of the properties written by the selected test shall contain a constructed value.
	<b>Testing Hints</b>	Where a constructed value can take on different forms, such as a constructed value that contains optional elements, or is a CHOICE, the tester should test all forms of the datatype.

#### 4.7.27 Can Write Proprietary Property Values of Basic Data Types

The IUT is able to write arbitrary proprietary properties with a data type of NULL, BOOLEAN, INTEGER, Unsigned, REAL, Double, Character String, Octet String, Time, Date, or BACnetObjectIdentifier.

<b>135.1-2013 - 8.23.1 - Writing a Single Property of a Single Object,  135.1-2013 - 8.23.2 - Writing Multiple Properties of a Single Object,  135.1-2013 - 8.23.3 - Writing Multiple Objects, One Property Each, or  135.1-2013 - 8.23.4 - Writing Multiple Objects, Multiple Properties for Each</b>		
	<b>Test Conditionality</b>	At least one of the tests (8.23.1..8.23.4) shall be executed against a proprietary property.
	<b>Test Directives</b>	At least one of the properties written by the selected test shall be proprietary.
	<b>Testing Hints</b>	



## 4.8 Data Sharing - WritePropertyMultiple - B

### 4.8.1 Base Requirements

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

<b>BTL - 7.2.2 - Write Support Test Procedure</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	For each property on which 9.23.1.7 - Writing to Properties Based on Data Type is executed on each writable datatype, that property can be skipped in the performance of 7.2.2. To satisfy this test item, test 7.2.2 need only be executed using WritePropertyMultiple.
	<b>Testing Hints</b>	
<b>BTL - 9.23.1.1 - Writing a Single Property to a Single Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.23.1.5 - Writing a Non-Commandable Property With a Priority</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.23.2.1 - Writing Multiple Properties with a Property Access Error</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	The IUT must contain a writable non-array property.
	<b>Testing Hints</b>	
<b>BTL - 9.23.2.2 - Writing Multiple Properties with an Object Access Error</b>		
	<b>Test Conditionality</b>	If the product does not contain, or cannot be made to contain an object with a writable property and a read-only property, then this test shall be omitted
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.23.2.3 - Writing Multiple Properties with a Write Access Error</b>		
	<b>Test Conditionality</b>	If the product does not contain, or cannot be made to contain an object with a writable property and a read-only property, then this test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.23.2.6 - Writing with a Property Value Having the Wrong Datatype</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.23.2.7 - Writing with a Property Value that is Out of Range</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.23.2.X1 - WritePropertyMultiple Reject Test</b>		
	<b>Test Conditionality</b>	Must be executed if the IUT claims Protocol Revision 10 or greater.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.23.2.X2 - Resizing a writable fixed size array property using WritePropertyMultiple service</b>		
	<b>Test Conditionality</b>	If IUT does not contain a writable fixed size array property, then this test shall be skipped.
	<b>Test Directives</b>	

	<b>Testing Hints</b>	For example, Weekly Schedule.
<b>BTL - 9.23.2.X6 - Writing first element of 'List of Write Access Specifications' with a Property Access Error</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.23.2.X3 - Writing first element of 'List of Write Access Specifications' with Object Access Error</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.23.2.X4 - Writing first element of 'List of Write Access Specifications with a Write Access Error</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.23.2.X5 - WritePropertyMultiple Reject Test for first element of 'List of Write Access Specifications'</b>		
	<b>Test Conditionality</b>	Must be executed if the IUT claims Protocol Revision 10 or greater.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.8.2 Contains Multiple Objects with Writable Properties

The IUT contains, or can be made to contain, multiple objects with writable properties.

<b>BTL - 9.23.1.3 - Writing a Single Property to Multiple Objects</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.8.3 Contains Objects with Multiple Writable Properties

The IUT contains, or can be made to contain, at least one object with multiple writable properties.

<b>BTL - 9.23.1.2 - Writing Multiple properties to a Single Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	The IUT must contain a writable non-array property.
	<b>Testing Hints</b>	

#### 4.8.4 Contains Multiple Objects with Multiple Writable Properties

<b>BTL - 9.23.1.4 - Writing Multiple Properties to Multiple Objects</b>		
	<b>Test Conditionality</b>	This test shall be skipped if the IUT has only a single object with multiple writable properties.
	<b>Test Directives</b>	The IUT must contain a writable non-array property.
	<b>Testing Hints</b>	

#### 4.8.5 Contains Writable Non-Array Properties

The IUT contains, or can be made to contain, a writeable non-array property.

<b>135.1-2013 - 9.23.1.8 - Writing to Properties Based on Data Type</b>		
	<b>Test Conditionality</b>	Must be executed.

	<b>Test Directives</b>	The test shall be executed against a writable non-array property of any data type.
	<b>Testing Hints</b>	

#### 4.8.6 Contains Writable Array Properties

The IUT contains, or can be made to contain, a writable array property.

135.1-2013 - 9.23.1.8 - Writing to Properties Based on Data Type		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	The test shall be executed against a whole writable array property (no array index provided, and all elements written).
	<b>Testing Hints</b>	
BTL - 9.23.2.5 - Writing Array Properties with an Array Index that is Out of Range		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.8.7 Contains Resizable Array Properties

The IUT contains, or can be made to contain, an array property that is resizable by writing to the 0<sup>th</sup> element.

BTL - 9.23.1.X4 - Writing an Array Size		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
BTL - 7.3.1.X16 - Array Resizing Test using WritePropertyMultiple Service		
	<b>Test Conditionality</b>	This test shall be executed if the IUT is protocol revision 4 or higher.
	<b>Test Directives</b>	Execute on at least one instance of each resizable array property, both standard and proprietary
	<b>Testing Hints</b>	

#### 4.8.8 Contains Writable List Properties

The IUT contains, or can be made to contain, a writable list property. A writable list property is one that is modifiable via any of the BACnet services: WriteProperty, WritePropertyMultiple, AddListElement, or RemoveListElement.

135.1-2013 - 9.23.1.8 - Writing to Properties Based on Data Type		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	The property that is written for this instance of this test must be a list property.
	<b>Testing Hints</b>	

#### 4.8.9 Contains Commandable Properties

The IUT contains, or can be made to contain, a commandable property. Commandable properties are ones that are controlled by the command prioritization mechanism described in Clause 19.2 of BACnet-2001.

135.1-2013 - 7.3.1.3 - Command Prioritization Test		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Use WritePropertyMultiple is used in place of WriteProperty.
	<b>Testing Hints</b>	
135.1-2013 - 9.23.1.6 - Writing a Commandable Property Without a Priority		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.8.10 Contains non-commandable Properties which Accept a Written NULL Value

The IUT contains, or can be made to contain, a writable property that accepts a written NULL value.

135.1-2013 - 9.23.1.8 - Writing to Properties Based on Data Type		
	<b>Test Conditionality</b>	If the IUT cannot be configured to contain a property that will accept and retain a NULL value, then this test shall be omitted. Note that commandable properties do not retain the NULL value and as such devices that accept NULL values only for the relinquishing of commanded values will not be subjected to this test.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	Schedule_Default and Present_Value of the Schedule Object, Alarm_Values and Fault_Values of the CharacterString Value Object and Low_Diff_Limit in the Loop Object are standard properties that should accept a written NULL.

#### 4.8.11 Contains Writable BOOLEAN Properties

The IUT contains, or can be made to contain, a writable property with a data type of BOOLEAN.

135.1-2013 - 9.23.1.8 - Writing to Properties Based on Data Type		
	<b>Test Conditionality</b>	If the IUT contains a property with a BOOLEAN data type when test 135.1-2013 - 7.2.2 is executed, this test can be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.8.12 Contains Writable Enumerated Properties

The IUT contains, or can be made to contain, a writable property with a data type of Enumerated.

135.1-2013 - 9.23.1.8 - Writing to Properties Based on Data Type		
	<b>Test Conditionality</b>	If the IUT contains a property with an Enumerated data type when test 135.1-2013 - 7.2.2 is executed, this test can be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.8.13 Contains Writable INTEGER Properties

The IUT contains, or can be made to contain, a writable property with a data type of INTEGER.

135.1-2013 - 9.23.1.8 - Writing to Properties Based on Data Type		
	<b>Test Conditionality</b>	If the IUT contains a property with an INTEGER data type when test 135.1-2013 - 7.2.2 is executed, this test can be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.8.14 Contains Writable Unsigned Properties

The IUT contains, or can be made to contain, a writable property with a data type of Unsigned.

135.1-2013 - 9.23.1.8 - Writing to Properties Based on Data Type		
	<b>Test Conditionality</b>	If the IUT contains a property with an Unsigned data type when test 135.1-2013 - 7.2.2 is executed, this test can be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.8.15 Contains Writable REAL Properties

The IUT contains, or can be made to contain, a writable property with a data type of REAL.

135.1-2013 - 9.23.1.8 - Writing to Properties Based on Data Type		
	<b>Test Conditionality</b>	If the IUT contains a property with a REAL data type when test 135.1-2013 - 7.2.2 is executed, this test can be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.8.16 Contains Writable Double Properties

The IUT contains, or can be made to contain, a writable property with a data type of Double.

135.1-2013 - 9.23.1.8 - Writing to Properties Based on Data Type		
	<b>Test Conditionality</b>	If the IUT contains a property with a Double data type when test 135.1-2013 - 7.2.2 is executed, this test can be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.8.17 Contains Writable Time Properties

The IUT contains, or can be made to contain, a writable property with a data type of Time.

135.1-2013 - 9.23.1.8 - Writing to Properties Based on Data Type		
	<b>Test Conditionality</b>	If the IUT contains a property with a Time data type when test 135.1-2013 - 7.2.2 is executed, this test can be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.8.18 Contains Writable Date Properties

The IUT contains, or can be made to contain, a writable property with a data type of Date.

135.1-2013 - 9.23.1.8 - Writing to Properties Based on Data Type		
	<b>Test Conditionality</b>	If the IUT contains a property with a Date data type when test 135.1-2013 - 7.2.2 is executed, this test can be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.8.19 Contains Writable Character String Properties

The IUT contains, or can be made to contain, a writable property with a data type of Character String.

135.1-2013 - 9.23.1.8 - Writing to Properties Based on Data Type		
	<b>Test Conditionality</b>	If the IUT contains a property with a Character String data type when test 135.1-2013 - 7.2.2 is executed, this test can be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.8.20 Contains Writable Octet String Properties

The IUT contains, or can be made to contain, a writable property with a data type of Octet String.

135.1-2013 - 9.23.1.8 - Writing to Properties Based on Data Type		
	<b>Test Conditionality</b>	If the IUT contains a property with a Octet String data type when test 135.1-2013 - 7.2.2 is executed, this test can be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.8.21 Contains Writable Bit String Properties

The IUT contains, or can be made to contain, a writable property with a data type of Bit String.

135.1-2013 - 9.23.1.8 - Writing to Properties Based on Data Type		
	<b>Test Conditionality</b>	If the IUT contains a property with a Bit String data type when test 135.1-2013 - 7.2.2 is executed, this test can be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.8.22 Contains Writable BACnetObjectIdentifier Properties

The IUT contains, or can be made to contain, a writable property with a data type of BACnetObjectIdentifier.

135.1-2013 - 9.23.1.8 - Writing to Properties Based on Data Type		
	<b>Test Conditionality</b>	If the IUT contains a property with a BACnetObjectIdentifier data type when test 135.1-2013 - 7.2.2 is executed, this test can be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.8.23 Contains Writable Properties with Non-Basic Data Types

The IUT contains, or can be made to contain, a writable property with a non-basic data type. A non-basic data type is one that is represented by a SEQUENCE or CHOICE construct when described in ASN.1.

135.1-2013 - 9.23.1.8 - Writing to Properties Based on Data Type		
	<b>Test Conditionality</b>	If properties of each of the writable non-basic data types that the IUT supports are present in the IUT when test 135.1-2013 - 7.2.2 is executed, this test can be skipped. Otherwise this test shall be repeated for non-basic writable data type that the IUT supports that was not tested by the execution of 135.1-2013 - 7.2.2.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.8.24 Contains Writable Proprietary Properties with Basic Data Types

The IUT contains, or can be made to contain, at least one writable proprietary property with a data type of NULL, BOOLEAN, INTEGER, Unsigned, REAL, Double, Character String, Octet String, Time, Date, or BACnetObjectIdentifier.

135.1-2013 - 9.23.1.8 - Writing to Properties Based on Data Type		
	<b>Test Conditionality</b>	This test shall be skipped until such time as the test tools support interaction with proprietary properties.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

## 4.9 Data Sharing - Change Of Value - A

### 4.9.1 Base Requirements

Base requirements must be met by any IUT claiming conformance to this BIBB. There are no base requirements for this section.

### 4.9.2 Subscribes with Lifetimes up to 8 Hours in Duration

The IUT is capable of subscribing with a lifetime less than or equal to 28800 seconds (8 hours).

135.1-2013 - 8.10.4 - Generates 8 Hour Lifetimes		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 4.9.3 Can Subscribe for Confirmed Notifications

The IUT can subscribe for, receive, and process confirmed Change of Value notifications.

135.1-2013 - 8.10.1 - Confirmed Notifications Subscription		
	Test Conditionality	Must be executed.
	Test Directives	This test shall not be executed with a lifetime of 0.
	Testing Hints	
BTL - 9.2.2.1 - Change of Value Notification Arrives after Subscription has Expired		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
BTL - 9.2.2.2 - Change of Value Notifications with Invalid Process Identifier		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
BTL - 9.2.2.4 - Change of Value Notifications with Invalid Monitored Object Identifier		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 4.9.4 Can Subscribe for Unconfirmed Notifications

The IUT can subscribe for, receive, and process unconfirmed Change of Value notifications.

135.1-2013 - 8.10.2 - Unconfirmed Notifications Subscription		
	Test Conditionality	Must be executed.
	Test Directives	This test shall not be executed with a lifetime of 0.
	Testing Hints	

### 4.9.5 Can Subscribe for COV from Analog Objects

The IUT can subscribe for, receive, and process Change of Value notifications from at least one object using type REAL in its parameters.

135.1-2013 - 9.2.1.1 - Change of Value Notifications		
	Test Conditionality	Either 9.2.1.1 or 9.3.2 must be executed.
	Test Directives	Test one instance of each object type where Present_Value is of data type REAL.
	Testing Hints	

135.1-2013 - 9.3.2 - Change of Value Notifications		
	<b>Test Conditionality</b>	Either 9.2.1.1 or 9.3.2 must be executed.
	<b>Test Directives</b>	Test one instance of each object type where Present_Value is of data type REAL.
	<b>Testing Hints</b>	

#### 4.9.6 Can Subscribe for COV from Binary Objects

The IUT can subscribe for, receive, and process Change of Value notifications from at least one object using type BACnetBinaryPV in its parameters.

135.1-2013 - 9.2.1.1 - Change of Value Notifications		
	<b>Test Conditionality</b>	Either 9.2.1.1 or 9.3.2 must be executed.
	<b>Test Directives</b>	Test one instance of each object type where Present_Value is of data type BACnetBinaryPV.
	<b>Testing Hints</b>	
135.1-2013 - 9.3.2 - Change of Value Notifications		
	<b>Test Conditionality</b>	Either 9.2.1.1 or 9.3.2 must be executed.
	<b>Test Directives</b>	Test one instance of each object type where Present_Value is of data type BACnetBinaryPV.
	<b>Testing Hints</b>	

#### 4.9.7 Can Subscribe for COV from Life Safety Objects

The IUT can subscribe for, receive, and process Change of Value notifications from at least one Life Safety Point or Life Safety Zone object using type BACnetLifeSafetyState in its parameters.

135.1-2013 - 9.2.1.1 - Change of Value Notifications		
	<b>Test Conditionality</b>	Either 9.2.1.1 or 9.3.2 must be executed.
	<b>Test Directives</b>	Test one instance of each object type where Present_Value is of data type BACnetLifeSafetyState.
	<b>Testing Hints</b>	
135.1-2013 - 9.3.2 - Change of Value Notifications		
	<b>Test Conditionality</b>	Either 9.2.1.1 or 9.3.2 must be executed.
	<b>Test Directives</b>	Test one instance of each object type where Present_Value is of data type BACnetLifeSafetyState.
	<b>Testing Hints</b>	

#### 4.9.8 Can Subscribe for COV from Loop Objects

The IUT can subscribe for, receive, and process Change of Value notifications from Loop objects.

135.1-2013 - 9.2.1.2 - Change of Value Notification from Loop Objects		
	<b>Test Conditionality</b>	Either 9.2.1.2 or 9.3.3 must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
135.1-2013 - 9.3.3 - Change of Value Notification from Loop Objects		
	<b>Test Conditionality</b>	Either 9.2.1.2 or 9.3.3 must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.9.9 Can Subscribe for COV from Multi-state Objects

The IUT can subscribe for, receive, and process Change of Value notifications from at least one object using type Unsigned in its parameters.

135.1-2013 - 9.2.1.1 - Change of Value Notifications		
--	--	--



	<b>Test Conditionality</b>	Either 9.2.1.1 or 9.3.2 must be executed.
	<b>Test Directives</b>	Test one instance of each object type where Present_Value is of data type Unsigned.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.3.2 - Change of Value Notifications</b>		
	<b>Test Conditionality</b>	Either 9.2.1.1 or 9.3.2 must be executed.
	<b>Test Directives</b>	Test one instance of each object type where Present_Value is of data type Unsigned.
	<b>Testing Hints</b>	

#### 4.9.10 Can Subscribe for COV from CharacterString Objects

The IUT supports change of value notifications for at least one object using type CharacterString in its parameters.

<b>135.1-2013 - 9.2.1.1 - Change of Value Notifications</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Test one instance of each object type where Present_Value is of data type CharacterString.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.3.2 - Change of Value Notifications</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Test one instance of each object type where Present_Value is of data type CharacterString.
	<b>Testing Hints</b>	

#### 4.9.11 Can Subscribe for COV from Date Objects

The IUT supports change of value notifications for at least one object using type Date in its parameters.

<b>135.1-2013 - 9.2.1.1 - Change of Value Notifications</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Test one instance of each object type where Present_Value is of data type Date.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.3.2 - Change of Value Notifications</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Test one instance of each object type where Present_Value is of data type Date.
	<b>Testing Hints</b>	

#### 4.9.12 Can Subscribe for COV from DateTime Objects

The IUT supports change of value notifications for at least one object using type BACnetDateTime in its parameters.

<b>135.1-2013 - 9.2.1.1 - Change of Value Notifications</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Test one instance of each object type where Present_Value is of data type BACnetDateTime.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.3.2 - Change of Value Notifications</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Test one instance of each object type where Present_Value is of data type BACnetDateTime.
	<b>Testing Hints</b>	

	Testing Hints	
--	---------------	--

#### 4.9.13 Can Subscribe for COV from Integer Objects

The IUT supports change of value notifications for at least one object of type Integer.

135.1-2013 - 9.2.1.1 - Change of Value Notifications		
	Test Conditionality	Must be executed.
	Test Directives	Test one instance of each object type where Present_Value is of data type Integer.
	Testing Hints	
135.1-2013 - 9.3.2 - Change of Value Notifications		
	Test Conditionality	Must be executed.
	Test Directives	Test one instance of each object type where Present_Value is of data type Integer.
	Testing Hints	

#### 4.9.14 Can Subscribe for COV from Large Analog Objects

The IUT supports change of value notifications for at least one object using type Integer in its parameters.

135.1-2013 - 9.2.1.1 - Change of Value Notifications		
	Test Conditionality	Must be executed.
	Test Directives	Test one instance of each object type where Present_Value is of data type Large Analog.
	Testing Hints	
135.1-2013 - 9.3.2 - Change of Value Notifications		
	Test Conditionality	Must be executed.
	Test Directives	Test one instance of each object type where Present_Value is of data type Large Analog.
	Testing Hints	

#### 4.9.15 Can Subscribe for COV from OctetString Objects

The IUT supports change of value notifications for at least one object of type OctetString.

135.1-2013 - 9.2.1.1 - Change of Value Notifications		
	Test Conditionality	Must be executed.
	Test Directives	Test one instance of each object type where Present_Value is of data type OctetString.
	Testing Hints	
135.1-2013 - 9.3.2 - Change of Value Notifications		
	Test Conditionality	Must be executed.
	Test Directives	Test one instance of each object type where Present_Value is of data type OctetString.
	Testing Hints	

#### 4.9.16 Can Subscribe for COV from Positive Integer Objects

The IUT supports change of value notifications for at least one object of type Positive Integer.

135.1-2013 - 9.2.1.1 - Change of Value Notifications		
	Test Conditionality	Must be executed.
	Test Directives	Test one instance of each object type where Present_Value is using data type Unsigned, including the possible value zero.
	Testing Hints	

<b>135.1-2013 - 9.3.2 - Change of Value Notifications</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Test one instance of each object type where Present_Value is using data type Unsigned, including the possible value zero.
	<b>Testing Hints</b>	

#### 4.9.17 Can Subscribe for COV from Time Objects

The IUT supports change of value notifications for at least one object of type Time.

<b>135.1-2013 - 9.2.1.1 - Change of Value Notifications</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Test one instance of each object type where Present_Value is of data type Time.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.3.2 - Change of Value Notifications</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Test one instance of each object type where Present_Value is of data type Time.
	<b>Testing Hints</b>	

#### 4.9.18 Can Subscribe for COV from Pulse Converter Objects

The IUT supports change of value notifications for at least one object of type Pulse Converter.

<b>135.1-2013 - 9.2.1.1 - Change of Value Notifications</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Test one instance of each object type where Present_Value is of data type REAL.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.3.2 - Change of Value Notifications</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Test one instance of each object type where Present_Value is of data type REAL.
	<b>Testing Hints</b>	

#### 4.9.19 Can Subscribe for COV from Access Door Objects

The IUT supports change of value notifications for at least one object of type Access Door.

<b>BTL - 9.2.1.X5 - ConfirmedCOVNotification from Access Door Object</b>		
	<b>Test Conditionality</b>	Either 9.2.1.X5 or 9.3.1.X6 must be executed
	<b>Test Directives</b>	Test at least one instance where object type is an Access Door.
	<b>Testing Hints</b>	
<b>BTL - 9.3.1.X6 - UnconfirmedCOVNotification from Access Door Object</b>		
	<b>Test Conditionality</b>	Either 9.2.1.X5 or 9.3.1.X6 must be executed
	<b>Test Directives</b>	Test at least one instance where object type is an Access Door.
	<b>Testing Hints</b>	

#### 4.9.20 Can Subscribe for COV from Load Control Objects

The IUT supports change of value notifications for at least one object of type Load Control.

<b>135.1-2013 - 9.2.1.1 - Change of Value Notifications</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Test one instance of each object type where Present_Value is of data type BACnetShedState.

	<b>Testing Hints</b>	
<b>135.1-2013 - 9.3.2 - Change of Value Notifications</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Test one instance of each object type where Present_Value is of data type BACnetShedState.
	<b>Testing Hints</b>	

#### 4.9.21 Can Subscribe for COV from Access Point Objects

The IUT supports change of value notifications for at least one object of type Access Point.

<b>135.1-2013 - 9.2.1.1 - Change of Value Notifications</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Test one instance of each object type.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.3.2 - Change of Value Notifications</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Test one instance of each object type.
	<b>Testing Hints</b>	

#### 4.9.22 Can Subscribe for COV from Credential Data Input Objects

The IUT supports change of value notifications for at least one object of type Credential Data Input.

<b>135.1-2013 - 9.2.1.1 - Change of Value Notifications</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Test one instance of each object type where Present_Value is of data type BACnetAuthenticationFactor.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.3.2 - Change of Value Notifications</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Test one instance of each object type where Present_Value is of data type BACnetAuthenticationFactor.
	<b>Testing Hints</b>	

#### 4.9.23 Can Subscribe for COV from Lighting Output Objects

The IUT supports change of value notifications for at least one object of type Lighting Output.

<b>135.1-2013 - 9.2.1.1 - Change of Value Notifications</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Test one instance of each object type where Present_Value is of data type REAL.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.3.2 - Change of Value Notifications</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Test one instance of each object type where Present_Value is of data type REAL.
	<b>Testing Hints</b>	

#### 4.9.24 Can Subscribe for COV from Binary Lighting Output Objects

The IUT supports change of value notifications for at least one object of type Binary Lighting Output.

<b>135.1-2013 - 9.2.1.1 - Change of Value Notifications</b>		
	<b>Test Conditionality</b>	Must be executed.

	<b>Test Directives</b>	Test one instance of each object type where Present_Value is of data type BACnetBinaryLightingPV.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.3.2 - Change of Value Notifications</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Test one instance of each object type where Present_Value is of data type BACnetBinaryLightingPV.
	<b>Testing Hints</b>	

#### 4.9.25 Can Subscribe for COV from Staging Objects

The IUT supports change of value notifications for at least one object of type Staging.

<b>135.1-2013 - 9.2.1.1 - Change of Value Notifications</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Test one instance of each object type.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.3.2 - Change of Value Notifications</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Test one instance of each object type.
	<b>Testing Hints</b>	

#### 4.9.26 Can Cancel Subscriptions

The IUT can explicitly cancel COV subscriptions (in contrast to just letting the subscription expire).

<b>135.1-2013 - 8.10.3 - Cancelling a Subscription</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.9.27 Can Subscribe for COV from Proprietary Objects

The IUT can subscribe for, receive, and process Change of Value notifications from Proprietary objects.

<b>BTL - 9.2.1.X4 - Change of Value Notification from proprietary Objects</b>		
	<b>Test Conditionality</b>	This test is not yet defined and shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.3.X9 - Change of Value Notification from Proprietary Objects</b>		
	<b>Test Conditionality</b>	This test is not yet defined and shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.9.28 Can Request Infinite Subscriptions

The IUT can generate COV subscription requests with no Lifetime parameter.

There are no tests for this functionality.

## 4.10 Data Sharing - Change Of Value - B

### 4.10.1 Base Requirements

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

<b>135.1-2013 - 9.10.1.1 - Confirmed COV Notifications</b>		
	<b>Test Conditionality</b>	The functionality of this test is covered by tests 8.2.X which may be executed for each object type that supports COV. If the 8.2.X tests have been executed, then this test may be omitted.
	<b>Test Directives</b>	This test shall only be executed against objects which will accept the subscription.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.10.1.2 - Unconfirmed COV Notifications</b>		
	<b>Test Conditionality</b>	The functionality of this test is covered by tests 8.3.1 through 8.3.n which may be executed for each object type that supports COV. If the 8.3.n tests have been executed, then this test may be omitted.
	<b>Test Directives</b>	This test shall only be executed against objects which will accept the subscription.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.10.1.4 - Cancelling COV Subscriptions</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.10.1.5 - Cancelling Expired or Non-Existing Subscriptions</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.10.1.7 - Finite Lifetime Subscriptions</b>		
	<b>Test Conditionality</b>	Optional. The functionality of this test is covered by tests 8.2.1 through n and 8.3.1 through n which are executed for each object type that supports COV.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.10.1.9 - Ensuring Subscription Lifetimes Are Not Effected By Time Changes</b>		
	<b>Test Conditionality</b>	This test must be executed if the device can execute TimeSynchronization or UTCTimeSynchronization.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.10.2.1 - The Monitored Object Does Not Support COV Notification</b>		
	<b>Test Conditionality</b>	This test shall only be executed if IUT contains objects which will not accept a COV subscription.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.10.1 - Active COV Subscriptions SubscribeCOV Test</b>		
	<b>Test Conditionality</b>	This test must be executed if the device claims conformance to protocol revision 1 or higher.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.10.2.X1 - The Monitored Object Does Not Exist</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.10.2.X2 - There Is No Space For A Subscription</b>		

	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.10.2.X3 - The LifeTime Parameter is Out of Range</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.10.2 Supports Lifetimes up to 8 Hours in Duration

The IUT will accept COV subscriptions with lifetimes up to 8 hours.

<b>135.1-2013 - 9.10.1.10 - Accepts 8 Hour Lifetimes</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.10.3 Supports 5 concurrent COV subscribers

The IUT supports 5 or more concurrent COV subscriptions

<b>BTL - 9.10.1.X1 - Ensuring 5 Concurrent COV Subscribers</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.10.4 Supports COV for Analog Input Objects

The IUT supports change of value notifications for at least one object of type Analog Input.

<b>BTL - 8.2.1 - Change of Value Notification from <del>an Analog Input, Analog Output, and Analog Value</del> a Numeric Object's Present Value Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.3.1 is executed against an Analog Input object.
	<b>Test Directives</b>	The selected object must be an Analog Input.
	<b>Testing Hints</b>	
<b>BTL - 8.2.2 - Change of Value Notification from <del>an Analog Input, Analog Output, and Analog Value</del> a Numeric Object's Status Flags Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.3.2 is executed against an Analog Input object.
	<b>Test Directives</b>	The selected object must be an Analog Input.
	<b>Testing Hints</b>	
<b>BTL - 8.3.1 - Change of Value Notification from <del>an Analog Input, Analog Output, and Analog Value</del> a Numeric Object's Present Value Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.2.1 is executed against an Analog Input object.
	<b>Test Directives</b>	The selected object must be an Analog Input.
	<b>Testing Hints</b>	
<b>BTL - 8.3.2 - Change of Value Notification from <del>an Analog Input, Analog Output, and Analog Value</del> a Numeric Object's Status Flags Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.2.2 is executed against an Analog Input object.
	<b>Test Directives</b>	The selected object must be an Analog Input.
	<b>Testing Hints</b>	

#### 4.10.5 Supports COV for Analog Output Objects

The IUT supports change of value notifications for at least one object of type Analog Output.

<b>BTL - 8.2.1 - Change of Value Notification from <del>an Analog Input, Analog Output, and Analog Value</del>, a Numeric Object's Present Value Property</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	The selected object must be an Analog Output.
	<b>Testing Hints</b>	This may be skipped if 8.3.1 is executed against an Analog Output object.
<b>BTL - 8.2.2 - Change of Value Notification from <del>an Analog Input, Analog Output, and Analog Value</del>, a Numeric Object's Status Flags Property</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	The selected object must be an Analog Output.
	<b>Testing Hints</b>	This may be skipped if 8.3.2 is executed against an Analog Output object.
<b>BTL - 8.3.1 - Change of Value Notification from <del>an Analog Input, Analog Output, and Analog Value</del>, a Numeric Object's Present Value Property</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	The selected object must be an Analog Output.
	<b>Testing Hints</b>	This may be skipped if 8.2.1 is executed against an Analog Output object.
<b>BTL - 8.3.2 - Change of Value Notification from <del>an Analog Input, Analog Output, and Analog Value</del>, a Numeric Object's Status Flags Property</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	The selected object must be an Analog Output.
	<b>Testing Hints</b>	This may be skipped if 8.2.2 is executed against an Analog Output object.

#### 4.10.6 Supports COV for Analog Value Objects

The IUT supports change of value notifications for at least one object of type Analog Value.

<b>BTL - 8.2.1 - Change of Value Notification from <del>an Analog Input, Analog Output, and Analog Value</del>, a Numeric Object's Present Value Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.3.1 is executed against an Analog Value object.
	<b>Test Directives</b>	The selected object must be an Analog Value.
	<b>Testing Hints</b>	
<b>BTL - 8.2.2 - Change of Value Notification from <del>an Analog Input, Analog Output, and Analog Value</del>, a Numeric Object's Status Flags Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.3.2 is executed against an Analog Value object.
	<b>Test Directives</b>	The selected object must be an Analog Value.
	<b>Testing Hints</b>	
<b>BTL - 8.3.1 - Change of Value Notification from <del>an Analog Input, Analog Output, and Analog Value</del>, a Numeric Object's Present Value Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.2.1 is executed against an Analog Value object.
	<b>Test Directives</b>	The selected object must be an Analog Value.
	<b>Testing Hints</b>	
<b>BTL - 8.3.2 - Change of Value Notification from <del>an Analog Input, Analog Output, and Analog Value</del>, a Numeric Object's Status Flags Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.2.2 is executed against an Analog Value object.
	<b>Test Directives</b>	The selected object must be an Analog Value.
	<b>Testing Hints</b>	



#### 4.10.7 Supports COV for Binary Input Objects

The IUT supports change of value notifications for at least one object of type Binary Input.

<b>BTL - 8.2.3 - Change of Value Notification from a <del>Binary Input, Binary Output, and Binary Value</del> Discrete Valued Object Present Value Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.3.3 is executed against a Binary Input object.
	<b>Test Directives</b>	The selected object must be a Binary Input.
	<b>Testing Hints</b>	
<b>BTL - 8.2.4 - Change of Value Notification from a <del>Binary Input, Binary Output, and Binary Value</del> Discrete Valued Object Status Flags Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.3.4 is executed against a Binary Input object.
	<b>Test Directives</b>	The selected object must be a Binary Input.
	<b>Testing Hints</b>	
<b>BTL - 8.3.3 - Change of Value Notification from a <del>Binary Input, Binary Output, and Binary Value</del> Discrete Valued Object Present Value Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.2.3 is executed against a Binary Input object.
	<b>Test Directives</b>	The selected object must be a Binary Input.
	<b>Testing Hints</b>	
<b>BTL - 8.3.4 - Change of Value Notification from a <del>Binary Input, Binary Output, and Binary Value</del> Discrete Valued Object Status Flags Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.2.4 is executed against a Binary Input object.
	<b>Test Directives</b>	The selected object must be a Binary Input.
	<b>Testing Hints</b>	

#### 4.10.8 Supports COV for Binary Output Objects

The IUT supports change of value notifications for at least one object of type Binary Output.

<b>BTL - 8.2.3 - Change of Value Notification from a <del>Binary Input, Binary Output, and Binary Value</del> Discrete Valued Object Present Value Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.3.3 is executed against a Binary Output object.
	<b>Test Directives</b>	The selected object must be a Binary Output.
	<b>Testing Hints</b>	
<b>BTL - 8.2.4 - Change of Value Notification from a <del>Binary Input, Binary Output, and Binary Value</del> Discrete Valued Object Status Flags Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.3.4 is executed against a Binary Output object.
	<b>Test Directives</b>	The selected object must be a Binary Output.
	<b>Testing Hints</b>	
<b>BTL - 8.3.3 - Change of Value Notification from a <del>Binary Input, Binary Output, and Binary Value</del> Discrete Valued Object Present Value Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.2.3 is executed against a Binary Output object.
	<b>Test Directives</b>	The selected object must be a Binary Output.
	<b>Testing Hints</b>	
<b>BTL - 8.3.4 - Change of Value Notification from a <del>Binary Input, Binary Output, and Binary Value</del> Discrete Valued Object Status Flags Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.2.4 is executed against a Binary Output object.
	<b>Test Directives</b>	The selected object must be a Binary Output.
	<b>Testing Hints</b>	

#### 4.10.9 Supports COV for Binary Value Objects

The IUT supports change of value notifications for at least one object of type Binary Value.

<b>BTL - 8.2.3 - Change of Value Notification from a <del>Binary Input, Binary Output, and Binary Value</del> Discrete Valued Object Present Value Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.3.3 is executed against a Binary Value object.

	<b>Test Directives</b>	The selected object must be a Binary Value.
	<b>Testing Hints</b>	
<b>BTL - 8.2.4 - Change of Value Notification from a <del>Binary Input, Binary Output, and Binary Value</del> Discrete Valued Object Status Flags Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.3.4 is executed against a Binary Value object.
	<b>Test Directives</b>	The selected object must be a Binary Value.
	<b>Testing Hints</b>	
<b>BTL - 8.3.3 - Change of Value Notification from a <del>Binary Input, Binary Output, and Binary Value</del> Discrete Valued Object Present Value Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.2.3 is executed against a Binary Value object.
	<b>Test Directives</b>	The selected object must be a Binary Value.
	<b>Testing Hints</b>	
<b>BTL - 8.3.4 - Change of Value Notification from a <del>Binary Input, Binary Output, and Binary Value</del> Discrete Valued Object Status Flags Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.2.4 is executed against a Binary Value object.
	<b>Test Directives</b>	The selected object must be a Binary Value.
	<b>Testing Hints</b>	

#### 4.10.10 Supports COV for Life Safety Point Objects

The IUT supports change of value notifications for at least one object of type Life Safety Point.

<b>BTL - 8.2.5 - Change of Value Notification from a <del>Multi-state Input, Multi-state Output, Multi-state Value, Life Safety Point, and Life Safety Zone</del> Multi-state or Other Discrete Valued Object Present Value Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.3.5 is executed against a Life Safety Point object.
	<b>Test Directives</b>	The selected object must be a Life Safety Point.
	<b>Testing Hints</b>	
<b>BTL - 8.2.6 - Change of Value Notification from a <del>Multi-state Input, Multi-state Output, Multi-state Value, Life Safety Point, and Life Safety Zone</del> Multi-state or Other Discrete Valued Object Status Flags Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.3.6 is executed against a Life Safety Point object.
	<b>Test Directives</b>	The selected object must be a Life Safety Point.
	<b>Testing Hints</b>	
<b>BTL - 8.3.5 - Change of Value Notification from a <del>Multi-state Input, Multi-state Output, Multi-state Value, Life Safety Point, and Life Safety Zone</del> Multi-state or Other Discrete Valued Object Present Value Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.2.5 is executed against a Life Safety Point object.
	<b>Test Directives</b>	The selected object must be a Life Safety Point.
	<b>Testing Hints</b>	
<b>BTL - 8.3.6 - Change of Value Notification from a <del>Multi-state Input, Multi-state Output, Multi-state Value, Life Safety Point, and Life Safety Zone</del> Multi-state or Other Discrete Valued Object Status Flags Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.2.6 is executed against a Life Safety Point object.
	<b>Test Directives</b>	The selected object must be a Life Safety Point.
	<b>Testing Hints</b>	

#### 4.10.11 Supports COV for Life Safety Zone Objects

The IUT supports change of value notifications for at least one object of type Life Safety Zone.

<b>BTL - 8.2.5 - Change of Value Notification from a <del>Multi-state Input, Multi-state Output, Multi-state Value, Life Safety Point, and Life Safety Zone</del> Multi-state or Other Discrete Valued Object Present Value Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.3.5 is executed against a Life Safety Zone object.
	<b>Test Directives</b>	The selected object must be a Life Safety Zone.
	<b>Testing Hints</b>	
<b>BTL - 8.2.6 - Change of Value Notification from a <del>Multi-state Input, Multi-state Output, Multi-state Value, Life Safety Point, and Life Safety Zone</del> Multi-state or Other Discrete Valued Object Status Flags Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.3.6 is executed against a Life Safety Zone object.
	<b>Test Directives</b>	The selected object must be a Life Safety Zone.
	<b>Testing Hints</b>	
<b>BTL - 8.3.5 - Change of Value Notification from a <del>Multi-state Input, Multi-state Output, Multi-state Value, Life Safety Point, and Life Safety Zone</del> Multi-state or Other Discrete Valued Object Present Value Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.2.5 is executed against a Life Safety Zone object.
	<b>Test Directives</b>	The selected object must be a Life Safety Zone.
	<b>Testing Hints</b>	
<b>BTL - 8.3.6 - Change of Value Notification from a <del>Multi-state Input, Multi-state Output, Multi-state Value, Life Safety Point, and Life Safety Zone</del> Multi-state or Other Discrete Valued Object Status Flags Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.2.6 is executed against a Life Safety Zone object.
	<b>Test Directives</b>	The selected object must be a Life Safety Zone.
	<b>Testing Hints</b>	

#### 4.10.12 Supports COV for Loop Objects

The IUT supports change of value notifications for at least one object of type Loop.

<b>BTL - 8.2.7 - Change of Value Notification from Loop Object Present Value Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.3.7 is executed against a Loop object.
	<b>Test Directives</b>	The selected object must be a Loop.
	<b>Testing Hints</b>	
<b>BTL - 8.2.8 - Change of Value Notification from a Loop Object Status Flags Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.3.8 is executed against a Loop object.
	<b>Test Directives</b>	The selected object must be a Loop.
	<b>Testing Hints</b>	
<b>BTL - 8.3.7 - Change of Value Notification from Loop Object Present Value Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.2.7 is executed against a Loop object.
	<b>Test Directives</b>	The selected object must be a Loop
	<b>Testing Hints</b>	
<b>135.1-2013 - 8.3.8 - Change of Value Notification from a Loop Object Status Flags Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.2.8 is executed against a Loop object.
	<b>Test Directives</b>	The selected object must be a Loop.
	<b>Testing Hints</b>	

#### 4.10.13 Supports COV for Multi-state Input Objects

The IUT supports change of value notifications for at least one object of type Multi-state Input.

<b>BTL - 8.2.5 - Change of Value Notification from a <del>Multi-state Input, Multi-state Output, Multi-state Value, Life Safety Point, and Life Safety Zone</del> Multi-state or Other Discrete Valued Object Present Value Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.3.5 is executed against a Multi-state Input object.
	<b>Test Directives</b>	The selected object must be a Multi-state Input.
	<b>Testing Hints</b>	
<b>BTL - 8.2.6 - Change of Value Notification from a <del>Multi-state Input, Multi-state Output, Multi-state Value, Life Safety Point, and Life Safety Zone</del> Multi-state or Other Discrete Valued Object Status Flags Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.3.6 is executed against a Multi-state Input object.
	<b>Test Directives</b>	The selected object must be a Multi-state Input.
	<b>Testing Hints</b>	
<b>BTL - 8.3.5 - Change of Value Notification from a <del>Multi-state Input, Multi-state Output, Multi-state Value, Life Safety Point, and Life Safety Zone</del> Multi-state or Other Discrete Valued Object Present Value Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.2.5 is executed against a Multi-state Input object.
	<b>Test Directives</b>	The selected object must be a Multi-state Input.
	<b>Testing Hints</b>	
<b>BTL - 8.3.6 - Change of Value Notification from a <del>Multi-state Input, Multi-state Output, Multi-state Value, Life Safety Point, and Life Safety Zone</del> Multi-state or Other Discrete Valued Object Status Flags Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.2.6 is executed against a Multi-state Input object.
	<b>Test Directives</b>	The selected object must be a Multi-state Input.
	<b>Testing Hints</b>	

#### 4.10.14 Supports COV for Multi-state Output Objects

The IUT supports change of value notifications for at least one object of type Multi-state Output.

<b>BTL - 8.2.5 - Change of Value Notification from a <del>Multi-state Input, Multi-state Output, Multi-state Value, Life Safety Point, and Life Safety Zone</del> Multi-state or Other Discrete Valued Object Present Value Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.3.5 is executed against a Multi-state Output object.
	<b>Test Directives</b>	The selected object must be a Multi-state Output.
	<b>Testing Hints</b>	
<b>BTL - 8.2.6 - Change of Value Notification from a <del>Multi-state Input, Multi-state Output, Multi-state Value, Life Safety Point, and Life Safety Zone</del> Multi-state or Other Discrete Valued Object Status Flags Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.3.6 is executed against a Multi-state Output object.
	<b>Test Directives</b>	The selected object must be a Multi-state Output.
	<b>Testing Hints</b>	
<b>BTL - 8.3.5 - Change of Value Notification from a <del>Multi-state Input, Multi-state Output, Multi-state Value, Life Safety Point, and Life Safety Zone</del> Multi-state or Other Discrete valued Object Present Value Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.2.5 is executed against a Multi-state Output object.
	<b>Test Directives</b>	The selected object must be a Multi-state Output.
	<b>Testing Hints</b>	

<b>BTL - 8.3.6 - Change of Value Notification from a <del>Multi-state Input, Multi-state Output, Multi-state Value, Life Safety Point, and Life Safety Zone</del> Multi-state or Other Discrete Valued Object Status Flags Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.2.6 is executed against a Multi-state Output object.
	<b>Test Directives</b>	The selected object must be a Multi-state Output.
	<b>Testing Hints</b>	

#### 4.10.15 Supports COV for Multi-state Value Objects

The IUT supports change of value notifications for at least one object of type Multi-state Value.

<b>BTL - 8.2.5 - Change of Value Notification from a <del>Multi-state Input, Multi-state Output, Multi-state Value, Life Safety Point, and Life Safety Zone</del> Multi-state or Other Discrete Valued Object Present Value Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.3.5 is executed against a Multi-state Value object.
	<b>Test Directives</b>	The selected object must be a Multi-state Value.
	<b>Testing Hints</b>	
<b>BTL - 8.2.6 - Change of Value Notification from a <del>Multi-state Input, Multi-state Output, Multi-state Value, Life Safety Point, and Life Safety Zone</del> Multi-state or Other Discrete Valued Object Status Flags Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.3.6 is executed against a Multi-state Value object.
	<b>Test Directives</b>	The selected object must be a Multi-state Value.
	<b>Testing Hints</b>	
<b>BTL - 8.3.5 - Change of Value Notification from a <del>Multi-state Input, Multi-state Output, Multi-state Value, Life Safety Point, and Life Safety Zone</del> Multi-state or Other Discrete Valued Object Present Value Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.2.5 is executed against a Multi-state Value object.
	<b>Test Directives</b>	The selected object must be a Multi-state Value.
	<b>Testing Hints</b>	
<b>BTL - 8.3.6 - Change of Value Notification from a <del>Multi-state Input, Multi-state Output, Multi-state Value, Life Safety Point, and Life Safety Zone</del> Multi-state or Other Discrete Valued Object Status Flags Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.2.6 is executed against a Multi-state Value object.
	<b>Test Directives</b>	The selected object must be a Multi-state Value.
	<b>Testing Hints</b>	

#### 4.10.16 Supports COV for CharacterString Value Objects

The IUT supports change of value notifications for at least one object of type CharacterString Value.

<b>BTL - 8.2.5 - Change of Value Notification from a <del>Multi-state Input, Multi-state Output, Multi-state Value, Life Safety Point, and Life Safety Zone</del> Multi-state or Other Discrete Valued Object Present Value Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.3.5 is executed against a CharacterString Value object.
	<b>Test Directives</b>	The selected object must be a CharacterString Value.
	<b>Testing Hints</b>	
<b>BTL - 8.2.6 - Change of Value Notification from a <del>Multi-state Input, Multi-state Output, Multi-state Value, Life Safety Point, and Life Safety Zone</del> Multi-state or Other Discrete Valued Object Status Flags Property</b>		

	<b>Test Conditionality</b>	This may be skipped if 8.3.6 is executed against a CharacterString Value object.
	<b>Test Directives</b>	The selected object must be a CharacterString Value.
	<b>Testing Hints</b>	
<b>BTL - 8.3.5 - Change of Value Notification from a <del>Multi-state Input, Multi-state Output, Multi-state Value, Life Safety Point, and Life Safety Zone</del> Multi-state or Other Discrete Valued Object Present Value Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.2.5 is executed against a CharacterString Output object.
	<b>Test Directives</b>	The selected object must be a CharacterString Value.
	<b>Testing Hints</b>	
<b>BTL - 8.3.6 - Change of Value Notification from a <del>Multi-state Input, Multi-state Output, Multi-state Value, Life Safety Point, and Life Safety Zone</del> Multi-state or Other Discrete Valued Object Status Flags Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.2.6 is executed against a CharacterString Value object.
	<b>Test Directives</b>	The selected object must be a CharacterString Value.
	<b>Testing Hints</b>	

#### 4.10.17 Supports COV for Date Value Objects

The IUT supports change of value notifications for at least one object of type Date Value.

<b>BTL - 8.2.5 - Change of Value Notification from a <del>Multi-state Input, Multi-state Output, Multi-state Value, Life Safety Point, and Life Safety Zone</del> Multi-state or Other Discrete Valued Object Present Value Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.3.5 is executed against a Date Value object.
	<b>Test Directives</b>	The selected object must be a Date Value.
	<b>Testing Hints</b>	
<b>BTL - 8.2.6 - Change of Value Notification from a <del>Multi-state Input, Multi-state Output, Multi-state Value, Life Safety Point, and Life Safety Zone</del> Multi-state or Other Discrete Valued Object Status Flags Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.3.6 is executed against a Date Value object.
	<b>Test Directives</b>	The selected object must be a Date Value.
	<b>Testing Hints</b>	
<b>BTL - 8.3.5 - Change of Value Notification from a <del>Multi-state Input, Multi-state Output, Multi-state Value, Life Safety Point, and Life Safety Zone</del> Multi-state or Other Discrete Valued Object Present Value Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.2.5 is executed against a Date Output object.
	<b>Test Directives</b>	The selected object must be a Date Value.
	<b>Testing Hints</b>	
<b>BTL - 8.3.6 - Change of Value Notification from a <del>Multi-state Input, Multi-state Output, Multi-state Value, Life Safety Point, and Life Safety Zone</del> Multi-state or Other Discrete Valued Object Status Flags Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.2.6 is executed against a Date Value object.
	<b>Test Directives</b>	The selected object must be a Date Value.
	<b>Testing Hints</b>	

#### 4.10.18 Supports COV for Date Pattern Value Objects

The IUT supports change of value notifications for at least one object of type Date Pattern Value.

<b>BTL - 8.2.5 - Change of Value Notification from a <del>Multi-state Input, Multi-state Output, Multi-state Value, Life Safety Point, and Life Safety Zone</del> Multi-state or Other Discrete Valued Object Present Value Property</b>		
--	--	--

	<b>Test Conditionality</b>	This may be skipped if 8.3.5 is executed against a Date Pattern Value object.
	<b>Test Directives</b>	The selected object must be a Date Pattern Value.
	<b>Testing Hints</b>	
<b>BTL - 8.2.6 - Change of Value Notification from a <del>Multi-state Input, Multi-state Output, Multi-state Value, Life Safety Point, and Life Safety Zone</del> Multi-state or Other Discrete Valued Object Status Flags Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.3.6 is executed against a Date Pattern Value object.
	<b>Test Directives</b>	The selected object must be a Date Pattern Value.
	<b>Testing Hints</b>	
<b>BTL - 8.3.5 - Change of Value Notification from a <del>Multi-state Input, Multi-state Output, Multi-state Value, Life Safety Point, and Life Safety Zone</del> Multi-state or Other Discrete Valued Object Present Value Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.2.5 is executed against a Date Pattern Output object.
	<b>Test Directives</b>	The selected object must be a Date Pattern Value.
	<b>Testing Hints</b>	
<b>BTL - 8.3.6 - Change of Value Notification from a <del>Multi-state Input, Multi-state Output, Multi-state Value, Life Safety Point, and Life Safety Zone</del> Multi-state or Other Discrete Valued Object Status Flags Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.2.6 is executed against a Date Pattern Value object.
	<b>Test Directives</b>	The selected object must be a Date Pattern Value.
	<b>Testing Hints</b>	

#### 4.10.19 Supports COV for DateTime Value Objects

The IUT supports change of value notifications for at least one object of type DateTime Value.

<b>BTL - 8.2.5 - Change of Value Notification from a <del>Multi-state Input, Multi-state Output, Multi-state Value, Life Safety Point, and Life Safety Zone</del> Multi-state or Other Discrete Valued Object Present Value Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.3.5 is executed against a DateTime Value object.
	<b>Test Directives</b>	The selected object must be a DateTime Value.
	<b>Testing Hints</b>	
<b>BTL - 8.2.6 - Change of Value Notification from a <del>Multi-state Input, Multi-state Output, Multi-state Value, Life Safety Point, and Life Safety Zone</del> Multi-state or Other Discrete Valued Object Status Flags Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.3.6 is executed against a DateTime Value object.
	<b>Test Directives</b>	The selected object must be a DateTime Value.
	<b>Testing Hints</b>	
<b>BTL - 8.3.5 - Change of Value Notification from a <del>Multi-state Input, Multi-state Output, Multi-state Value, Life Safety Point, and Life Safety Zone</del> Multi-state or Other Discrete Valued Object Present Value Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.2.5 is executed against a DateTime Output object.
	<b>Test Directives</b>	The selected object must be a DateTime Value.
	<b>Testing Hints</b>	
<b>BTL - 8.3.6 - Change of Value Notification from a <del>Multi-state Input, Multi-state Output, Multi-state Value, Life Safety Point, and Life Safety Zone</del> Multi-state or Other Discrete Valued Object Status Flags Property</b>		

	<b>Test Conditionality</b>	This may be skipped if 8.2.6 is executed against a DateTime Value object.
	<b>Test Directives</b>	The selected object must be a DateTime Value.
	<b>Testing Hints</b>	

#### 4.10.20 Supports COV for DateTime Pattern Value Objects

The IUT supports change of value notifications for at least one object of type DateTime Pattern Value.

<b>BTL - 8.2.5 - Change of Value Notification from a <del>Multi-state Input, Multi-state Output, Multi-state Value, Life Safety Point, and Life Safety Zone</del> Multi-state or Other Discrete Valued Object Present Value Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.3.5 is executed against a DateTime Pattern Value object.
	<b>Test Directives</b>	The selected object must be a DateTime Pattern Value.
	<b>Testing Hints</b>	
<b>BTL - 8.2.6 - Change of Value Notification from a <del>Multi-state Input, Multi-state Output, Multi-state Value, Life Safety Point, and Life Safety Zone</del> Multi-state or Other Discrete Valued Object Status Flags Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.3.6 is executed against a DateTime Pattern Value object.
	<b>Test Directives</b>	The selected object must be a DateTime Pattern Value.
	<b>Testing Hints</b>	
<b>BTL - 8.3.5 - Change of Value Notification from a <del>Multi-state Input, Multi-state Output, Multi-state Value, Life Safety Point, and Life Safety Zone</del> Multi-state or Other Discrete Valued Object Present Value Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.2.5 is executed against a DateTime Pattern Output object.
	<b>Test Directives</b>	The selected object must be a DateTime Pattern Value.
	<b>Testing Hints</b>	
<b>BTL - 8.3.6 - Change of Value Notification from a <del>Multi-state Input, Multi-state Output, Multi-state Value, Life Safety Point, and Life Safety Zone</del> Multi-state or Other Discrete Valued Object Status Flags Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.2.6 is executed against a DateTime Pattern Value object.
	<b>Test Directives</b>	The selected object must be a DateTime Pattern Value.
	<b>Testing Hints</b>	

#### 4.10.21 Supports COV for Integer Value Objects

The IUT supports change of value notifications for at least one object of type Integer Value.

<b>BTL - 8.2.1 - Change of Value Notification from an <del>Analog Input, Analog Output, and Analog Value</del> Numeric Object's Present Value Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.3.1 is executed against an Integer Value object.
	<b>Test Directives</b>	The selected object must be an Integer Value.
	<b>Testing Hints</b>	
<b>BTL - 8.2.2 - Change of Value Notification from an <del>Analog Input, Analog Output, and Analog Value</del> Numeric Object's Status Flags Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.3.2 is executed against an Integer Value object.
	<b>Test Directives</b>	The selected object must be an Integer Value.
	<b>Testing Hints</b>	
<b>BTL - 8.3.1 - Change of Value Notification from an <del>Analog Input, Analog Output, and Analog Value</del>, a Numeric Object's Present Value Property</b>		



	<b>Test Conditionality</b>	This may be skipped if 8.2.1 is executed against an Integer Value object.
	<b>Test Directives</b>	The selected object must be an Integer Value.
	<b>Testing Hints</b>	
<b>BTL - 8.3.2 - Change of Value Notification from <del>an Analog Input, Analog Output, and Analog Value</del>, a Numeric Object's Status Flags Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.2.2 is executed against an Integer Value object.
	<b>Test Directives</b>	The selected object must be an Integer Value.
	<b>Testing Hints</b>	

#### 4.10.22 Supports COV for Large Analog Value Objects

The IUT supports change of value notifications for at least one object of type Large Analog Value.

<b>BTL - 8.2.1 - Change of Value Notification from <del>an Analog Input, Analog Output, and Analog Value</del>, a Numeric Object's Present Value Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.3.1 is executed against a Large Analog Value object.
	<b>Test Directives</b>	The selected object must be a Large Analog Value.
	<b>Testing Hints</b>	
<b>BTL - 8.2.2 - Change of Value Notification from <del>an Analog Input, Analog Output, and Analog Value</del>, a Numeric Object's Status Flags Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.3.2 is executed against a Large Analog Value object.
	<b>Test Directives</b>	The selected object must be a Large Analog Value.
	<b>Testing Hints</b>	
<b>BTL - 8.3.1 - Change of Value Notification from <del>an Analog Input, Analog Output, and Analog Value</del>, a Numeric Object's Present Value Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.2.1 is executed against a Large Analog Value object.
	<b>Test Directives</b>	The selected object must be a Large Analog Value.
	<b>Testing Hints</b>	
<b>BTL - 8.3.2 - Change of Value Notification from <del>an Analog Input, Analog Output, and Analog Value</del>, a Numeric Object's Status Flags Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.2.2 is executed against a Large Analog Value object.
	<b>Test Directives</b>	The selected object must be a Large Analog Value.
	<b>Testing Hints</b>	

#### 4.10.23 Supports COV for Positive Integer Value Objects

The IUT supports change of value notifications for at least one object of type Positive Integer Value.

<b>BTL - 8.2.1 - Change of Value Notification from <del>an Analog Input, Analog Output, and Analog Value</del>, a Numeric Object's Present Value Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.3.1 is executed against a Positive Integer Value object.
	<b>Test Directives</b>	The selected object must be a Positive Integer Value.
	<b>Testing Hints</b>	
<b>BTL - 8.2.2 - Change of Value Notification from <del>an Analog Input, Analog Output, and Analog Value</del>, a Numeric Object's Status Flags Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.3.2 is executed against a Positive Integer Value object.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

	Test Directives	The selected object must be a Positive Integer Value.
	Testing Hints	
<b>BTL - 8.3.1 - Change of Value Notification from an <del>Analog Input, Analog Output, and Analog Value</del>, a Numeric Object's Present Value Property</b>		
	Test Conditionality	This may be skipped if 8.2.1 is executed against a Positive Integer Value object.
	Test Directives	The selected object must be a Positive Integer Value.
	Testing Hints	
<b>BTL - 8.3.2 - Change of Value Notification from an <del>Analog Input, Analog Output, and Analog Value</del>, a Numeric Object's Status Flags Property</b>		
	Test Conditionality	This may be skipped if 8.2.2 is executed against a Positive Integer Value object.
	Test Directives	The selected object must be a Positive Integer Value.
	Testing Hints	

#### 4.10.24 Supports COV for Time Value Objects

The IUT supports change of value notifications for at least one object of type Time Value.

<b>BTL - 8.2.5 - Change of Value Notification from a <del>Multi-state Input, Multi-state Output, Multi-state Value, Life Safety Point, and Life Safety Zone</del> Multi-state or Other Discrete Valued Object Present Value Property</b>		
	Test Conditionality	This may be skipped if 8.3.5 is executed against a Time Value object.
	Test Directives	The selected object must be a Time Value.
	Testing Hints	
<b>BTL - 8.2.6 - Change of Value Notification from a <del>Multi-state Input, Multi-state Output, Multi-state Value, Life Safety Point, and Life Safety Zone</del> Multi-state or Other Discrete Valued Object Status Flags Property</b>		
	Test Conditionality	This may be skipped if 8.3.6 is executed against a Time Value object.
	Test Directives	The selected object must be a Time Value.
	Testing Hints	
<b>BTL - 8.3.5 - Change of Value Notification from a <del>Multi-state Input, Multi-state Output, Multi-state Value, Life Safety Point, and Life Safety Zone</del> Multi-state or Other Discrete Valued Object Present Value Property</b>		
	Test Conditionality	This may be skipped if 8.2.5 is executed against a Time Value object.
	Test Directives	The selected object must be a Time Value.
	Testing Hints	
<b>BTL - 8.3.6 - Change of Value Notification from a <del>Multi-state Input, Multi-state Output, Multi-state Value, Life Safety Point, and Life Safety Zone</del> Multi-state or Other Discrete Valued Object Status Flags Property</b>		
	Test Conditionality	This may be skipped if 8.2.6 is executed against a Time Value object.
	Test Directives	The selected object must be a Time Value.
	Testing Hints	

#### 4.10.25 Supports COV for Time Pattern Value Objects

The IUT supports change of value notifications for at least one object of type Time Pattern Value.

<b>BTL - 8.2.5 - Change of Value Notification from a <del>Multi-state Input, Multi-state Output, Multi-state Value, Life Safety Point, and Life Safety Zone</del> Multi-state or Other Discrete Valued Object Present Value Property</b>		
	Test Conditionality	This may be skipped if 8.3.5 is executed against a Time Pattern Value object.
	Test Directives	The selected object must be a Time Pattern Value.
	Testing Hints	

<b>BTL - 8.2.6 - Change of Value Notification from a <del>Multi-state Input, Multi-state Output, Multi-state Value, Life Safety Point, and Life Safety Zone</del> Multi-state or Other Discrete Valued Object Status Flags Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.3.6 is executed against a Time Pattern Value object.
	<b>Test Directives</b>	The selected object must be a Time Pattern Value.
	<b>Testing Hints</b>	
<b>BTL - 8.3.5 - Change of Value Notification from a <del>Multi-state Input, Multi-state Output, Multi-state Value, Life Safety Point, and Life Safety Zone</del> Multi-state or Other Discrete Valued Object Present Value Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.2.5 is executed against a Time Pattern Value object.
	<b>Test Directives</b>	The selected object must be a Time Pattern Value.
	<b>Testing Hints</b>	
<b>BTL - 8.3.6 - Change of Value Notification from a <del>Multi-state Input, Multi-state Output, Multi-state Value, Life Safety Point, and Life Safety Zone</del> Multi-state or Other Discrete Valued Object Status Flags Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.2.6 is executed against a Time Pattern Value object.
	<b>Test Directives</b>	The selected object must be a Time Pattern Value.
	<b>Testing Hints</b>	

#### 4.10.26 Supports COV for OctetString Value Objects

The IUT supports change of value notifications for at least one object of type OctetString Value.

<b>BTL - 8.2.5 - Change of Value Notification from a <del>Multi-state Input, Multi-state Output, Multi-state Value, Life Safety Point, and Life Safety Zone</del> Multi-state or Other Discrete Valued Object Present Value Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.3.5 is executed against an OctetString Value object.
	<b>Test Directives</b>	The selected object must be an OctetString Value.
	<b>Testing Hints</b>	
<b>BTL - 8.2.6 - Change of Value Notification from a <del>Multi-state Input, Multi-state Output, Multi-state Value, Life Safety Point, and Life Safety Zone</del> Multi-state or Other Discrete Valued Object Status Flags Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.3.6 is executed against an OctetString Value object.
	<b>Test Directives</b>	The selected object must be an OctetString Value.
	<b>Testing Hints</b>	
<b>BTL - 8.3.5 - Change of Value Notification from a <del>Multi-state Input, Multi-state Output, Multi-state Value, Life Safety Point, and Life Safety Zone</del> Multi-state or Other Discrete Valued Object Present Value Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.2.5 is executed against an OctetString Value object.
	<b>Test Directives</b>	The selected object must be an OctetString Value.
	<b>Testing Hints</b>	
<b>BTL - 8.3.6 - Change of Value Notification from a <del>Multi-state Input, Multi-state Output, Multi-state Value, Life Safety Point, and Life Safety Zone</del> Multi-state or Other Discrete Valued Object Status Flags Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.2.6 is executed against an OctetString Value object.
	<b>Test Directives</b>	The selected object must be an OctetString Value.
	<b>Testing Hints</b>	

#### 4.10.27 Supports COV for Pulse Converter Objects

The IUT supports change of value notifications for at least one object of type Pulse Converter.

<b>BTL - 8.2.X9 - ConfirmedCOVNotification Pulse Converter changing Present_Value</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 8.2.X10 - ConfirmedCOVNotification Pulse Converter changing Status_Flags</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 8.3.X12 - UnconfirmedCOVNotification Pulse Converter changing Present_Value</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 8.3.X13 - UnconfirmedCOVNotification Pulse Converter changing Status_Flags</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.10.28 Supports COV for Access Door Objects

The IUT supports change of value notifications for at least one object of type Access Door.

<b>BTL - 8.2.X11 - Change of Value Notification from an Access Door object Present_Value, Status_Flags and Door_Alarm_State property</b>		
	<b>Test Conditionality</b>	This may be skipped if BTL - 8.3.X14 is executed against an Access Door object.
	<b>Test Directives</b>	Test at least one instance where object type is an Access Door.
	<b>Testing Hints</b>	
<b>BTL - 8.3.X14 - Change of Value Notification from an Access Door object Present_Value, Status_Flags and Door_Alarm_State property</b>		
	<b>Test Conditionality</b>	This may be skipped if BTL - 8.2.X11 is executed against an Access Door object.
	<b>Test Directives</b>	Test at least one instance where object type is an Access Door.
	<b>Testing Hints</b>	

#### 4.10.29 Supports COV for Load Control Objects

The IUT supports change of value notifications for at least one object of type Load Control.

Contact BTL for interim tests for this object.

#### 4.10.30 Supports COV for Access Point Objects

The IUT supports change of value notifications for at least one object of type Access Point.

Contact BTL for interim tests for this object.

#### 4.10.31 Supports COV for Credential Data Input Objects

The IUT supports change of value notifications for at least one object of type Credential Data Input.

Contact BTL for interim tests for this object.

#### 4.10.32 Supports COV for Lighting Output Objects

The IUT supports change of value notifications for at least one object of type Lighting Output.

<b>BTL - 8.2.1 - Change of Value Notification from an <del>Analog Input, Analog Output, Analog Value</del> Numeric Object's Present Value Property</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	This may be skipped if 8.3.1 is executed against a Lighting Output object.
<b>BTL - 8.2.2 - Change of Value Notification from an <del>Analog Input, Analog Output, Analog Value</del> Numeric Object's Status Flags Property</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	The selected object must be a Lighting Output.
	<b>Testing Hints</b>	This may be skipped if 8.3.2 is executed against a Lighting Output object.
<b>BTL - 8.3.1 - Change of Value Notification from a Numeric Object's Present Value Property</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	The selected object must be a Lighting Output.
	<b>Testing Hints</b>	This may be skipped if 8.2.1 is executed against a Lighting Output object.
<b>BTL - 8.3.2 - Change of Value Notification from a Numeric Object's Status Flags Property</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	The selected object must be a Lighting Output.
	<b>Testing Hints</b>	This may be skipped if 8.2.2 is executed against a Lighting Output object.

#### 4.10.33 Supports COV for Binary Lighting Output Objects

The IUT supports change of value notifications for at least one object of type Binary Lighting Output.

<b>BTL - 8.2.3 - Change of Value Notification from a Discrete Valued Object Present Value Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.3.3 is executed against a Binary Lighting Output object.
	<b>Test Directives</b>	The selected object must be a Binary Lighting Output object.
	<b>Testing Hints</b>	
<b>BTL - 8.2.4 - Change of Value Notification from a Discrete Valued Object Status Flags Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.3.4 is executed against a Binary Lighting Output object.
	<b>Test Directives</b>	The selected object must be a Binary Lighting Output object.
	<b>Testing Hints</b>	
<b>BTL - 8.3.3 - Change of Value Notification from a Discrete Valued Object Present Value Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.2.3 is executed against a Binary Lighting Output object.
	<b>Test Directives</b>	The selected object must be a Binary Lighting Output object.
	<b>Testing Hints</b>	
<b>BTL - 8.3.4 - Change of Value Notification from a Discrete Valued Object Status Flags Property</b>		
	<b>Test Conditionality</b>	This may be skipped if 8.2.4 is executed against a Binary Lighting Output object.
	<b>Test Directives</b>	The selected object must be a Binary Lighting Output object.
	<b>Testing Hints</b>	

#### 4.10.34 Supports COV for Staging Objects

The IUT supports change of value notifications for at least one object of type Staging.

Contact BTL for interim tests for this object.

#### 4.10.35 Supports COV for Proprietary Objects

The IUT supports change of value notifications for at least one proprietary object.

There is no test defined for this functionality at this time.

#### 4.10.36 Will Accept Infinite COV Subscriptions

The IUT will accept subscriptions that do not include a lifetime parameter.

The BTL does not consider the use of infinite subscriptions a wise implementation choice due to the inability of the server to timeout the subscription if the client disappears. For the sake of interoperability, the BTL considers it wise for all COV server implementations to accept infinite subscriptions.

<b>135.1-2013 - 9.10.1.3 - Explicit Indefinite Lifetime COV Subscriptions</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 8.2.9 - Missing Lifetime Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

## 4.11 Data Sharing - View - A

### 4.11.1 Base Requirements

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

135.1-2013 - 8.18.3 - Reading and Presenting Properties		
	<b>Test Conditionality</b>	Must be executed. Note: if the IUT also claims support for DS-AV-A this test may be omitted.
	<b>Test Directives</b>	Repeat the test for <u>each</u> of the standard object types and associated properties specified by DS-V-A.
	<b>Testing Hints</b>	

### 4.11.2 Supports DS-RP-A

The IUT shall support DS-RP-A in order to read properties for presentation.

Verify Checklist		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support for DS-RP-A.
	<b>Testing Hints</b>	

### 4.11.3 Supports Primitive Value Objects

Presentation supports all Object Types defined at Protocol\_Revision less than or equal to itself.

## 4.12 Data Sharing - Advanced View - A

### 4.12.1 Base Requirements

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

135.1-2013 - 8.18.3 - Reading and Presenting Properties		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Repeat the test for <u>all</u> standard objects and properties, excluding the Life Safety and Access Control objects, and the Object_Identifier and Object_Type properties.
	<b>Testing Hints</b>	For properties that contain a CHOICE construct, the IUT shall be capable of reading and presenting each of the forms of the datatype as defined in the IUT's claimed protocol revision. Full accuracy presentation is not required throughout the IUT, but there should be at least one place provided by the IUT that allows the presentation of each property to be presented in such a way that the presentation requirements of DS-AV-A are met.

### 4.12.2 Supports DS-RP-A

The IUT shall support DS-RP-A in order to read properties for presentation.

Verify Checklist		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support for DS-RP-A.
	<b>Testing Hints</b>	



## 4.13 Data Sharing - Modify - A

### 4.13.1 Base Requirements

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

135.1-2013 - 8.22.4 - Accepting Input and Modifying Properties		
	<b>Test Conditionality</b>	Must be executed, unless the IUT also claims support for DS-AM-A.
	<b>Test Directives</b>	Repeat the test for <u>each</u> of the required object types listed in the table in the BIBB definition. Repeat for <u>each</u> of the required properties listed in the table in the BIBB definition, except for those properties which are commandable. Repeat the test for a variety of values that cover the range of values required by the “Minimum Writable Value Ranges” table in the DS-M-A BIBB definition.
	<b>Testing Hints</b>	
135.1-2013 - 8.22.5 - Accepting Input and Commanding/Relinquishing Properties		
	<b>Test Conditionality</b>	Must be executed, unless the IUT also claims support for DS-AM-A.
	<b>Test Directives</b>	This test should be executed at priority 8 only, i.e. $PR_1 = 8$ .
	<b>Testing Hints</b>	

### 4.13.2 Supports DS-WP-A

The IUT shall support DS-WP-A in order to update properties modified by the user.

Verify Checklist		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support for DS-WP-A.
	<b>Testing Hints</b>	

### 4.13.3 Supports Primitive Value Objects

Modification is supported for all Object Types defined at Protocol\_Revision less than or equal to itself.

## 4.14 Data Sharing - Advanced Modify - A

### 4.14.1 Base Requirements

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

135.1-2013 - 8.22.4 - Accepting Input and Modifying Properties		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Repeat the test for all standard objects and properties, excluding the Life Safety and Access Control objects, and the Object_Identifier and Object_Type properties. Also exclude any properties that are required to be read-only by the BACnet standard, and exclude properties which are commandable because those are covered by a different test. Repeat the test for a variety of values that cover the range of values required by the “Minimum Writable Value Ranges” table in the DS-M-A BIBB definition.
	<b>Testing Hints</b>	
135.1-2013 - 8.22.5 - Accepting Input and Commanding/Relinquishing Properties		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Repeat the test for all priority values, 1 through 16.
	<b>Testing Hints</b>	

### 4.14.2 Supports DS-WP-A

The IUT shall support DS-WP-A in order to update properties modified by the user.

Verify Checklist		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support for DS-WP-A.
	<b>Testing Hints</b>	

---

## 4.15 Initiates ReadRange

---

### 4.15.1 Base Requirements

Base requirements must be met by any IUT that initiates ReadRange.

BTL - 8.21.1 - Reading Values with no Specified Range, or BTL - 8.21.3 - Reading a Range of Values by Position		
	Test Conditionality	May be skipped if the IUT initiates ReadRange in another tested BIBB that uses ReadRange (eg T-ATR-A).
	Test Directives	
	Testing Hints	

## 4.16 Executes ReadRange

### 4.16.1 Base Requirements

Base requirements must be met by any IUT that executes ReadRange.

<b>BTL - 9.21.1.X1 - ReadRange Support for All List Properties</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.21.2.1 - Attempting to Read a Property That Does not Exist</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.21.2.2 - Attempting to Read a Property That is not a List or Array of Lists</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.21.2.3 - Attempting to Read a non-Array Property with an Array Index</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

## 4.17 Data Sharing - Change Of Value Unsubscribed - A

### 4.17.1 Base Requirements

There are no Base Requirements for this BIBB.

### 4.17.2 Accepts Unconfirmed COV Notifications Containing NULL Values

The IUT accepts Unconfirmed COV Notifications containing NULL values.

BTL - 9.10.3.X1 - Unsubscribed COV Notification Execution Test		
	Test Conditionality	Must be executed.
	Test Directives	Apply the test where one of the entries in the Notification is the NULL value, and at least one other entry in the Notification is a constructed value.
	Testing Hints	

### 4.17.3 Accepts Unconfirmed COV Notifications Containing BOOLEAN Values

The IUT accepts Unconfirmed COV Notifications containing BOOLEAN values.

BTL - 9.10.3.X1 - Unsubscribed COV Notification Execution Test		
	Test Conditionality	Must be executed.
	Test Directives	Apply the test where one of the entries in the Notification is a BOOLEAN value, and at least one other entry in the Notification is a constructed value.
	Testing Hints	

### 4.17.4 Accepts Unconfirmed COV Notifications Containing Enumerated Values

The IUT accepts Unconfirmed COV Notifications containing Enumerated values.

BTL - 9.10.3.X1 - Unsubscribed COV Notification Execution Test		
	Test Conditionality	Must be executed.
	Test Directives	Apply the test where one of the entries in the Notification is an Enumerated value, and at least one other entry in the Notification is a constructed value.
	Testing Hints	

### 4.17.5 Accepts Unconfirmed COV Notifications Containing INTEGER Values

The IUT accepts Unconfirmed COV Notifications containing INTEGER values.

BTL - 9.10.3.X1 - Unsubscribed COV Notification Execution Test		
	Test Conditionality	Must be executed.
	Test Directives	Apply the test where one of the entries in the Notification is an INTEGER value, and at least one other entry in the Notification is a constructed value.
	Testing Hints	

#### 4.17.6 Accepts UnconfirmedCOVNotifications Containing Unsigned Values

The IUT accepts UnconfirmedCOVNotifications containing Unsigned values.

BTL - 9.10.3.X1 - Unsubscribed COVNotification Execution Test		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Apply the test where one of the entries in the Notification is an Unsigned value, and at least one other entry in the Notification is a constructed value.
	<b>Testing Hints</b>	

#### 4.17.7 Accepts UnconfirmedCOVNotifications Containing REAL Values

The IUT accepts UnconfirmedCOVNotifications containing REAL values.

BTL - 9.10.3.X1 - Unsubscribed COVNotification Execution Test		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Apply the test where one of the entries in the Notification is a REAL value, and at least one other entry in the Notification is a constructed value.
	<b>Testing Hints</b>	

#### 4.17.8 Accepts UnconfirmedCOVNotifications Containing Double Values

The IUT accepts UnconfirmedCOVNotifications containing Double values.

BTL - 9.10.3.X1 - Unsubscribed COVNotification Execution Test		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Apply the test where one of the entries in the Notification is a Double value, and at least one other entry in the Notification is a constructed value.
	<b>Testing Hints</b>	

#### 4.17.9 Accepts UnconfirmedCOVNotifications Containing Time Values

The IUT accepts UnconfirmedCOVNotifications containing Time values.

BTL - 9.10.3.X1 - Unsubscribed COVNotification Execution Test		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Apply the test where one of the entries in the Notification is a Time value, and at least one other entry in the Notification is a constructed value.
	<b>Testing Hints</b>	

#### 4.17.10 Accepts UnconfirmedCOVNotifications Containing Date Values

The IUT accepts UnconfirmedCOVNotifications containing Date values.

BTL - 9.10.3.X1 - Unsubscribed COVNotification Execution Test		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Apply the test where one of the entries in the Notification is a Date value, and at least one other entry in the Notification is a constructed value.
	<b>Testing Hints</b>	

#### 4.17.11 Accepts UnconfirmedCOVNotifications Containing DateTime Values

The IUT accepts UnconfirmedCOVNotifications containing DateTime values.

BTL - 9.10.3.X1 - Unsubscribed COVNotification Execution Test		
	Test Conditionality	Must be executed.
	Test Directives	Apply the test where one of the entries in the Notification is a DateTime value, and at least one other entry in the Notification is a constructed value.
	Testing Hints	

#### 4.17.12 Accepts UnconfirmedCOVNotifications Containing Octet String Values

The IUT accepts UnconfirmedCOVNotifications containing Octet String values.

BTL - 9.10.3.X1 - Unsubscribed COVNotification Execution Test		
	Test Conditionality	Must be executed.
	Test Directives	Apply the test where one of the entries in the Notification is an Octet String value, and at least one other entry in the Notification is a constructed value.
	Testing Hints	

#### 4.17.13 Accepts UnconfirmedCOVNotifications Containing Bit String Values

The IUT accepts UnconfirmedCOVNotifications containing Bit String values.

BTL - 9.10.3.X1 - Unsubscribed COVNotification Execution Test		
	Test Conditionality	Must be executed.
	Test Directives	Apply the test where one of the entries in the Notification is a Bit String value, and at least one other entry in the Notification is a constructed value.
	Testing Hints	

#### 4.17.14 Accepts UnconfirmedCOVNotifications Containing Character String Values

The IUT accepts UnconfirmedCOVNotifications containing Character String values.

BTL - 9.10.3.X1 - Unsubscribed COVNotification Execution Test		
	Test Conditionality	Must be executed.
	Test Directives	Apply the test where one of the entries in the Notification is a Character String value, and at least one other entry in the Notification is a constructed value.
	Testing Hints	

#### 4.17.15 Accepts UnconfirmedCOVNotifications Containing BACnetObjectIdentifier Values

The IUT accepts UnconfirmedCOVNotifications containing BACnetObjectIdentifier values.

BTL - 9.10.3.X1 - Unsubscribed COVNotification Execution Test		
	Test Conditionality	Must be executed.

	<b>Test Directives</b>	Apply the test where one of the entries in the Notification is a BACnetObjectIdentifier value, and at least one other entry in the Notification is a constructed value At least one of the properties observed to be processed by the test shall contain a BACnetObjectIdentifier value.
	<b>Testing Hints</b>	

#### 4.17.16 Accepts UnconfirmedCOVNotifications Containing Constructed Values

The IUT accepts UnconfirmedCOVNotifications containing constructed property values, whole arrays, and lists.

<b>BTL - 9.10.3.X1 - Unsubscribed COVNotification Execution Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Apply the test where one of the entries in the Notification is a constructed value, whole array, or list.
	<b>Testing Hints</b>	



---

## 4.18 Data Sharing - Change Of Value Unsubscribed - B

---

### 4.18.1 Base Requirements

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

BTL - 8.3.X11 - Unsubscribed COV Service Initiation Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

## 4.19 Data Sharing - Change Of Value Property - A

### 4.19.1 Base Requirements

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

<b>BTL - 8.11.X1.3 - Change of Value Notification Arrives after Subscription has Expired</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 8.11.X1.2 - Change of Value Notifications with Invalid Process Identifier</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 8.11.X1.4 - Change of Value Notifications with Invalid Monitored Object Identifier</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 8.11.X1.5 - Change of Value Notifications with Invalid Monitored property</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 4.19.2 Subscribes with Lifetimes up to 8 Hours in Duration

The IUT is capable of subscribing with a lifetime less than or equal to 28800 seconds (8 hours).

<b>BTL - 8.11.X4 - Generates 8 Hour Lifetimes</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 4.19.3 Can Subscribe for Confirmed Notifications

The IUT can subscribe for, receive, and process confirmed Change of Value notifications.

<b>BTL - 8.11.1 - Confirmed Notifications Subscription</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 4.19.4 Can Subscribe for Unconfirmed Notifications

The IUT can subscribe for, receive, and process unconfirmed Change of Value notifications.

<b>BTL - 8.11.2 - Unconfirmed Notifications Subscription</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 4.19.5 Can Subscribe to Non-array Properties

Purpose: To verify that the IUT can subscribe for, receive, and process a Change of Value notification that does not contain the 'Property Array Index' parameter and can correctly process the response

<b>BTL - 8.11.X1.1 - Change of Value Notifications</b>		
--	--	--

	<b>Test Conditionality</b>	Either a confirmed or an unconfirmed COV notification may be observed.
	<b>Test Directives</b>	Execute test using 'Monitored Property Identifier' = (any valid non-array property which the vendor supports in a SubscribeCOVProperty-Request)
	<b>Testing Hints</b>	

#### 4.19.6 Can Subscribe to Array Elements

Purpose: To verify that the IUT can subscribe for, receive, and process a Change of Value notification that references a specific element of an array property and can correctly process the response

<b>BTL - 8.11.X1.1 - Change of Value Notifications</b>		
	<b>Test Conditionality</b>	Either a confirmed or an unconfirmed COV notification may be observed.
	<b>Test Directives</b>	Execute test using 'Monitored Property Identifier' = (any valid array property which the vendor supports in a SubscribeCOVProperty-Request and optional array index with value different from 0)
	<b>Testing Hints</b>	

#### 4.19.7 Can Subscribe to the Size of an Array

Purpose: To verify that the IUT can subscribe for, receive, and process a Change of Value notification that references the size of an array property and can correctly process the response

<b>BTL - 8.11.X1.1 - Change of Value Notifications</b>		
	<b>Test Conditionality</b>	Either a confirmed or an unconfirmed COV notification may be observed.
	<b>Test Directives</b>	Execute test using 'Monitored Property Identifier' = (any valid array property which the vendor supports in a SubscribeCOVProperty-Request) and optional array index with value equal to 0)
	<b>Testing Hints</b>	

#### 4.19.8 Can Subscribe to Whole Arrays

Purpose: To verify that the IUT can subscribe for, receive, and process a Change of Value notification that does not contain the 'Property Array Index' parameter for an array property and can correctly process the response.

<b>BTL - 8.11.X1.1 - Change of Value Notifications</b>		
	<b>Test Conditionality</b>	Either a confirmed or an unconfirmed COV notification may be observed.
	<b>Test Directives</b>	Execute test using 'Monitored Property Identifier' = (any valid array property which the vendor supports in a SubscribeCOVProperty-Request) object with no optional array index)
	<b>Testing Hints</b>	

#### 4.19.9 Can Subscribe to List Properties

Purpose: To verify that the IUT can subscribe for, receive, and process a Change of Value notification that references a list property and can correctly process the response

<b>BTL - 8.11.X1.1 - Change of Value Notifications</b>		
	<b>Test Conditionality</b>	Either a confirmed or an unconfirmed COV notification may be observed.

<b>Test Directives</b>	Execute test using 'Monitored Property Identifier' = (any list property which the vendor supports in a SubscribeCOVProperty-Request))
<b>Testing Hints</b>	

#### 4.19.10 Can Subscribe with a COV Increment

The IUT can subscribe with the parameter 'COV Increment' for, receive, and process Change of Value notifications

<b>BTL - 8.11.X1.1 - Change of Value Notifications</b>		
	<b>Test Conditionality</b>	Either a confirmed or an unconfirmed COV notification may be observed.
	<b>Test Directives</b>	Execute test using 'Monitored Property Identifier' = (any property which the vendor supports in a SubscribeCOVProperty-Request) and ensure that IUT generates a SubscribeCOVProperty-Request which contains 'COV Increment' parameter,
	<b>Testing Hints</b>	

#### 4.19.11 Can Subscribe to NULL Property Values

The IUT can subscribe for, receive, and process Change of Value notifications from property that contains a NULL value

<b>BTL - 8.11.X1.1 - Change of Value Notifications</b>		
	<b>Test Conditionality</b>	Either a confirmed or an unconfirmed COV notification may be observed.
	<b>Test Directives</b>	Execute test using 'Monitored Property Identifier' = (any property which the vendor supports in a SubscribeCOVProperty-Request that can contain a NULL value)
	<b>Testing Hints</b>	Schedule_Default and Present_Value of the Schedule Object, Alarm_Values and Fault_Values of the CharacterString Value Object and Low_Diff_Limit in the Loop Object are standard properties that should accept a written NULL.

#### 4.19.12 Can Subscribe to BOOLEAN Property Values

The IUT can subscribe for, receive, and process Change of Value notifications from BOOLEAN property values

<b>BTL - 8.11.X1.1 - Change of Value Notifications</b>		
	<b>Test Conditionality</b>	Either a confirmed or an unconfirmed COV notification may be observed.
	<b>Test Directives</b>	Execute test using 'Monitored Property Identifier' = (any property which the vendor supports in a SubscribeCOVProperty-Request that can contain a BOOLEAN value)
	<b>Testing Hints</b>	

#### 4.19.13 Can Subscribe to Enumerated Property Values

The IUT can subscribe for, receive, and process Change of Value notifications from Enumerated property values

<b>BTL - 8.11.X1.1 - Change of Value Notifications</b>		
--	--	--

	<b>Test Conditionality</b>	Either a confirmed or an unconfirmed COV notification may be observed.
	<b>Test Directives</b>	Execute test using 'Monitored Property Identifier' = (any property which the vendor supports in a SubscribeCOVProperty-Request that can contain an Enumerated value )
	<b>Testing Hints</b>	

#### 4.19.14 Can Subscribe to INTEGER Property Values

The IUT can subscribe for, receive, and process Change of Value notifications from INTEGER property values

<b>BTL - 8.11.X1.1 - Change of Value Notifications</b>		
	<b>Test Conditionality</b>	Either a confirmed or an unconfirmed COV notification may be observed.
	<b>Test Directives</b>	Execute test using 'Monitored Property Identifier' = (any property which the vendor supports in a SubscribeCOVProperty-Request that can contain an INTEGER value )
	<b>Testing Hints</b>	

#### 4.19.15 Can Subscribe to Unsigned Property Values

The IUT can subscribe for, receive, and process Change of Value notifications from Unsigned property values

<b>BTL - 8.11.X1.1 - Change of Value Notifications</b>		
	<b>Test Conditionality</b>	Either a confirmed or an unconfirmed COV notification may be observed.
	<b>Test Directives</b>	Execute test using 'Monitored Property Identifier' = (any property which the vendor supports in a SubscribeCOVProperty-Request that can contain an Unsigned value )
	<b>Testing Hints</b>	

#### 4.19.16 Can Subscribe to REAL Property Values

The IUT can subscribe for, receive, and process Change of Value notifications from REAL property values

<b>BTL - 8.11.X1.1 - Change of Value Notifications</b>		
	<b>Test Conditionality</b>	Either a confirmed or an unconfirmed COV notification may be observed.
	<b>Test Directives</b>	Execute test using 'Monitored Property Identifier' = (any property which the vendor supports in a SubscribeCOVProperty-Request that can contain a REAL value )
	<b>Testing Hints</b>	

#### 4.19.17 Can Subscribe to Double Property Values

The IUT can subscribe for, receive, and process Change of Value notifications from Double property values

<b>BTL - 8.11.X1.1 - Change of Value Notifications</b>		
	<b>Test Conditionality</b>	Either a confirmed or an unconfirmed COV notification may be observed.
	<b>Test Directives</b>	Execute test using 'Monitored Property Identifier' = (any property which the vendor supports in a SubscribeCOVProperty-Request that can contain a Double value )

	Testing Hints	
--	---------------	--

#### 4.19.18 Can Subscribe to Time Property Values

The IUT can subscribe for, receive, and process Change of Value notifications from Time property values

BTL - 8.11.X1.1 - Change of Value Notifications		
	Test Conditionality	Either a confirmed or an unconfirmed COV notification may be observed.
	Test Directives	Execute test using 'Monitored Property Identifier' = (any property which the vendor supports in a SubscribeCOVProperty-Request that can contain a Time value )
	Testing Hints	

#### 4.19.19 Can Subscribe to Date Property Values

The IUT can subscribe for, receive, and process Change of Value notifications from Date property values

BTL - 8.11.X1.1 - Change of Value Notifications		
	Test Conditionality	Either a confirmed or an unconfirmed COV notification may be observed.
	Test Directives	Execute test using 'Monitored Property Identifier' = (any property which the vendor supports in a SubscribeCOVProperty-Request that can contain a Date value )
	Testing Hints	

#### 4.19.20 Can Subscribe to CharacterString Property Values

The IUT can subscribe for, receive, and process Change of Value notifications from CharacterString property values

BTL - 8.11.X1.1 - Change of Value Notifications		
	Test Conditionality	Either a confirmed or an unconfirmed COV notification may be observed.
	Test Directives	Execute test using 'Monitored Property Identifier' = (any property which the vendor supports in a SubscribeCOVProperty-Request that can contain a CharacterString value )
	Testing Hints	

#### 4.19.21 Can Subscribe to OctetString Property Values

The IUT can subscribe for, receive, and process Change of Value notifications from OctetString property values

BTL - 8.11.X1.1 - Change of Value Notifications		
	Test Conditionality	Either a confirmed or an unconfirmed COV notification may be observed.
	Test Directives	Execute test using 'Monitored Property Identifier' = (any property which the vendor supports in a SubscribeCOVProperty-Request that can contain an OctetString value )
	Testing Hints	

#### 4.19.22 Can Subscribe to BitString Property Values

The IUT can subscribe for, receive, and process Change of Value notifications from BitString property values

BTL - 8.11.X1.1 - Change of Value Notifications		
	<b>Test Conditionality</b>	Either a confirmed or an unconfirmed COV notification may be observed.
	<b>Test Directives</b>	Execute test using 'Monitored Property Identifier' = (any property which the vendor supports in a SubscribeCOVProperty-Request that can contain a BitString value )
	<b>Testing Hints</b>	

#### 4.19.23 Can Subscribe to BACnetObjectIdentifier Property Values

The IUT can subscribe for, receive, and process Change of Value notifications from BACnetObjectIdentifier property values

BTL - 8.11.X1.1 - Change of Value Notifications		
	<b>Test Conditionality</b>	Either a confirmed or an unconfirmed COV notification may be observed.
	<b>Test Directives</b>	Execute test using 'Monitored Property Identifier' = (any property which the vendor supports in a SubscribeCOVProperty-Request that can contain a BACnetObjectIdentifier value )
	<b>Testing Hints</b>	

#### 4.19.24 Can Subscribe to Constructed Property Values

The IUT can subscribe for, receive, and process Change of Value notifications from constructed property values

BTL - 8.11.X1.1 - Change of Value Notifications		
	<b>Test Conditionality</b>	Either a confirmed or an unconfirmed COV notification may be observed.
	<b>Test Directives</b>	Execute test using 'Monitored Property Identifier' = (any property which the vendor supports in a SubscribeCOVProperty-Request that can contain a constructed value )
	<b>Testing Hints</b>	

#### 4.19.25 Can Subscribe to Proprietary Property Values of Basic Data Types

The IUT can subscribe for, receive, and process Change of Value notifications from proprietary property values of basic data types

BTL - 8.11.X1.1 - Change of Value Notifications		
	<b>Test Conditionality</b>	Either a confirmed or an unconfirmed COV notification may be observed.
	<b>Test Directives</b>	Execute test using 'Monitored Property Identifier' = (any property which the vendor supports in a SubscribeCOVProperty-Request that can contain a proprietary value of basic data types)
	<b>Testing Hints</b>	

#### 4.19.26 Can Cancel Subscriptions

The IUT can explicitly cancel COV subscriptions (in contrast to just letting the subscription expire).

BTL - 8.11.3 - Canceling a Subscription		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	



## 4.20 Data Sharing - Change Of Value Property - B

### 4.20.1 Base Requirements

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

<b>BTL - 9.11.1.1 - Confirmed COV Notifications</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Select parameters for an object and property which supports SubscribeCOVProperty. <del>Repeat for each different datatype for which SubscribeCOVProperty is supported..</del>
	<b>Testing Hints</b>	
<b>BTL - 9.11.1.2 - Unconfirmed COV Notifications</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Apply the test to an object and property which supports SubscribeCOVProperty. <del>Repeat for each different datatype for which SubscribeCOVProperty is supported..</del>
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.11.1.4 - Canceling COV Subscriptions</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.11.1.5 - Canceling Expired or Non-Existing Subscriptions</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.11.1.7 - Finite Lifetime Subscriptions</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.11.1.8 - Updating Existing Subscriptions</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.11.1.9 - Client-Supplied COV Increment</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.11.2.1 - The Monitored Object Does Not Support COV Notification</b>		
	<b>Test Conditionality</b>	Must be executed, unless all objects support SubscribeCOVProperty on at least one of its properties.
	<b>Test Directives</b>	Apply the test to a property in an object that does not support COV (on any property).
	<b>Testing Hints</b>	
<b>BTL - 9.11.2.2 - The Monitored Property Does Not Support COV Notification</b>		
	<b>Test Conditionality</b>	Must be executed, unless all objects support SubscribeCOVProperty on all properties.
	<b>Test Directives</b>	Apply the test to a property for which the IUT does not support COV, which is contained in an object that does support COV (on a different property).
	<b>Testing Hints</b>	
<b>BTL - 9.11.2.X11 - Monitored Object Does Not Exist</b>		
	<b>Test Conditionality</b>	Must be executed if Protocol_Revision >= 15
	<b>Test Directives</b>	

	<b>Testing Hints</b>	
<b>BTL - 9.11.2.X12 - Monitored Property Does Not Exist</b>		
	<b>Test Conditionality</b>	Must be executed if Protocol Revision $\geq 15$
	<b>Test Directives</b>	Be sure to test at least one property identifier that is within the ASHRAE allocated range for standard property identifiers, but that has not yet been defined.
	<b>Testing Hints</b>	
<b>BTL - 9.11.2.X13 - There Is No Space For Subscription</b>		
	<b>Test Conditionality</b>	Must be executed
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.11.2.X14 - The Lifetime Parameter is Out of Range</b>		
	<b>Test Conditionality</b>	Must be executed if Protocol Revision $\geq 15$
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.20.2 Supports Lifetimes up to 8 Hours in Duration

The IUT will accept COVP subscriptions with lifetimes up to 8 hours.

<b>BTL - 9.11.1.X10 - Accepts SubscribeCOVProperty-Requests with 8 Hour Lifetimes</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 4.20.3 Supports COVP for Status\_Flags Changes

The IUT supports change of value notifications for Status\_Flags changes

<b>BTL - 9.11.1.X21 - Confirmed Change of Value Notification from Status_Flags Property</b>		
	<b>Test Conditionality</b>	Must be executed if object type contains a Status_Flag and property which supports SubscribeCOVProperty.
	<b>Test Directives</b>	Repeat test for at least one object of each type that has at least one property which supports SubscribeCOVProperty
	<b>Testing Hints</b>	
<b>BTL - 9.11.1.X22 - Unconfirmed Change of Value Notification from Status_Flags Property</b>		
	<b>Test Conditionality</b>	Must be executed if object type contains a Status_Flag and property which supports SubscribeCOVProperty.
	<b>Test Directives</b>	Repeat test for at least one object of each type that has at least one property which supports SubscribeCOVProperty
	<b>Testing Hints</b>	

#### 4.20.4 Supports COVP to Non-array Properties

The IUT supports change of value notifications for at least one non-array property

<b>BTL - 9.11.1.1 - Confirmed COV Notifications for a SubscribeCOVProperty subscription</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Select parameters for an object and property which supports SubscribeCOVProperty.
	<b>Testing Hints</b>	
<b>BTL - 9.11.1.2 - Unconfirmed COV Notifications for a SubscribeCOVProperty subscription</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Select parameters for an object and property which supports SubscribeCOVProperty

	Testing Hints	
--	---------------	--

#### 4.20.5 Supports COVP to Array Elements

The IUT supports change of value notifications for at least one array element.

BTL - 9.11.1.1 - Confirmed COV Notifications for a SubscribeCOVProperty subscription		
	Test Conditionality	Must be executed.
	Test Directives	Select parameters for an object and property which supports SubscribeCOVProperty.
	Testing Hints	
BTL - 9.11.1.2 - Unconfirmed COV Notifications for a SubscribeCOVProperty subscription		
	Test Conditionality	Must be executed.
	Test Directives	Select parameters for an object and property which supports SubscribeCOVProperty
	Testing Hints	

#### 4.20.6 Supports COVP to the Size of an Array

The IUT supports change of value notifications for at least one index 0 of an array

BTL - 9.11.1.1 - Confirmed COV Notifications for a SubscribeCOVProperty subscription		
	Test Conditionality	Must be executed.
	Test Directives	Select parameters for an object and property which supports SubscribeCOVProperty
	Testing Hints	
BTL - 9.11.1.2 - Unconfirmed COV Notifications for a SubscribeCOVProperty subscription		
	Test Conditionality	Must be executed.
	Test Directives	Select parameters for an object and property which supports SubscribeCOVProperty
	Testing Hints	

#### 4.20.7 Supports COVP to Whole Arrays

The IUT supports change of value notifications for at least one whole array

BTL - 9.11.1.1 - Confirmed COV Notifications for a SubscribeCOVProperty subscription		
	Test Conditionality	Must be executed.
	Test Directives	Select parameters for an object and property which supports SubscribeCOVProperty.
	Testing Hints	
BTL - 9.11.1.2 - Unconfirmed COV Notifications for a SubscribeCOVProperty subscription		
	Test Conditionality	Must be executed.
	Test Directives	Select parameters for an object and property which supports SubscribeCOVProperty
	Testing Hints	

#### 4.20.8 Supports COVP to a List Property

The IUT supports change of value notifications for at least one list property

BTL - 9.11.1.1 - Confirmed COV Notifications for a SubscribeCOVProperty subscription		
	Test Conditionality	Must be executed.
	Test Directives	Select parameters for an object and property which supports SubscribeCOVProperty.
	Testing Hints	

<b>BTL - 9.11.1.2 - Unconfirmed COV Notifications for a SubscribeCOVProperty subscription</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Select parameters for an object and property which supports SubscribeCOVProperty
	<b>Testing Hints</b>	

#### 4.20.9 Supports COVP to NULL Property Values

The IUT supports change of value notifications for at least one property that contains a NULL value.

<b>BTL - 9.11.1.X11 - Confirmed Change of Value Notification from Property Value</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Apply to at least 1 property of the specified datatype. Ensure that after all applications of this test (regardless of the property datatype it is applied for), that the test has been applied at least once to each object type which supports COVP on one or more of its properties.
	<b>Testing Hints</b>	Schedule_Default and Present_Value of the Schedule Object, Alarm_Values and Fault_Values of the CharacterString Value Object and Low_Diff_Limit in the Loop Object are standard properties that can contain or accept a written NULL.
<b>BTL - 9.11.1.X12 - Unconfirmed Change of Value Notification from Property Value</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Apply to at least 1 property of the specified datatype. Ensure that after all applications of this test (regardless of the property datatype it is applied for), that the test has been applied at least once to each object type which supports COVP on one or more of its properties.
	<b>Testing Hints</b>	Schedule_Default and Present_Value of the Schedule Object, Alarm_Values and Fault_Values of the CharacterString Value Object and Low_Diff_Limit in the Loop Object are standard properties that can contain or accept a written NULL.

#### 4.20.10 Supports COVP to BOOLEAN Property Values

The IUT supports change of value notifications for at least one BOOLEAN property value.

<b>BTL - 9.11.1.X11 - Confirmed Change of Value Notification from Property Value</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Apply to at least 1 property of the specified datatype. Ensure that after all applications of this test (regardless of the property datatype it is applied for), that the test has been applied at least once to each object type which supports COVP on one or more of its properties.
	<b>Testing Hints</b>	
<b>BTL - 9.11.1.X12 - Unconfirmed Change of Value Notification from Property Value</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Apply to at least 1 property of the specified datatype. Ensure that after all applications of this test (regardless of the property datatype it is applied for), that the test has been applied at least once to each object type which supports COVP on one or more of its properties.
	<b>Testing Hints</b>	

### 4.20.11 Supports COVP to Enumerated Property Values

The IUT supports change of value notifications for at least one Enumerated property value.

<b>BTL - 9.11.1.X11 - Confirmed Change of Value Notification from Property Value</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Apply to at least 1 property of the specified datatype. Ensure that after all applications of this test (regardless of the property datatype it is applied for), that the test has been applied at least once to each object type which supports COVP on one or more of its properties.
	<b>Testing Hints</b>	
<b>BTL - 9.11.1.X12 - Unconfirmed Change of Value Notification from Property Value</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Apply to at least 1 property of the specified datatype. Ensure that after all applications of this test (regardless of the property datatype it is applied for), that the test has been applied at least once to each object type which supports COVP on one or more of its properties.
	<b>Testing Hints</b>	

### 4.20.12 Supports COVP to INTEGER Property Values

The IUT supports change of value notifications for at least one INTEGER property value.

<b>BTL - 9.11.1.X11 - Confirmed Change of Value Notification from Property Value</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Apply to at least 1 property of the specified datatype. Ensure that after all applications of this test (regardless of the property datatype it is applied for), that the test has been applied at least once to each object type which supports COVP on one or more of its properties.
	<b>Testing Hints</b>	
<b>BTL - 9.11.1.X12 - Unconfirmed Change of Value Notification from Property Value</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Apply to at least 1 property of the specified datatype. Ensure that after all applications of this test (regardless of the property datatype it is applied for), that the test has been applied at least once to each object type which supports COVP on one or more of its properties.
	<b>Testing Hints</b>	

### 4.20.13 Supports COVP to Unsigned Property Values

The IUT supports change of value notifications for at least one Unsigned Property value.

<b>BTL - 9.11.1.X11 - Confirmed Change of Value Notification from Property Value</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Apply to at least 1 property of the specified datatype. Ensure that after all applications of this test (regardless of the property datatype it is applied for), that the test has been applied at least once to each object type which supports COVP on one or more of its properties.
	<b>Testing Hints</b>	
<b>BTL - 9.11.1.X12 - Unconfirmed Change of Value Notification from Property Value</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Apply to at least 1 property of the specified datatype.

		Ensure that after all applications of this test (regardless of the property datatype it is applied for), that the test has been applied at least once to each object type which supports COVP on one or more of its properties.
	<b>Testing Hints</b>	

#### 4.20.14 Supports COVP to REAL Property Values

The IUT supports change of value notifications for at least one REAL property value.

<b>BTL - 9.11.1.X11 - Confirmed Change of Value Notification from Property Value</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Apply to at least 1 property of the specified datatype. Ensure that after all applications of this test (regardless of the property datatype it is applied for), that the test has been applied at least once to each object type which supports COVP on one or more of its properties.
	<b>Testing Hints</b>	
<b>BTL - 9.11.1.X12 - Unconfirmed Change of Value Notification from Property Value</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Apply to at least 1 property of the specified datatype. Ensure that after all applications of this test (regardless of the property datatype it is applied for), that the test has been applied at least once to each object type which supports COVP on one or more of its properties.
	<b>Testing Hints</b>	

#### 4.20.15 Supports COVP to Double Property Values

The IUT supports change of value notifications for at least one Double property value.

<b>BTL - 9.11.1.X11 - Confirmed Change of Value Notification from Property Value</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Apply to at least 1 property of the specified datatype. Ensure that after all applications of this test (regardless of the property datatype it is applied for), that the test has been applied at least once to each object type which supports COVP on one or more of its properties.
	<b>Testing Hints</b>	
<b>BTL - 9.11.1.X12 - Unconfirmed Change of Value Notification from Property Value</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Apply to at least 1 property of the specified datatype. Ensure that after all applications of this test (regardless of the property datatype it is applied for), that the test has been applied at least once to each object type which supports COVP on one or more of its properties.
	<b>Testing Hints</b>	

#### 4.20.16 Supports COVP to Time Property Values

The IUT supports change of value notifications for at least one Time property value.

<b>BTL - 9.11.1.X11 - Confirmed Change of Value Notification from Property Value</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Apply to at least 1 property of the specified datatype.

		Ensure that after all applications of this test (regardless of the property datatype it is applied for), that the test has been applied at least once to each object type which supports COVP on one or more of its properties.
	<b>Testing Hints</b>	
<b>BTL - 9.11.1.X12 - Unconfirmed Change of Value Notification from Property Value</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Apply to at least 1 property of the specified datatype. Ensure that after all applications of this test (regardless of the property datatype it is applied for), that the test has been applied at least once to each object type which supports COVP on one or more of its properties.
	<b>Testing Hints</b>	

#### 4.20.17 Supports COVP to Date Property Values

The IUT supports change of value notifications for at least one Date property value.

<b>BTL - 9.11.1.X11 - Confirmed Change of Value Notification from Property Value</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Apply to at least 1 property of the specified datatype. Ensure that after all applications of this test (regardless of the property datatype it is applied for), that the test has been applied at least once to each object type which supports COVP on one or more of its properties.
	<b>Testing Hints</b>	
<b>BTL - 9.11.1.X12 - Unconfirmed Change of Value Notification from Property Value</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Apply to at least 1 property of the specified datatype. Ensure that after all applications of this test (regardless of the property datatype it is applied for), that the test has been applied at least once to each object type which supports COVP on one or more of its properties.
	<b>Testing Hints</b>	

#### 4.20.18 Supports COVP to CharacterString Property Values

The IUT supports change of value notifications for at least one CharacterString property value.

<b>BTL - 9.11.1.X11 - Confirmed Change of Value Notification from Property Value</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Apply to at least 1 property of the specified datatype. Ensure that after all applications of this test (regardless of the property datatype it is applied for), that the test has been applied at least once to each object type which supports COVP on one or more of its properties.
	<b>Testing Hints</b>	
<b>BTL - 9.11.1.X12 - Unconfirmed Change of Value Notification from Property Value</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Apply to at least 1 property of the specified datatype. Ensure that after all applications of this test (regardless of the property datatype it is applied for), that the test has been applied at least once to each object type which supports COVP on one or more of its properties.
	<b>Testing Hints</b>	

#### 4.20.19 Supports COVP to OctetString Property Values

The IUT supports change of value notifications for at least one property with value of type OctetString.

BTL - 9.11.1.X11 - Confirmed Change of Value Notification from Property Value		
	Test Conditionality	Must be executed.
	Test Directives	Apply to at least 1 property of the specified datatype. Ensure that after all applications of this test (regardless of the property datatype it is applied for), that the test has been applied at least once to each object type which supports COVP on one or more of its properties.
	Testing Hints	
BTL - 9.11.1.X12 - Unconfirmed Change of Value Notification from Property Value		
	Test Conditionality	Must be executed.
	Test Directives	Apply to at least 1 property of the specified datatype. Ensure that after all applications of this test (regardless of the property datatype it is applied for), that the test has been applied at least once to each object type which supports COVP on one or more of its properties.
	Testing Hints	

#### 4.20.20 Supports COVP to BitString Property Values

The IUT supports change of value notifications for at least one property with value of type BitString.

BTL - 9.11.1.X11 - Confirmed Change of Value Notification from Property Value		
	Test Conditionality	Must be executed.
	Test Directives	Apply to at least 1 property of the specified datatype. Ensure that after all applications of this test (regardless of the property datatype it is applied for), that the test has been applied at least once to each object type which supports COVP on one or more of its properties.
	Testing Hints	
BTL - 9.11.1.X12 - Unconfirmed Change of Value Notification from Property Value		
	Test Conditionality	Must be executed.
	Test Directives	Apply to at least 1 property of the specified datatype. Ensure that after all applications of this test (regardless of the property datatype it is applied for), that the test has been applied at least once to each object type which supports COVP on one or more of its properties.
	Testing Hints	

#### 4.20.21 Supports COVP to BACnetObjectIdentifier Property Values

The IUT supports change of value notifications for at least one property with value of type BACnetObjectIdentifier.

BTL - 9.11.1.X11 - Confirmed Change of Value Notification from Property Value		
	Test Conditionality	Must be executed.
	Test Directives	Apply to at least 1 property of the specified datatype. Ensure that after all applications of this test (regardless of the property datatype it is applied for), that the test has been applied at least once to each object type which supports COVP on one or more of its properties.
	Testing Hints	



<b>BTL - 9.11.1.X12 - Unconfirmed Change of Value Notification from Property Value</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Apply to at least 1 property of the specified datatype. Ensure that after all applications of this test (regardless of the property datatype it is applied for), that the test has been applied at least once to each object type which supports COVP on one or more of its properties.
	<b>Testing Hints</b>	

#### 4.20.22 Supports COVP to Constructed Property Values

The IUT supports change of value notifications for at least one constructed property value.

<b>BTL - 9.11.1.X11 - Confirmed Change of Value Notification from Property Value</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Apply to at least 1 property of the specified datatype. Ensure that after all applications of this test (regardless of the property datatype it is applied for), that the test has been applied at least once to each object type which supports COVP on one or more of its properties.
	<b>Testing Hints</b>	
<b>BTL - 9.11.1.X12 - Unconfirmed Change of Value Notification from Property Value</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Apply to at least 1 property of the specified datatype. Ensure that after all applications of this test (regardless of the property datatype it is applied for), that the test has been applied at least once to each object type which supports COVP on one or more of its properties.
	<b>Testing Hints</b>	

---

## **4.21 Data Sharing - WriteGroup - A**

---

### **4.21.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

## 4.22 Data Sharing - WriteGroup - I - B

### 4.22.1 Base Requirements

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

There are no base requirements tests for this section.

### 4.22.2 Supports the Channel Object that can Propagate Values to Local Objects

The B device shall execute the WriteGroup service when appropriate using given primitive values. This will be confirmed using Channel objects.

BTL - 9.X40.1.X1 - Channel and Group Number Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
BTL - 9.X40.1.X2 - Write Priority and Overriding Priority Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
BTL - 9.X40.1.X3 - Relinquish Control Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 4.22.3 Supports the Channel Object with a Writable Allow\_Group\_Delay\_Inhibit Property

The B device shall inhibit any execution delay when appropriate.

BTL - 9.X40.1.X4 - Inhibit Delay Test with WriteGroup		
	Test Conditionality	This test shall be skipped if the Channel object does not support at least 2 entries in the List_Of_Object_Property_References.
	Test Directives	
	Testing Hints	

## 4.23 Data Sharing - WriteGroup - E - B

### 4.23.1 Base Requirements

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

There are no base requirements tests for this section.

### 4.23.2 Supports DS-WG-I-B

The IUT shall support DS-WG-I-B in order to claim DS-WG-E-B

Verify Checklist		
	Test Conditionality	Must be executed.
	Test Directives	Verify that the IUT claims support for DS-WG-I-B
	Testing Hints	

### 4.23.3 Supports DS-WP-A

The IUT shall support DS-WP-A in order to claim DS-WG-E-B

Verify Checklist		
	Test Conditionality	Must be executed.
	Test Directives	Verify that the IUT claims support for DS-WP-A
	Testing Hints	

### 4.23.4 Supports the Channel Object that can propagate values to objects on remote devices

The B device shall execute the WriteGroup service when appropriate, and the Channel object can propagate values to remote devices by using either WriteProperty or WritePropertyMultiple request.

BTL - 7.3.2.X40.4 - Propagation Entirety Test		
	Test Conditionality	Must be executed.
	Test Directives	Write to the Channel Object by a WriteGroup. Use external object references for List Of Object Property References.
	Testing Hints	

---

## **4.24 Data Sharing - Value Source Information - B**

---

### **4.24.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

---

## **4.25 Data Sharing - Change Of Value Multiple - A**

---

### **4.25.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

---

## **4.26 Data Sharing - Change Of Value Multiple - B**

---

### **4.26.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

---

## **4.27 Data Sharing - Life Safety View - A**

---

### **4.27.1 Base Requirements**

Contact BTL for interim tests for this BIBB.



---

## **4.28 Data Sharing - Life Safety Advanced View - A**

---

### **4.28.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

---

## **4.29 Data Sharing - Life Safety Modify - A**

---

### **4.29.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

---

## **4.30 Data Sharing - Life Safety Advanced Modify - A**

---

### **4.30.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

---

## **4.31 Data Sharing - Access Control View - A**

---

### **4.31.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

---

## **4.32 Data Sharing - Access Control Advanced View - A**

---

### **4.32.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

---

## **4.33 Data Sharing - Access Control Modify - A**

---

### **4.33.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

---

## **4.34 Data Sharing - Access Control Advanced Modify - A**

---

### **4.34.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

---

## **4.35 Data Sharing - Access Control User Configuration - A**

---

### **4.35.1 Base Requirements**

Contact BTL for interim tests for this BIBB.



---

## **4.36 Data Sharing - Access Control User Configuration - B**

---

### **4.36.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

---

## **4.37 Data Sharing - Access Control Site Configuration - A**

---

### **4.37.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

---

## **4.38 Data Sharing - Access Control Site Configuration - B**

---

### **4.38.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

---

## **4.39 Data Sharing - Access Control Access Door - A**

---

### **4.39.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

---

## **4.40 Data Sharing - Access Control Access Door - B**

---

### **4.40.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

---

## **4.41 Data Sharing - Access Control Credential Data Input - A**

---

### **4.41.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

---

## **4.42 Data Sharing - Access Control Credential Data Input - B**

---

### **4.42.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

---

## **4.43 Data Sharing - Lighting Output - A**

---

### **4.43.1 Base Requirements**

Contact BTL for interim tests for this BIBB.



---

## **4.44 Data Sharing - Lighting Output Status - A**

---

### **4.44.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

---

## **4.45 Data Sharing - Advanced Lighting Output - A**

---

### **4.45.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

---

## **4.46 Data Sharing - Lighting Output - B**

---

### **4.46.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

---

## **4.47 Data Sharing - Binary Lighting Output - B**

---

### **4.47.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

---

## **4.48 Data Sharing - Lighting Output Management - A**

---

### **4.48.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

---

## **4.49 Data Sharing - Lighting View - A**

---

### **4.49.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

---

## **4.50 Data Sharing - Lighting Advanced View - A**

---

### **4.50.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

---

## **4.51 Data Sharing - Lighting Modify - A**

---

### **4.51.1Base Requirements**

Contact BTL for interim tests for this BIBB.



---

## **4.52 Data Sharing - Lighting Advanced Modify - A**

---

### **4.52.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

---

## **4.53 Data Sharing - Elevator View - A**

---

### **4.53.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

---

## **4.54 Data Sharing - Elevator Advanced View - A**

---

### **4.54.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

---

## **4.55 Data Sharing - Elevator Modify - A**

---

### **4.55.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

---

## **4.56 Data Sharing - Elevator Advanced Modify - A**

---

### **4.56.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

---

## **5 Alarm and Event Management BIBBs**

---

## 5.1 Alarm and Event Management - Notification - A

### 5.1.1 Base Requirements

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

Note that the BTL requires that the IUT support execution of confirmed and unconfirmed notifications for all of the standard BACnet event algorithms generated both intrinsically and algorithmically, excluding BUFFER\_READY, ACCESS\_EVENT and CHANGE\_OF\_LIFE\_SAFETY in order to claim conformance to this BIBB.

<b>BTL - 9.4.X1 - Unsupported Message Text Character Set ConfirmedEventNotification Test</b>		
	<b>Test Conditionality</b>	If the IUT supports all character sets, this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.5.X1 - Unsupported Message Text Character Set UnconfirmedEventNotification Test</b>		
	<b>Test Conditionality</b>	If the IUT supports all character sets, this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 5.1.2 Executes ConfirmedEventNotifications

The IUT is capable of executing ConfirmedEventNotifications. This functionality will be covered by the testing of the individual algorithms.

<b>No Specific Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT's EPICS claims that it supports the ConfirmedEventNotification service.
	<b>Testing Hints</b>	

### 5.1.3 Executes UnconfirmedEventNotifications

The IUT is capable of executing UnconfirmedEventNotifications. There are currently no tests defined for this functional item.

<b>No Specific Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT's EPICS claims that it supports the UnconfirmedEventNotification service.
	<b>Testing Hints</b>	

### 5.1.4 Processes Intrinsically Generated Notifications

The IUT is capable of executing ConfirmedEventNotifications that reference an object type other than Event Enrollment.

<b>135.1-2013 - 9.4.1 - ConfirmedEventNotification Using the Time Form of the 'Timestamp' Parameter and Conveying a Text Message, 135.1-2013 - 9.4.2 - ConfirmedEventNotification Using the DateTime Form of the 'Timestamp' Parameter and no Text Message, or 135.1-2013 - 9.4.3 - ConfirmedEventNotification Using the Sequence Number Form of the 'Timestamp' Parameter and no Text Message</b>		
	<b>Test Conditionality</b>	At least one of the tests must be executed with the Event Object Identifier referencing a BACnet object other than an Event Enrollment object.
	<b>Test Directives</b>	

	Testing Hints	
--	---------------	--

### 5.1.5 Processes Algorithmically Generated Notifications

The IUT is capable of executing ConfirmedEventNotifications that reference an Event Enrollment object.

<b>135.1-2013 - 9.4.1 - ConfirmedEventNotification Using the Time Form of the ‘Timestamp’ Parameter and Conveying a Text Message,</b> <b>135.1-2013 - 9.4.2 - ConfirmedEventNotification Using the DateTime Form of the ‘Timestamp’ Parameter and no Text Message, or</b> <b>135.1-2013 - 9.4.3 - ConfirmedEventNotification Using the Sequence Number Form of the ‘Timestamp’ Parameter and no Text Message</b>		
	<b>Test Conditionality</b>	At least one of the tests must be executed with the Event Object Identifier referencing an Event Enrollment object.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 5.1.6 Processes CHANGE\_OF\_BITSTRING Notifications

The IUT is capable of executing event notifications that convey a CHANGE\_OF\_BITSTRING event transition.

<b>135.1-2013 - 9.4.1 - ConfirmedEventNotification Using the Time Form of the ‘Timestamp’ Parameter and Conveying a Text Message,</b> <b>135.1-2013 - 9.4.2 - ConfirmedEventNotification Using the DateTime Form of the ‘Timestamp’ Parameter and no Text Message, or</b> <b>135.1-2013 - 9.4.3 - ConfirmedEventNotification Using the Sequence Number Form of the ‘Timestamp’ Parameter and no Text Message</b>		
	<b>Test Conditionality</b>	At least one of the tests must be executed with the Event Type set to CHANGE_OF_BITSTRING.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	This test should be repeated for To-Normal, To-OffNormal and To-Fault transitions.

### 5.1.7 Processes CHANGE\_OF\_STATE Notifications

The IUT is capable of executing event notifications that convey a CHANGE\_OF\_STATE event transition.

<b>135.1-2013 - 9.4.1 - ConfirmedEventNotification Using the Time Form of the ‘Timestamp’ Parameter and Conveying a Text Message,</b> <b>135.1-2013 - 9.4.2 - ConfirmedEventNotification Using the DateTime Form of the ‘Timestamp’ Parameter and no Text Message, or</b> <b>135.1-2013 - 9.4.3 - ConfirmedEventNotification Using the Sequence Number Form of the ‘Timestamp’ Parameter and no Text Message</b>		
	<b>Test Conditionality</b>	At least one of the tests must be executed with the Event Type set to CHANGE_OF_STATE.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	This test should be repeated for To-Normal, To-OffNormal and To-Fault transitions.

### 5.1.8 Processes CHANGE\_OF\_VALUE Notifications

The IUT is capable of executing event notifications that convey a CHANGE\_OF\_VALUE event transition.



<b>135.1-2013 - 9.4.1 - ConfirmedEventNotification Using the Time Form of the ‘Timestamp’ Parameter and Conveying a Text Message,</b> <b>135.1-2013 - 9.4.2 - ConfirmedEventNotification Using the DateTime Form of the ‘Timestamp’ Parameter and no Text Message, or</b> <b>135.1-2013 - 9.4.3 - ConfirmedEventNotification Using the Sequence Number Form of the ‘Timestamp’ Parameter and no Text Message</b>		
	<b>Test Conditionality</b>	At least one of the tests must be executed with the Event Type set to CHANGE_OF_VALUE.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	This test should be repeated for both To-Normal and To-Fault transitions.

### 5.1.9 Processes COMMAND\_FAILURE Notifications

The IUT is capable of executing event notifications that convey a COMMAND\_FAILURE event transition.

<b>135.1-2013 - 9.4.1 - ConfirmedEventNotification Using the Time Form of the ‘Timestamp’ Parameter and Conveying a Text Message,</b> <b>135.1-2013 - 9.4.2 - ConfirmedEventNotification Using the DateTime Form of the ‘Timestamp’ Parameter and no Text Message, or</b> <b>135.1-2013 - 9.4.3 - ConfirmedEventNotification Using the Sequence Number Form of the ‘Timestamp’ Parameter and no Text Message</b>		
	<b>Test Conditionality</b>	At least one of the tests must be executed with the Event Type set to COMMAND_FAILURE.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	This test should be repeated for To-Normal, To-OffNormal and To-Fault transitions.

### 5.1.10Processes FLOATING\_LIMIT Notifications

The IUT is capable of executing event notifications that convey a FLOATING\_LIMIT event transition.

<b>135.1-2013 - 9.4.1 - ConfirmedEventNotification Using the Time Form of the ‘Timestamp’ Parameter and Conveying a Text Message,</b> <b>135.1-2013 - 9.4.2 - ConfirmedEventNotification Using the DateTime Form of the ‘Timestamp’ Parameter and no Text Message, or</b> <b>135.1-2013 - 9.4.3 - ConfirmedEventNotification Using the Sequence Number Form of the ‘Timestamp’ Parameter and no Text Message</b>		
	<b>Test Conditionality</b>	At least one of the tests must be executed with the Event Type set to FLOATING_LIMIT.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	This test should be repeated for To-Normal, To-OffNormal and To-Fault transitions. This test should be repeated for High-Limit and Low-Limit transitions.

### 5.1.11Processes OUT\_OF\_RANGE Notifications

The IUT is capable of executing event notifications that convey an OUT\_OF\_RANGE event transition.

<b>135.1-2013 - 9.4.1 - ConfirmedEventNotification Using the Time Form of the ‘Timestamp’ Parameter and Conveying a Text Message,</b> <b>135.1-2013 - 9.4.2 - ConfirmedEventNotification Using the DateTime Form of the ‘Timestamp’ Parameter and no Text Message, or</b> <b>135.1-2013 - 9.4.3 - ConfirmedEventNotification Using the Sequence Number Form of the ‘Timestamp’ Parameter and no Text Message</b>		
	<b>Test Conditionality</b>	At least one of the tests must be executed with the Event Type set to OUT_OF_RANGE.

	<b>Test Directives</b>	
	<b>Testing Hints</b>	This test should be repeated for To-Normal, To-OffNormal and To-Fault transitions. This test should be repeated for High-Limit and Low-Limit transitions.

### 5.1.12 Processes UNSIGNED\_RANGE Notifications

The IUT is capable of executing event notifications that convey a UNSIGNED\_RANGE event transition.

<b>135.1-2013 - 9.4.1 - ConfirmedEventNotification Using the Time Form of the ‘Timestamp’ Parameter and Conveying a Text Message,</b> <b>135.1-2013 - 9.4.2 - ConfirmedEventNotification Using the DateTime Form of the ‘Timestamp’ Parameter and no Text Message, or</b> <b>135.1-2013 - 9.4.3 - ConfirmedEventNotification Using the Sequence Number Form of the ‘Timestamp’ Parameter and no Text Message</b>		
	<b>Test Conditionality</b>	At least one of the tests must be executed with the Event Type set to UNSIGNED_RANGE.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	This test should be repeated for To-Normal, To-OffNormal and To-Fault transitions. This test should be repeated for High-Limit and Low-Limit transitions.

### 5.1.13 Processes Notifications that convey a proprietary Event Type

The IUT is capable of executing event notifications that convey a proprietary Event Type.

<b>135.1-2013 - 9.4.1 - ConfirmedEventNotification Using the Time Form of the ‘Timestamp’ Parameter and Conveying a Text Message,</b> <b>135.1-2013 - 9.4.2 - ConfirmedEventNotification Using the DateTime Form of the ‘Timestamp’ Parameter and no Text Message, or</b> <b>135.1-2013 - 9.4.3 - ConfirmedEventNotification Using the Sequence Number Form of the ‘Timestamp’ Parameter and no Text Message</b>		
	<b>Test Conditionality</b>	At least one of the tests must be executed with the Event Type set to a proprietary value.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	This test should be repeated for To-Normal, To-OffNormal and To-Fault transitions. This test should be repeated for each of the standard off-normal event state values, and at least 1 proprietary event state value. The test should be repeated with the list of BACnetPropertyValue empty and non-empty.

### 5.1.14 Processes Event Notifications with Timestamps of the BACnetDateTime Form

The IUT is capable of executing ConfirmedEventNotifications that contain a timestamp of the BACnetDateTime form.

<b>135.1-2013 - 9.4.2 - ConfirmedEventNotification Using the DateTime Form of the ‘Timestamp’ Parameter and no Text Message</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 5.1.15 Processes Event Notifications with Timestamps of the Time Form

The IUT is capable of executing ConfirmedEventNotifications that contain a timestamp of the Time form.

<b>135.1-2013 - 9.4.1 - ConfirmedEventNotification Using the Time Form of the ‘Timestamp’ Parameter and Conveying a Text Message</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### **5.1.16Processes Event Notifications with Timestamps of the Sequence Number Form**

The IUT is capable of executing ConfirmedEventNotifications that contain a timestamp of the Sequence Number form.

<b>135.1-2013 - 9.4.3 - ConfirmedEventNotification Using the Sequence Number Form of the ‘Timestamp’ Parameter and no Text Message</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### **5.1.17Processes EXTENDED Notifications that convey a proprietary set of Event Values**

The IUT is capable of executing event notifications that use the EXTENDED event notification CHOICE. EXTENDED Notifications convey an Alert or proprietary set of Event Values.

<b>135.1-2013 - 9.4.1 - ConfirmedEventNotification Using the Time Form of the ‘Timestamp’ Parameter and Conveying a Text Message, 135.1-2013 - 9.4.2 - ConfirmedEventNotification Using the DateTime Form of the ‘Timestamp’ Parameter and no Text Message, or 135.1-2013 - 9.4.3 - ConfirmedEventNotification Using the Sequence Number Form of the ‘Timestamp’ Parameter and no Text Message</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	At least one of the tests must be executed with the Event Type set to EXTENDED.
	<b>Testing Hints</b>	

### **5.1.18Processes DOUBLE\_OUT\_OF\_RANGE Notifications**

The IUT is capable of executing event notifications that convey a DOUBLE\_OUT\_OF\_RANGE event transition.

<b>135.1-2013 - 9.4.1 - ConfirmedEventNotification Using the Time Form of the ‘Timestamp’ Parameter and Conveying a Text Message, 135.1-2013 - 9.4.2 - ConfirmedEventNotification Using the DateTime Form of the ‘Timestamp’ Parameter and no Text Message, or 135.1-2013 - 9.4.3 - ConfirmedEventNotification Using the Sequence Number Form of the ‘Timestamp’ Parameter and no Text Message</b>		
	<b>Test Conditionality</b>	Must be executed if the device claims conformance to protocol revision 10 or higher.
	<b>Test Directives</b>	At least one of the tests must be executed with the Event Type set to DOUBLE_OUT_OF_RANGE. The test should be repeated for TO NORMAL and TO OFFNORMAL transitions.
	<b>Testing Hints</b>	

### **5.1.19Processes SIGNED\_OUT\_OF\_RANGE Notifications**

The IUT is capable of executing event notifications that convey a SIGNED\_OUT\_OF\_RANGE event transition.

<b>135.1-2013 - 9.4.1 - ConfirmedEventNotification Using the Time Form of the ‘Timestamp’ Parameter and Conveying a Text Message,</b> <b>135.1-2013 - 9.4.2 - ConfirmedEventNotification Using the DateTime Form of the ‘Timestamp’ Parameter and no Text Message, or</b> <b>135.1-2013 - 9.4.3 - ConfirmedEventNotification Using the Sequence Number Form of the ‘Timestamp’ Parameter and no Text Message</b>		
	<b>Test Conditionality</b>	Must be executed if the device claims conformance to protocol revision 10 or higher.
	<b>Test Directives</b>	At least one of the tests must be executed with the Event Type set to SIGNED_OUT_OF_RANGE. The test should be repeated for TO_NORMAL and TO_OFFNORMAL transitions.
	<b>Testing Hints</b>	

### 5.1.20Processes UNSIGNED\_OUT\_OF\_RANGE Notifications

The IUT is capable of executing event notifications that convey a UNSIGNED\_OUT\_OF\_RANGE event transition.

<b>135.1-2013 - 9.4.1 - ConfirmedEventNotification Using the Time Form of the ‘Timestamp’ Parameter and Conveying a Text Message,</b> <b>135.1-2013 - 9.4.2 - ConfirmedEventNotification Using the DateTime Form of the ‘Timestamp’ Parameter and no Text Message, or</b> <b>135.1-2013 - 9.4.3 - ConfirmedEventNotification Using the Sequence Number Form of the ‘Timestamp’ Parameter and no Text Message</b>		
	<b>Test Conditionality</b>	Must be executed if the device claims conformance to protocol revision 10 or higher.
	<b>Test Directives</b>	At least one of the tests must be executed with the Event Type set to UNSIGNED_OUT_OF_RANGE. This test should be repeated for TO_NORMAL and TO_OFFNORMAL transitions.
	<b>Testing Hints</b>	

### 5.1.21Processes CHANGE\_OF\_CHARACTERSTRING Notifications

The IUT is capable of executing event notifications that convey a CHANGE\_OF\_CHARACTERSTRING event transition.

<b>135.1-2013 - 9.4.1 - ConfirmedEventNotification Using the Time Form of the ‘Timestamp’ Parameter and Conveying a Text Message,</b> <b>135.1-2013 - 9.4.2 - ConfirmedEventNotification Using the DateTime Form of the ‘Timestamp’ Parameter and no Text Message, or</b> <b>135.1-2013 - 9.4.3 - ConfirmedEventNotification Using the Sequence Number Form of the ‘Timestamp’ Parameter and no Text Message</b>		
	<b>Test Conditionality</b>	Must be executed if the device claims conformance to protocol revision 10 or higher.
	<b>Test Directives</b>	At least one of the tests must be executed with the Event Type set to CHANGE_OF_CHARACTERSTRING. The test should be repeated for TO_NORMAL and TO_OFFNORMAL transitions.
	<b>Testing Hints</b>	

### 5.1.22Processes CHANGE\_OF\_STATUS\_FLAGS Notifications

The IUT is capable of executing event notifications that convey a CHANGE\_OF\_STATUS\_FLAGS event transition.

<b>135.1-2013 - 9.4.1 - ConfirmedEventNotification Using the Time Form of the ‘Timestamp’ Parameter and Conveying a Text Message,</b> <b>135.1-2013 - 9.4.2 - ConfirmedEventNotification Using the DateTime Form of the ‘Timestamp’ Parameter and no Text Message, or</b> <b>135.1-2013 - 9.4.3 - ConfirmedEventNotification Using the Sequence Number Form of the ‘Timestamp’ Parameter and no Text Message</b>		
	<b>Test Conditionality</b>	Must be executed if the device claims conformance to protocol revision 11 or higher.
	<b>Test Directives</b>	At least one of the tests must be executed with the Event Type set to CHANGE OF STATUS FLAGS.
	<b>Testing Hints</b>	

### 5.1.23Processes CHANGE\_OF\_RELIABILITY Notifications

The IUT is capable of executing event notifications that convey a CHANGE\_OF\_RELIABILITY event transition.

<b>135.1-2013 - 9.4.1 - ConfirmedEventNotification Using the Time Form of the ‘Timestamp’ Parameter and Conveying a Text Message,</b> <b>135.1-2013 - 9.4.2 - ConfirmedEventNotification Using the DateTime Form of the ‘Timestamp’ Parameter and no Text Message, or</b> <b>135.1-2013 - 9.4.3 - ConfirmedEventNotification Using the Sequence Number Form of the ‘Timestamp’ Parameter and no Text Message</b>		
	<b>Test Conditionality</b>	Must be executed if the device claims conformance to protocol revision 13 or higher.
	<b>Test Directives</b>	At least one of the tests must be executed with the Event Type set to CHANGE OF RELIABILITY.
	<b>Testing Hints</b>	

### 5.1.24Processes CHANGE\_OF\_DISCRETE\_VALUE Notifications

The IUT is capable of executing event notifications that convey a CHANGE\_OF\_DISCRETE\_VALUE event transition.

<b>135.1-2013 - 9.4.1 - ConfirmedEventNotification Using the Time Form of the ‘Timestamp’ Parameter and Conveying a Text Message,</b> <b>135.1-2013 - 9.4.2 - ConfirmedEventNotification Using the DateTime Form of the ‘Timestamp’ Parameter and no Text Message, or</b> <b>135.1-2013 - 9.4.3 - ConfirmedEventNotification Using the Sequence Number Form of the ‘Timestamp’ Parameter and no Text Message</b>		
	<b>Test Conditionality</b>	Must be executed if the device claims conformance to protocol revision 16 or higher.
	<b>Test Directives</b>	At least one of the tests must be executed with the Event Type set to CHANGE OF DISCRETE VALUE.
	<b>Testing Hints</b>	

### 5.1.25Processes CHANGE\_OF\_TIMER Notifications

The IUT is capable of executing event notifications that convey a CHANGE\_OF\_TIMER event transition.

<b>135.1-2013 - 9.4.1 - ConfirmedEventNotification Using the Time Form of the ‘Timestamp’ Parameter and Conveying a Text Message,</b> <b>135.1-2013 - 9.4.2 - ConfirmedEventNotification Using the DateTime Form of the ‘Timestamp’ Parameter and no Text Message, or</b> <b>135.1-2013 - 9.4.3 - ConfirmedEventNotification Using the Sequence Number Form of the ‘Timestamp’ Parameter and no Text Message</b>		
--	--	--

	<b>Test Conditionality</b>	Must be executed if the device claims conformance to protocol revision 17 or higher.
	<b>Test Directives</b>	At least one of the tests must be executed with the Event Type set to CHANGE_OF_TIMER.
	<b>Testing Hints</b>	

### 5.1.26 Processes CHANGE\_OF\_RELIABILITY - FAULT\_OUT\_OF\_RANGE Notifications

The IUT is capable of executing event notifications that convey a CHANGE\_OF\_RELIABILITY - FAULT\_OUT\_OF\_RANGE event transition.

<b>135.1-2013 - 9.4.1 - ConfirmedEventNotification Using the Time Form of the ‘Timestamp’ Parameter and Conveying a Text Message,</b> <b>135.1-2013 - 9.4.2 - ConfirmedEventNotification Using the DateTime Form of the ‘Timestamp’ Parameter and no Text Message, or</b> <b>135.1-2013 - 9.4.3 - ConfirmedEventNotification Using the Sequence Number Form of the ‘Timestamp’ Parameter and no Text Message</b>		
	<b>Test Conditionality</b>	Must be executed if the device claims conformance to protocol revision 16 or higher.
	<b>Test Directives</b>	At least one of the tests must be executed with the Event Type set to CHANGE_OF_RELIABILITY - FAULT_OUT_OF_RANGE.
	<b>Testing Hints</b>	

### 5.1.27 Processes CHANGE\_OF\_LIFE\_SAFETY Notifications

The IUT is capable of executing event notifications that convey a CHANGE\_OF\_LIFE\_SAFETY event transition.

<b>135.1-2013 - 9.4.1 - ConfirmedEventNotification Using the Time Form of the ‘Timestamp’ Parameter and Conveying a Text Message,</b> <b>135.1-2013 - 9.4.2 - ConfirmedEventNotification Using the DateTime Form of the ‘Timestamp’ Parameter and no Text Message, or</b> <b>135.1-2013 - 9.4.3 - ConfirmedEventNotification Using the Sequence Number Form of the ‘Timestamp’ Parameter and no Text Message</b>		
	<b>Test Conditionality</b>	Must be executed if the device claims conformance to protocol revision 13 or higher.
	<b>Test Directives</b>	At least one of the tests must be executed with the Event Type set to CHANGE_OF_LIFE_SAFETY.
	<b>Testing Hints</b>	

### 5.1.28 Processes ACCESS\_EVENT Notifications

The IUT is capable of executing event notifications that convey an ACCESS\_EVENT event transition.

<b>135.1-2013 - 9.4.1 - ConfirmedEventNotification Using the Time Form of the ‘Timestamp’ Parameter and Conveying a Text Message,</b> <b>135.1-2013 - 9.4.2 - ConfirmedEventNotification Using the DateTime Form of the ‘Timestamp’ Parameter and no Text Message, or</b> <b>135.1-2013 - 9.4.3 - ConfirmedEventNotification Using the Sequence Number Form of the ‘Timestamp’ Parameter and no Text Message</b>		
	<b>Test Conditionality</b>	Must be executed if the device claims conformance to protocol revision 13 or higher.
	<b>Test Directives</b>	At least one of the tests must be executed with the Event Type set to ACCESS_EVENT.
	<b>Testing Hints</b>	

## 5.2 Alarm and Event Management - Notification - Internal - B

### 5.2.1 Base Requirements

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

<b>BTL - 7.3.1.10.1 - Event Enable Tests for TO_OFFNORMAL and TO_NORMAL</b>		
	<b>Test Conditionality</b>	If the IUT cannot be configured to meet the configuration requirements then this test shall be skipped.
	<b>Test Directives</b>	If Event Enrollment objects are supported, ensure this functionality is tested on Event Enrollment objects.
	<b>Testing Hints</b>	The BTL will apply this to a single object. The pretester should apply it to all objects that support alarm generation.
<b>135.1-2013 - 7.3.1.12 - Notify Type Test</b>		
	<b>Test Conditionality</b>	If the IUT cannot be configured to meet the 135.1-2013 configuration requirements then this test shall be skipped.
	<b>Test Directives</b>	If Event Enrollment objects are supported, ensure this functionality is also tested on Event Enrollment objects.
	<b>Testing Hints</b>	
<b>135.1-2013 - 8.4 - ConfirmedEventNotification Service Initiation Tests</b>		
	<b>Test Conditionality</b>	Must be executed unless IUT only supports read-only Recipient_List properties and does not claim Notification Forwarder objects. Any of the 8.4 tests can be used to ensure that the IUT properly generates ConfirmedEventNotification requests. The specific tests that can be executed are detailed under the test cases for the specific algorithms. As long as one of the tests is executed using ConfirmedEventNotifications, then this test case shall be satisfied.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 8.5 - UnconfirmedEventNotification Service Initiation Tests</b>		
	<b>Test Conditionality</b>	Must be executed. Any of the 8.5 tests can be used to ensure that the IUT properly generates UnconfirmedEventNotification requests. The specific tests that can be executed are detailed under the test cases for the specific algorithms. As long as one of the tests is executed using UnconfirmedEventNotifications, then this test case shall be satisfied.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.X9.1 - Event Detection Enable Inhibits Event Generation</b>		
	<b>Test Conditionality</b>	If Protocol Revision < 13, then this test shall be skipped.
	<b>Test Directives</b>	The objects selected by the tester should include all variants that differ in the set of supported alarming properties, or the writability of any of those properties. At least one instance of each variant should be selected.
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.X9.2 - Event Detection Enable Inhibits FAULT</b>		
	<b>Test Conditionality</b>	If Protocol Revision < 13, then this test shall be skipped.
	<b>Test Directives</b>	The objects selected by the tester should include all variants that differ in the set of supported alarming properties, or the writability of any of those properties. At least one instance of each variant should be selected.
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.X6.1 - Event Algorithm Inhibit Test</b>		

	<b>Test Conditionality</b>	If the IUT has no object in which the Event_Algorithm_Inhibit property is present and does not support the Event_Algorithm_Inhibit_Ref property, or has no object in which Event_Detection_Enable can be made TRUE, this test shall be skipped. If the IUT cannot be configured to contain any object capable of an event transition, then this test shall be skipped.
	<b>Test Directives</b>	The object types selected by the tester should include all variants that differ in the set of supported alarming properties, or the writability of any of those properties. At least one instance of each variant should be selected.
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.X7.1 - Event_Algorithm_Inhibit_Ref Test</b>		
	<b>Test Conditionality</b>	If the IUT has no object in which the Event_Algorithm_Inhibit_Ref property is present or has no object in which Event_Detection_Enable can be made TRUE, this test shall be skipped.
	<b>Test Directives</b>	The object types selected by the tester should include all variants that differ in the set of supported alarming properties, or the writability of any of those properties. At least one instance of each variant should be selected.
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.X7.2 - Event_Algorithm_Inhibit Writable Test</b>		
	<b>Test Conditionality</b>	If the IUT has no object in which the Event_Algorithm_Inhibit_Ref property is absent or can be made uninitialized or has no object in which Event_Detection_Enable can be made TRUE, this test shall be skipped.
	<b>Test Directives</b>	The object types selected by the tester should include all variants that differ in the set of supported alarming properties, or the writability of any of those properties. At least one instance of each variant should be selected.
	<b>Testing Hints</b>	
<b>BTL - 8.5.X9.10 - After FAULT-to-NORMAL, Re-Notification of OFFNORMAL</b>		
	<b>Test Conditionality</b>	If the IUT has no object in which CHANGE_OF_RELIABILITY is implemented in an object that can be configured into an offnormal state, this test shall be skipped.
	<b>Test Directives</b>	The objects selected by the tester should include all variants that differ in the set of supported alarming properties, or the writability of any of those properties. At least one instance of each variant shall be selected.
	<b>Testing Hints</b>	

### 5.2.2 Supports AE-INFO-B

The IUT must support AE-INFO-B if it claims support for AE-N-I-B.

Verify Checklist		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support for AE-INFO-B in the Checklist.
	<b>Testing Hints</b>	

### 5.2.3 Supports the Notification Class Object

The IUT supports the Notification Class object in order to send notifications.

Verify Checklist		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support for the Notification Class Object in the Checklist.
	<b>Testing Hints</b>	



### 5.2.4 Supports AE-ACK-B

The IUT supports AE-ACK-B in order to execute the AcknowledgeAlarm Service Service if the IUT is able to send event-notifications with service parameter AckRequired = True.

Verify Checklist		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support for AE-ACK-B in the Checklist.
	<b>Testing Hints</b>	

### 5.2.5 Implements Intrinsic Alarming

The IUT contains, or can be made to contain, an object other than an Event Enrollment object that can generate ConfirmedEventNotifications and UnconfirmedEventNotifications.

Verify Checklist		
	<b>Test Conditionality</b>	
	<b>Test Directives</b>	This functionality will be tested by the clause 8.4 or 8.5 algorithm tests listed later in this section.
	<b>Testing Hints</b>	

### 5.2.6 Supports the Event Enrollment object

The IUT contains, or can be made to contain an Event Enrollment object that can generate ConfirmedEventNotifications and UnconfirmedEventNotifications.

Verify Checklist		
	<b>Test Conditionality</b>	
	<b>Test Directives</b>	Ensure this functionality is tested on Event Enrollment objects by the clause 8.4 or 8.5 algorithm tests listed later in this section.
	<b>Testing Hints</b>	

### 5.2.7 Implements the CHANGE\_OF\_BITSTRING Algorithm

The IUT contains, or can be made to contain, an object that can generate ConfirmedEventNotifications and UnconfirmedEventNotifications with an Event\_Type of CHANGE\_OF\_BITSTRING.

135.1-2013 - 8.4.1 - CHANGE OF BITSTRING Tests (ConfirmedEventNotification)		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test must be repeated once for each object type that is capable of generating event notifications with an Event_Type of CHANGE_OF_BITSTRING.
	<b>Testing Hints</b>	
135.1-2013 - 8.5.1 - CHANGE OF BITSTRING Tests (UnconfirmedEventNotification)		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test must be repeated once for each object type that is capable of generating event notifications with an Event_Type of CHANGE_OF_BITSTRING.
	<b>Testing Hints</b>	

### 5.2.8 Implements the CHANGE\_OF\_STATE Algorithm

The IUT contains, or can be made to contain, an object that can generate ConfirmedEventNotifications and UnconfirmedEventNotifications with an Event\_Type of CHANGE\_OF\_STATE.

135.1-2013 - 8.4.2 - CHANGE OF STATE Tests (ConfirmedEventNotification)		
	<b>Test Conditionality</b>	Must be executed.

	<b>Test Directives</b>	This test must be repeated once for each object type that is capable of generating event notifications with an Event_Type of CHANGE_OF_STATE.
	<b>Testing Hints</b>	
	<b>135.1-2013 - 8.5.2 - CHANGE_OF_STATE Tests (UnconfirmedEventNotification)</b>	
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test must be repeated once for each object type that is capable of generating event notifications with an Event_Type of CHANGE_OF_STATE.
	<b>Testing Hints</b>	

### 5.2.9 Implements the Numeric Form of the CHANGE\_OF\_VALUE Algorithm

The IUT contains, or can be made to contain, an object that can generate ConfirmedEventNotifications and UnconfirmedEventNotifications with an Event\_Type of CHANGE\_OF\_VALUE where the monitored value is of data type Real.

<b>135.1-2013 - 8.4.3.1 - Numerical Algorithm (CHANGE_OF_VALUE Tests - ConfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test must be repeated once for each object type that is capable of generating event notifications with an Event_Type of CHANGE_OF_VALUE where the monitored value is of data type REAL.
	<b>Testing Hints</b>	
<b>135.1-2013 - 8.5.3.1 - Numerical Algorithm (CHANGE_OF_VALUE Tests - UnconfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test must be repeated once for each object type that is capable of generating event notifications with an Event_Type of CHANGE_OF_VALUE where the monitored value is of data type REAL.
	<b>Testing Hints</b>	

### 5.2.10 Implements the Bit String Form of the CHANGE\_OF\_VALUE Algorithm

The IUT contains, or can be made to contain, an object that can generate ConfirmedEventNotifications and UnconfirmedEventNotifications with an Event\_Type of CHANGE\_OF\_VALUE where the monitored value is of data type Bit String.

<b>135.1-2013 - 8.4.3.2 - Bitstring Algorithm (CHANGE_OF_VALUE Tests - ConfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test must be repeated once for each object type that is capable of generating event notifications with an Event_Type of CHANGE_OF_VALUE where the monitored value is of data type Bit String.
	<b>Testing Hints</b>	
<b>135.1-2013 - 8.5.3.2 - Bitstring Algorithm (CHANGE_OF_VALUE Tests - UnconfirmedEventNotification)</b>		

	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test must be repeated once for each object type that is capable of generating event notifications with an Event_Type of CHANGE_OF_VALUE where the monitored value is of data type Bit String.
	<b>Testing Hints</b>	

### 5.2.11 Implements the COMMAND\_FAILURE Algorithm

The IUT contains, or can be made to contain, an object that can generate ConfirmedEventNotifications and UnconfirmedEventNotifications with an Event\_Type of COMMAND\_FAILURE.

<b>BTL - 8.4.4 - COMMAND_FAILURE Tests (ConfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test must be repeated once for each object type that is capable of generating event notifications with an Event_Type of COMMAND_FAILURE.
	<b>Testing Hints</b>	
<b>BTL - 8.5.4 - COMMAND_FAILURE Tests (UnconfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test must be repeated once for each object type that is capable of generating event notifications with an Event_Type of COMMAND_FAILURE.
	<b>Testing Hints</b>	

### 5.2.12 Implements the FLOATING\_LIMIT Algorithm

The IUT contains, or can be made to contain, an object that can generate ConfirmedEventNotifications and UnconfirmedEventNotifications with an Event\_Type of FLOATING\_LIMIT.

<b>135.1-2013 - 8.4.5 - FLOATING_LIMIT Tests (ConfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test must be repeated once for each object type that is capable of generating event notifications with an Event_Type of FLOATING_LIMIT.
	<b>Testing Hints</b>	
<b>135.1-2013 - 8.5.5 - FLOATING_LIMIT Tests (UnconfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test must be repeated once for each object type that is capable of generating event notifications with an Event_Type of FLOATING_LIMIT.
	<b>Testing Hints</b>	

### 5.2.13 Implements the OUT\_OF\_RANGE Algorithm

The IUT contains, or can be made to contain, an object that can generate ConfirmedEventNotifications and UnconfirmedEventNotifications with an Event\_Type of OUT\_OF\_RANGE.

<b>135.1-2013 - 8.4.6 - OUT OF RANGE Tests (ConfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed.

	<b>Test Directives</b>	This test must be repeated once for each object type that is capable of generating event notifications with an Event_Type of OUT_OF_RANGE.
	<b>Testing Hints</b>	
<b>135.1-2013 - 8.5.6 - OUT OF RANGE Tests (UnconfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test must be repeated once for each object type that is capable of generating event notifications with an Event_Type of OUT_OF_RANGE.
	<b>Testing Hints</b>	

### 5.2.14 Implements a Proprietary Algorithm Using Complex Notifications

The IUT contains, or can be made to contain, an object that can generate ConfirmedEventNotifications and UnconfirmedEventNotifications with an Event\_Type of COMPLEX\_EVENT\_TYPE.

<b>BTL - 8.4.X5 - Proprietary Algorithm Tests (ConfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	<i>This test is not yet defined and shall be skipped.</i>
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 8.5.X5 - Proprietary Algorithm Tests (UnconfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	<i>This test is not yet defined and shall be skipped.</i>
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 5.2.15 Implements a Standard or Proprietary Algorithm Using Extended Notifications

The IUT contains, or can be made to contain, an object that can generate ConfirmedEventNotifications and UnconfirmedEventNotifications with an Event\_Type of EXTENDED.

<b>BTL - 8.4.X6 - Extended Algorithm Tests (ConfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 8.5.X6 - Extended Algorithm Tests (UnconfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 5.2.16 Generates Event Notifications with Timestamps of the BacnetDateTime Form

The IUT generates, or can be made to generate, ConfirmedEventNotifications and/or UnconfirmedEventNotifications with the Time Stamp parameter taking the BACnetDateTime form.

<b>135.1-2013 - 8.4 - ConfirmedEventNotification Service Initiation Tests</b>		
	<b>Test Conditionality</b>	Any of the 8.4 tests can be used to ensure that the IUT properly generates ConfirmedEventNotification requests using the BACnetDateTime form. The specific tests that can be executed are detailed under the test cases for the specific algorithms. As long as one of the tests is executed using ConfirmedEventNotifications and the notification that is generated contains a timestamp of the BACnetDateTime form, then this test case shall be satisfied.
	<b>Test Directives</b>	

	<b>Testing Hints</b>	
<b>135.1-2013 - 8.5 - UnconfirmedEventNotification Service Initiation Tests</b>		
	<b>Test Conditionality</b>	Any of the 8.5 tests can be used to ensure that the IUT properly generates UnconfirmedEventNotification requests using the BACnetDateTime form. The specific tests that can be executed are detailed under the test cases for the specific algorithms. As long as one of the tests is executed using UnconfirmedEventNotifications and the notification that is generated contains a timestamp of the BACnetDateTime form, then this test case shall be satisfied.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 5.2.17Generates Event Notifications with Timestamps of the Time Form

The IUT generates, or can be made to generate, ConfirmedEventNotifications and/or UnconfirmedEventNotifications with the Time Stamp parameter taking the Time form.

<b>135.1-2013 - 8.4 - ConfirmedEventNotification Service Initiation Tests</b>		
	<b>Test Conditionality</b>	Any of the 8.4 tests can be used to ensure that the IUT properly generates ConfirmedEventNotification requests using the Time form. The specific tests that can be executed are detailed under the test cases for the specific algorithms. As long as one of the tests is executed using ConfirmedEventNotifications and the notification that is generated contains a timestamp of the Time form, then this test case shall be satisfied.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 8.5 - UnconfirmedEventNotification Service Initiation Tests</b>		
	<b>Test Conditionality</b>	Any of the 8.5 tests can be used to ensure that the IUT properly generates UnconfirmedEventNotification requests using the Time form. The specific tests that can be executed are detailed under the test cases for the specific algorithms. As long as one of the tests is executed using UnconfirmedEventNotifications and the notification that is generated contains a timestamp of the Time form, then this test case shall be satisfied.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 5.2.18Generates Event Notifications with Timestamps of the Sequence Number Form

The IUT generates, or can be made to generate, ConfirmedEventNotifications and/or UnconfirmedEventNotifications with the Time Stamp parameter taking the Sequence Number form.

<b>135.1-2013 - 8.4 - ConfirmedEventNotification Service Initiation Tests</b>		
	<b>Test Conditionality</b>	Any of the 8.4 tests can be used to ensure that the IUT properly generates ConfirmedEventNotification requests using the Sequence Number form. The specific tests that can be executed are detailed under the test cases for the specific algorithms. As long as one of the tests is executed using ConfirmedEventNotifications and the notification that is generated contains a timestamp of the Sequence Number form, then this test case shall be satisfied.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 8.5 - UnconfirmedEventNotification Service Initiation Tests</b>		

	<b>Test Conditionality</b>	Any of the 8.5 tests can be used to ensure that the IUT properly generates UnconfirmedEventNotification requests using the Time form. The specific tests that can be executed are detailed under the test cases for the specific algorithms. As long as one of the tests is executed using UnconfirmedEventNotifications and the notification that is generated contains a timestamp of the Sequence Number form, then this test case shall be satisfied.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 5.2.19 Implements Intrinsic Alarming in an Analog object

The IUT contains, or can be made to contain, an Analog object that can generate ConfirmedEventNotifications and UnconfirmedEventNotifications and thus contains a Limit\_Enable property.

<b>BTL - 7.3.1.13.1 - Limit_Enable Test, LowLimitEnable</b>		
	<b>Test Conditionality</b>	If Limit_Enable is not configurable, this test may be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.13.2 - Limit_Enable Test, HighLimitEnable</b>		
	<b>Test Conditionality</b>	If Limit_Enable is not configurable, this test may be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 5.2.20 Supports Writable Event\_Parameters Properties

The IUT contains, or can be made to contain, an Event Enrollment object with a writable Event\_Parameters property.

<b>135.1-2013 - 7.3.2.11.1 - Event_Type Test</b>		
	<b>Test Conditionality</b>	If the IUT has a Protocol_Revision of 3 or less or if no Event_Enrollment object exists, or can be made to exist, in the IUT that accepts different CHOICES of Event_Parameters, then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 5.2.21 Implements the DOUBLE\_OUT\_OF\_RANGE Algorithm

The IUT contains, or can be made to contain, an object that can generate ConfirmedEventNotifications and UnconfirmedEventNotifications with an Event\_Type of DOUBLE\_OUT\_OF\_RANGE.

<b>BTL - 8.4.X1 - DOUBLE_OUT_OF_RANGE Tests (ConfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test must be repeated once for each object type that is capable of generating event notifications with an Event_Type of DOUBLE_OUT_OF_RANGE.
	<b>Testing Hints</b>	
<b>BTL - 8.5.X1 - DOUBLE_OUT_OF_RANGE Tests (UnconfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test must be repeated once for each object type that is capable of generating event notifications with an Event_Type of DOUBLE_OUT_OF_RANGE.
	<b>Testing Hints</b>	

### 5.2.22 Implements the SIGNED\_OUT\_OF\_RANGE Algorithm

The IUT contains, or can be made to contain, an object that can generate ConfirmedEventNotifications and UnconfirmedEventNotifications with an Event\_Type of SIGNED\_OUT\_OF\_RANGE.

<b>BTL - 8.4.X2 - SIGNED_OUT_OF_RANGE Tests (ConfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test must be repeated once for each object type that is capable of generating event notifications with an Event_Type of SIGNED_OUT_OF_RANGE.
	<b>Testing Hints</b>	
<b>BTL - 8.5.X2 - SIGNED_OUT_OF_RANGE Tests (UnconfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test must be repeated once for each object type that is capable of generating event notifications with an Event_Type of SIGNED_OUT_OF_RANGE.
	<b>Testing Hints</b>	

### 5.2.23 Implements the UNSIGNED\_OUT\_OF\_RANGE Algorithm

The IUT contains, or can be made to contain, an object that can generate ConfirmedEventNotifications and UnconfirmedEventNotifications with an Event\_Type of UNSIGNED\_OUT\_OF\_RANGE.

<b>BTL - 8.4.X3 - UNSIGNED_OUT_OF_RANGE Tests (ConfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test must be repeated once for each object type that is capable of generating event notifications with an Event_Type of UNSIGNED_OUT_OF_RANGE.
	<b>Testing Hints</b>	
<b>BTL - 8.5.X3 - UNSIGNED_OUT_OF_RANGE Tests (UnconfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test must be repeated once for each object type that is capable of generating event notifications with an Event_Type of UNSIGNED_OUT_OF_RANGE.
	<b>Testing Hints</b>	

### 5.2.24 Implements the CHANGE\_OF\_CHARACTERSTRING Algorithm

The IUT contains, or can be made to contain, an object that can generate ConfirmedEventNotifications and UnconfirmedEventNotifications with an Event\_Type of CHANGE\_OF\_CHARACTERSTRING.

<b>BTL - 8.4.X4 - CHANGE_OF_CHARACTERSTRING Tests (ConfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test must be repeated once for each object type that is capable of generating event notifications with an Event_Type of CHANGE_OF_CHARACTERSTRING.
	<b>Testing Hints</b>	
<b>BTL - 8.5.X4 - CHANGE_OF_CHARACTERSTRING Tests (UnconfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test must be repeated once for each object type that is capable of generating event notifications with an Event_Type of CHANGE_OF_CHARACTERSTRING.
	<b>Testing Hints</b>	

### 5.2.25 Implements the CHANGE\_OF\_STATUS\_FLAGS Algorithm

The IUT contains, or can be made to contain, an object that can generate ConfirmedEventNotifications and UnconfirmedEventNotifications with an Event\_Type of CHANGE\_OF\_STATUS\_FLAGS.

<b>BTL - 8.4.X8 - CHANGE_OF_STATUS_FLAGS Test (ConfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test must be repeated once for each object type that is capable of generating event notifications with an Event_Type of CHANGE_OF_STATUS_FLAGS.
	<b>Testing Hints</b>	
<b>BTL - 8.5.X8 - CHANGE_OF_STATUS_FLAGS Test (UnconfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test must be repeated once for each object type that is capable of generating event notifications with an Event_Type of CHANGE_OF_STATUS_FLAGS.
	<b>Testing Hints</b>	

### 5.2.26 Implements the UNSIGNED\_RANGE Algorithm

The IUT contains, or can be made to contain, an object such as an Accumulator object, that can generate EventNotifications with an Event\_Type of UNSIGNED\_RANGE.

<b>BTL - 8.4.X7 - UNSIGNED_RANGE ConfirmedEventNotification Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test must be repeated once for each object type that is capable of generating event notifications with an Event_Type of UNSIGNED_RANGE.
	<b>Testing Hints</b>	
<b>BTL - 8.5.X7 - UNSIGNED_RANGE UnconfirmedEventNotification Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test must be repeated once for each object type that is capable of generating event notifications with an Event_Type of UNSIGNED_RANGE.
	<b>Testing Hints</b>	

### 5.2.27 Supports Event\_Message\_Texts Property

The IUT contains one or more objects that support the Event\_Message\_Texts property.

<b>BTL - 7.3.1.X4 - Event_Message_Texts Tests</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Repeat test once for each object type in the IUT that contains an Event_Message_Texts property.
	<b>Testing Hints</b>	

### 5.2.28 Supports Event\_Message\_Texts\_Config Property

The IUT contains one or more objects that support the Event\_Message\_Texts\_Config property.

<b>BTL - 7.3.1.X5 - Event_Message_Texts_Config Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Repeat for each supported transition type (TO_OFFNORMAL, TO_FAULT, TO_NORMAL). Different objects may be selected for different transitions.
	<b>Testing Hints</b>	



### 5.2.29 Implements Intrinsic Alarming in an Integer object

The IUT contains, or can be made to contain, an Integer object that can generate ConfirmedEventNotifications and UnconfirmedEventNotifications and thus contains a Limit\_Enable property.

BTL - 7.3.1.13.1 - Limit_Enable Test, LowLimitEnable		
	Test Conditionality	If Limit_Enable is not configurable, this test may be skipped.
	Test Directives	
	Testing Hints	
BTL - 7.3.1.13.2 - Limit_Enable Test, HighLimitEnable		
	Test Conditionality	If Limit_Enable is not configurable, this test may be skipped.
	Test Directives	
	Testing Hints	

### 5.2.30 Implements the CHANGE\_OF\_RELIABILITY - NONE

The IUT contains, or can be made to contain, an object that can generate EventNotifications with an Event\_Type of CHANGE\_OF\_RELIABILITY and supports the specified algorithm.

BTL - 8.5.X9.1 - CHANGE_OF_RELIABILITY with No Fault Algorithm		
	Test Conditionality	The IUT contains, or can be made to contain, an object that can generate EventNotifications with an Event_Type of CHANGE_OF_RELIABILITY and which does not apply a standardized fault algorithm.
	Test Directives	Apply this test to all object types that support fault detection but do not apply a standardized fault algorithm.
	Testing Hints	

### 5.2.31 Implements the CHANGE\_OF\_RELIABILITY - FAULT\_CHARACTERSTRING Algorithm

The IUT contains, or can be made to contain, an object that can generate EventNotifications with an Event\_Type of CHANGE\_OF\_RELIABILITY and supports the specified algorithm.

BTL - 8.5.X9.2 - CHANGE_OF_RELIABILITY with the FAULT_CHARACTERSTRING Algorithm		
	Test Conditionality	Must be executed.
	Test Directives	The objects selected by the tester should include all variants that differ in the set of supported alarming properties, or the writability of any of those properties. At least one instance of each variant shall be selected.
	Testing Hints	

### 5.2.32 Implements the CHANGE\_OF\_RELIABILITY - FAULT\_EXTENDED Algorithm

The IUT contains, or can be made to contain, an object that can generate EventNotifications with an Event\_Type of CHANGE\_OF\_RELIABILITY and supports the specified algorithm.

BTL - 8.5.X9.3 - CHANGE_OF_RELIABILITY with the FAULT_EXTENDED Algorithm		
	Test Conditionality	Must be executed.
	Test Directives	The objects selected by the tester should include all variants that differ in the set of supported alarming properties, or the writability of any of those properties. At least one instance of each variant shall be selected.

	Testing Hints	
--	---------------	--

### 5.2.33 Implements the CHANGE\_OF\_RELIABILITY - FAULT\_LIFE\_SAFETY Algorithm

The IUT contains, or can be made to contain, an object that can generate EventNotifications with an Event\_Type of CHANGE\_OF\_RELIABILITY and supports the specified algorithm.

BTL - 8.5.X9.4 - CHANGE OF RELIABILITY with the FAULT_LIFE_SAFETY Algorithm		
	Test Conditionality	Must be executed.
	Test Directives	The objects selected by the tester should include all variants that differ in the set of supported alarming properties, or the writability of any of those properties. At least one instance of each variant shall be selected.
	Testing Hints	

### 5.2.34 Implements the CHANGE\_OF\_RELIABILITY - FAULT\_STATE Algorithm

The IUT contains, or can be made to contain, an object that can generate EventNotifications with an Event\_Type of CHANGE\_OF\_RELIABILITY and supports the specified algorithm.

BTL - 8.5.X9.5 - CHANGE OF RELIABILITY with the FAULT_STATE Algorithm		
	Test Conditionality	Must be executed.
	Test Directives	The objects selected by the tester should include all variants that differ in the set of supported alarming properties, or the writability of any of those properties. At least one instance of each variant shall be selected.
	Testing Hints	

### 5.2.35 Implements the CHANGE\_OF\_RELIABILITY - FAULT\_STATUS\_FLAGS Algorithm

The IUT contains, or can be made to contain, an object that can generate EventNotifications with an Event\_Type of CHANGE\_OF\_RELIABILITY and supports the specified algorithm.

BTL - 8.5.X9.6 - CHANGE OF RELIABILITY with the FAULT_STATUS_FLAGS Algorithm		
	Test Conditionality	Must be executed.
	Test Directives	The objects selected by the tester should include all variants that differ in the set of supported alarming properties, or the writability of any of those properties. At least one instance of each variant shall be selected.
	Testing Hints	

### 5.2.36 Implements the CHANGE\_OF\_RELIABILITY - FAULT\_LISTED algorithm

The IUT contains, or can be made to contain, an object that can generate EventNotifications with an Event\_Type of CHANGE\_OF\_RELIABILITY and supports the specified algorithm.

Contact BTL for interim tests for this algorithm.

### 5.2.37 Supports CHANGE\_OF\_RELIABILITY in the Event Enrollment Object

The IUT contains, or can be made to contain, an Event Enrollment object that can generate EventNotifications with an Event\_Type of CHANGE\_OF\_RELIABILITY.

<b>BTL - 8.5.X9.7.1 - Internal Faults Take Precedence Over Monitored Object Faults</b>		
	<b>Test Conditionality</b>	If the IUT does not support an Event Enrollment object which can detect internal faults and monitor an object which detects faults, then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 8.5.X9.7.2 - Monitored Object Faults Take Precedence Over Fault Algorithms</b>		
	<b>Test Conditionality</b>	If the IUT does not support an Event Enrollment object which monitors an object which detects faults and which applies a fault algorithm, then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 8.5.X9.7.3 - Internal Faults Take Precedence Over Fault Algorithms</b>		
	<b>Test Conditionality</b>	If the IUT does not support an Event Enrollment object which can detect internal faults and which applies a fault algorithm, then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 8.5.X9.8 - CHANGE_OF_RELIABILITY of Event Enrollment Object, Monitored Object Fault</b>		
	<b>Test Conditionality</b>	If the IUT has no Event Enrollment object where the Monitored_Object that can transition to fault, this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 8.5.X9.9 - CHANGE_OF_RELIABILITY of Event Enrollment Object Fault</b>		
	<b>Test Conditionality</b>	If the IUT has no Event Enrollment object that detects an internal unreliable operational fault, this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 5.2.38 Implements the CHANGE\_OF\_LIFE\_SAFETY Algorithm

Contact the BTL for interim tests for this algorithm.

### 5.2.39 Implements the ACCESS\_EVENT Algorithm

Contact the BTL for interim tests for this algorithm.

### 5.2.40 Implements the CHANGE\_OF\_DISCRETE\_VALUE Algorithm

The IUT contains, or can be made to contain, an object that can generate EventNotifications with an Event\_Type of CHANGE\_OF\_DISCRETE\_VALUE.

<b>BTL - 8.4.X10 - CHANGE_OF_DISCRETE_VALUE Test (ConfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test must be repeated for each datatype, BOOLEAN, Unsigned, Integer, Enumerated, CharacterString, Octet String, Date, Time, BACnetObjectIdentifier, or BACnetDateTime supported by the Object_Property_Reference property of the Event Enrollment object.
	<b>Testing Hints</b>	

<b>BTL - 8.5.X10 - CHANGE OF DISCRETE VALUE Test (UnconfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test must be repeated for each datatype, BOOLEAN, Unsigned, Integer, Enumerated, CharacterString, Octet String, Date, Time, BACnetObjectIdentifier, or BACnetDateTime supported by the Object Property Reference property of the Event Enrollment object.
	<b>Testing Hints</b>	

### 5.2.41 Implements the CHANGE\_OF\_TIMER Algorithm

Contact the BTL for interim tests for this algorithm.

### 5.2.42 Implements the CHANGE\_OF\_RELIABILITY - FAULT\_OUT\_OF\_RANGE Algorithm

The IUT contains, or can be made to contain, an object that can generate EventNotifications with an Event\_Type of CHANGE\_OF\_RELIABILITY and supports the specified algorithm.

<b>BTL - 8.4.X9 - CHANGE_OF_RELIABILITY with the FAULT_OUT_OF_RANGE Algorithm (ConfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed
	<b>Test Directives</b>	<p>This test shall be executed for different Reliability transitions which are supported in the IUT such as No_Fault_Detected -Under_Range, Under_Range - No_Fault_Detected, No_Fault_Detected - OverRange, Over_Range - No_Fault_Detected, No_Fault_Detected - Under_Range, Under_Range - Over_Range, Over_Range - Under_Range and Under_Range - No_Fault_Detected.</p> <p>This test must be repeated once for each object type that is capable of generating event notifications with an Event_Type of CHANGE_OF_RELIABILITY with the FAULT_OUT_OF_RANGE</p>
	<b>Testing Hints</b>	
<b>BTL - 8.5.X9 - CHANGE_OF_RELIABILITY with the FAULT_OUT_OF_RANGE Algorithm (UnconfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed
	<b>Test Directives</b>	<p>This test shall be executed for different Reliability transitions which are supported in the IUT such as No_Fault_Detected -Under_Range, Under_Range - No_Fault_Detected, No_Fault_Detected - OverRange, Over_Range - No_Fault_Detected, No_Fault_Detected - Under_Range, Under_Range - Over_Range, Over_Range - Under_Range and Under_Range - No_Fault_Detected.</p> <p>This test must be repeated once for each object type that is capable of generating event notifications with an Event_Type of CHANGE_OF_RELIABILITY with the FAULT_OUT_OF_RANGE</p>
	<b>Testing Hints</b>	

## 5.3 Alarm and Event Management - Notification - External - B

### 5.3.1 Base Requirements

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

<b>BTL - 8.5.X9.10 - After FAULT-to-NORMAL, Re-Notification of OFFNORMAL</b>		
	<b>Test Conditionality</b>	If the IUT has no object in which CHANGE_OF_RELIABILITY is implemented and which can be configured into an offnormal state, this test shall be skipped.
	<b>Test Directives</b>	The objects selected by the tester should include all variants that differ in the set of supported alarming properties, or the writability of any of those properties. At least one instance of each variant shall be selected.
	<b>Testing Hints</b>	

### 5.3.2 Supports AE-N-I-B

The IUT must support AE-N-I-B if it claims support for AE-N-E-B.

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

### 5.3.3 Supports DS-RP-A for Retrieving Monitored Values

The IUT is able to use ReadProperty to retrieve property values that are monitored via an Event Enrollment object.

<b>135.1-2013 - 8.4 - ConfirmedEventNotification Service Initiation Tests or 135.1-2013 - 8.5 - UnconfirmedEventNotification Service Initiation Tests</b>		
	<b>Test Conditionality</b>	The specific tests that can be executed are detailed under the test cases for the specific algorithms. If at least one of the referenced tests is executed with the monitored object in a device that does not support ReadPropertyMultiple or SubscribeCOV, then this test case shall be satisfied.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	This functionality will be tested by the clause 8.4 or 8.5 algorithm tests listed later in this section.

### 5.3.4 Supports the Event Enrollment Object

The IUT contains, or can be made to contain an Event Enrollment object, that can generate ConfirmedEventNotifications and UnconfirmedEventNotifications.

<b>Verify Checklist</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support for the Event Enrollment object in the Checklist. The functionality will be tested by the clause 8.4 or 8.5 algorithm tests listed later in this section.
	<b>Testing Hints</b>	

### 5.3.5 Implements the CHANGE\_OF\_BITSTRING Algorithm

The IUT contains, or can be made to contain, an Event Enrollment object that can generate ConfirmedEventNotifications and UnconfirmedEventNotifications with an Event\_Type of CHANGE\_OF\_BITSTRING.

<b>135.1-2013 - 8.4.1 - CHANGE OF BITSTRING Tests (ConfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test shall be executed with an Event Enrollment object that is configured to monitor a property in a device other than the IUT.
	<b>Testing Hints</b>	
<b>135.1-2013 - 8.5.1 - CHANGE OF BITSTRING Tests (UnconfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test shall be executed with an Event Enrollment object that is configured to monitor a property in a device other than the IUT.
	<b>Testing Hints</b>	

### 5.3.6 Implements the CHANGE\_OF\_STATE Algorithm

The IUT contains, or can be made to contain, an Event Enrollment object that can generate ConfirmedEventNotifications and UnconfirmedEventNotifications with an Event\_Type of CHANGE\_OF\_STATE.

<b>135.1-2013 - 8.4.2 - CHANGE OF STATE Tests (ConfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test shall be executed with an Event Enrollment object that is configured to monitor a property in a device other than the IUT.
	<b>Testing Hints</b>	
<b>135.1-2013 - 8.5.2 - CHANGE OF STATE Tests (UnconfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test shall be executed with an Event Enrollment object that is configured to monitor a property in a device other than the IUT.
	<b>Testing Hints</b>	

### 5.3.7 Implements the Numeric Form of the CHANGE\_OF\_VALUE Algorithm

The IUT contains, or can be made to contain, an Event Enrollment object that can generate ConfirmedEventNotifications and UnconfirmedEventNotifications with an Event\_Type of CHANGE\_OF\_VALUE where the monitored value is of data type Real.

<b>135.1-2013 - 8.4.3.1 - Numerical Algorithm (CHANGE_OF_VALUE Tests - ConfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test shall be executed with an Event Enrollment object that is configured to monitor a property in a device other than the IUT.
	<b>Testing Hints</b>	
<b>135.1-2013 - 8.5.3.1 - Numerical Algorithm (CHANGE_OF_VALUE Tests - UnconfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test shall be executed with an Event Enrollment object that is configured to monitor a property in a device other than the IUT.
	<b>Testing Hints</b>	

### 5.3.8 Implements the Bit String Form of the CHANGE\_OF\_VALUE Algorithm

The IUT contains, or can be made to contain, an Event Enrollment object that can generate ConfirmedEventNotifications and UnconfirmedEventNotifications with an Event\_Type of CHANGE\_OF\_VALUE where the monitored value is of datatype bit string.

<b>135.1-2013 - 8.4.3.2 - Bitstring Algorithm (CHANGE_OF_VALUE Tests - ConfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test shall be executed with an Event Enrollment object that is configured to monitor a property in a device other than the IUT.
	<b>Testing Hints</b>	
<b>135.1-2013 - 8.5.3.2 - Bitstring Algorithm (CHANGE_OF_VALUE Tests - UnconfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test shall be executed with an Event Enrollment object that is configured to monitor a property in a device other than the IUT.
	<b>Testing Hints</b>	

### 5.3.9 Implements the COMMAND\_FAILURE Algorithm

The IUT contains, or can be made to contain, an Event Enrollment object that can generate ConfirmedEventNotifications and UnconfirmedEventNotifications with an Event\_Type of COMMAND\_FAILURE.

<b>135.1-2013 - 8.4.4 - COMMAND FAILURE Tests (ConfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test shall be executed with an Event Enrollment object that is configured to monitor a property in a device other than the IUT.
	<b>Testing Hints</b>	
<b>135.1-2013 - 8.5.4 - COMMAND FAILURE Tests (UnconfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test shall be executed with an Event Enrollment object that is configured to monitor a property in a device other than the IUT.
	<b>Testing Hints</b>	

### 5.3.10 Implements the FLOATING\_LIMIT Algorithm

The IUT contains, or can be made to contain, an Event Enrollment object that can generate ConfirmedEventNotifications and UnconfirmedEventNotifications with an Event\_Type of FLOATING\_LIMIT.

<b>135.1-2013 - 8.4.5 - FLOATING_LIMIT Tests (ConfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test shall be executed with an Event Enrollment object that is configured to monitor a property in a device other than the IUT.
	<b>Testing Hints</b>	
<b>135.1-2013 - 8.5.5 - FLOATING_LIMIT Tests (UnconfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test shall be executed with an Event Enrollment object that is configured to monitor a property in a device other than the IUT.
	<b>Testing Hints</b>	

### 5.3.11 Implements the OUT\_OF\_RANGE Algorithm

The IUT contains, or can be made to contain, an Event Enrollment object that can generate ConfirmedEventNotifications and UnconfirmedEventNotifications with an Event\_Type of OUT\_OF\_RANGE.

<b>135.1-2013 - 8.4.6 - OUT OF RANGE Tests (ConfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test shall be executed with an Event Enrollment object that is configured to monitor a property in a device other than the IUT.

	<b>Testing Hints</b>	
<b>135.1-2013 - 8.5.6 - OUT OF RANGE Tests (UnconfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test shall be executed with an Event Enrollment object that is configured to monitor a property in a device other than the IUT.
	<b>Testing Hints</b>	

### 5.3.12 Implements the DOUBLE\_OUT\_OF\_RANGE Algorithm

The IUT contains, or can be made to contain, an Event Enrollment object that can generate ConfirmedEventNotifications and UnconfirmedEventNotifications with an Event\_Type of DOUBLE\_OUT\_OF\_RANGE.

<b>BTL - 8.4.X1 - DOUBLE OUT OF RANGE Tests (ConfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test shall be executed with an Event Enrollment object that is configured to monitor a property in a device other than the IUT
	<b>Testing Hints</b>	
<b>BTL - 8.5.X1 - DOUBLE OUT OF RANGE Tests (UnconfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test shall be executed with an Event Enrollment object that is configured to monitor a property in a device other than the IUT
	<b>Testing Hints</b>	

### 5.3.13 Implements the SIGNED\_OUT\_OF\_RANGE Algorithm

The IUT contains, or can be made to contain, an Event Enrollment object that can generate ConfirmedEventNotifications and UnconfirmedEventNotifications with an Event\_Type of SIGNED\_OUT\_OF\_RANGE.

<b>BTL - 8.4.X2 - SIGNED OUT OF RANGE Tests (ConfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test shall be executed with an Event Enrollment object that is configured to monitor a property in a device other than the IUT
	<b>Testing Hints</b>	
<b>BTL - 8.5.X2 - SIGNED OUT OF RANGE Tests (UnconfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test shall be executed with an Event Enrollment object that is configured to monitor a property in a device other than the IUT
	<b>Testing Hints</b>	

### 5.3.14 Implements the UNSIGNED\_OUT\_OF\_RANGE Algorithm

The IUT contains, or can be made to contain, an Event Enrollment object that can generate ConfirmedEventNotifications and UnconfirmedEventNotifications with an Event\_Type of UNSIGNED\_OUT\_OF\_RANGE.

<b>BTL - 8.4.X3 - UNSIGNED OUT OF RANGE Tests (ConfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test shall be executed with an Event Enrollment object that is configured to monitor a property in a device other than the IUT
	<b>Testing Hints</b>	
<b>BTL - 8.5.X3 - UNSIGNED OUT OF RANGE Tests (UnconfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test shall be executed with an Event Enrollment object that is configured to monitor a property in a device other than the IUT
	<b>Testing Hints</b>	



### 5.3.15 Implements the CHANGE\_OF\_CHARACTERSTRING Algorithm

The IUT contains, or can be made to contain, an Event Enrollment object that can generate ConfirmedEventNotifications and UnconfirmedEventNotifications with an Event\_Type of CHANGE\_OF\_CHARACTERSTRING.

<b>BTL - 8.4.X4 - CHANGE OF CHARACTERSTRING Tests (ConfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test shall be executed with an Event Enrollment object that is configured to monitor a property in a device other than the IUT
	<b>Testing Hints</b>	
<b>BTL - 8.5.X4 - CHANGE OF CHARACTERSTRING Tests (UnconfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test shall be executed with an Event Enrollment object that is configured to monitor a property in a device other than the IUT
	<b>Testing Hints</b>	

### 5.3.16 Implements the CHANGE\_OF\_STATUS\_FLAGS Algorithm

The IUT contains, or can be made to contain, an Event Enrollment object that can generate ConfirmedEventNotifications and UnconfirmedEventNotifications with an Event\_Type of CHANGE\_OF\_STATUS\_FLAGS.

<b>BTL - 8.4.X8 - CHANGE OF STATUS_FLAGS Test (ConfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test must be repeated once for each object type that is capable of generating event notifications with an Event_Type of CHANGE_OF_STATUS_FLAGS.
	<b>Testing Hints</b>	
<b>BTL - 8.5.X8 - CHANGE OF STATUS_FLAGS Test (UnconfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test must be repeated once for each object type that is capable of generating event notifications with an Event_Type of CHANGE_OF_STATUS_FLAGS.
	<b>Testing Hints</b>	

### 5.3.17 Implements the UNSIGNED\_RANGE Algorithm

The IUT contains, or can be made to contain an Event Enrollment object that can generate EventNotifications with an Event\_Type of UNSIGNED\_RANGE.

<b>BTL - 8.4.X7 - UNSIGNED_RANGE ConfirmedEventNotification Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test shall be executed with an Event Enrollment object that is configured to monitor a property in a device other than the IUT.
	<b>Testing Hints</b>	
<b>BTL - 8.5.X7 - UNSIGNED_RANGE UnconfirmedEventNotification Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test shall be executed with an Event Enrollment object that is configured to monitor a property in a device other than the IUT.
	<b>Testing Hints</b>	

### 5.3.18 Implements the CHANGE\_OF\_RELIABILITY - NONE

The IUT contains, or can be made to contain, an object that can generate EventNotifications with an Event\_Type of CHANGE\_OF\_RELIABILITY and supports the specified algorithm.

<b>BTL - 8.5.X9.1 - CHANGE OF RELIABILITY with No Fault Algorithm</b>		
	<b>Test Conditionality</b>	The IUT contains, or can be made to contain, an object that can generate EventNotifications with an Event_Type of CHANGE_OF_RELIABILITY and which does not apply a standardized fault algorithm.
	<b>Test Directives</b>	Apply this test to all object types that support fault detection but do not apply a standardized fault algorithm.
	<b>Testing Hints</b>	

### 5.3.19 Implements the CHANGE\_OF\_RELIABILITY - FAULT\_CHARACTERSTRING Algorithm

The IUT contains, or can be made to contain, an object that can generate EventNotifications with an Event\_Type of CHANGE\_OF\_RELIABILITY and supports the specified algorithm.

<b>BTL - 8.5.X9.2 - CHANGE_OF_RELIABILITY with the FAULT_CHARACTERSTRING Algorithm</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Repeat the test for each object type that support this fault algorithm,
	<b>Testing Hints</b>	

### 5.3.20 Implements the CHANGE\_OF\_RELIABILITY - FAULT\_EXTENDED Algorithm

The IUT contains, or can be made to contain, an object that can generate EventNotifications with an Event\_Type of CHANGE\_OF\_RELIABILITY and supports the specified algorithm.

<b>BTL - 8.5.X9.3 - CHANGE OF RELIABILITY with the FAULT_EXTENDED Algorithm</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Repeat the test for each object type that support this fault algorithm,
	<b>Testing Hints</b>	

### 5.3.21 Implements the CHANGE\_OF\_RELIABILITY - FAULT\_LIFE\_SAFETY Algorithm

The IUT contains, or can be made to contain, an object that can generate EventNotifications with an Event\_Type of CHANGE\_OF\_RELIABILITY and supports the specified algorithm.

<b>BTL - 8.5.X9.4 - CHANGE_OF_RELIABILITY with the FAULT_LIFE_SAFETY Algorithm</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Repeat the test for each object type that support this fault algorithm,
	<b>Testing Hints</b>	

### 5.3.22 Implements the CHANGE\_OF\_RELIABILITY - FAULT\_STATE Algorithm

The IUT contains, or can be made to contain, an object that can generate EventNotifications with an Event\_Type of CHANGE\_OF\_RELIABILITY and supports the specified algorithm.

<b>BTL - 8.5.X9.5 - CHANGE OF RELIABILITY with the FAULT STATE Algorithm</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Repeat the test for each object type that support this fault algorithm,
	<b>Testing Hints</b>	

### 5.3.23 Implements the CHANGE\_OF\_RELIABILITY - FAULT\_STATUS\_FLAGS Algorithm

The IUT contains, or can be made to contain, an object that can generate EventNotifications with an Event\_Type of CHANGE\_OF\_RELIABILITY and supports the specified algorithm.

<b>BTL - 8.5.X9.6 - CHANGE OF RELIABILITY with the FAULT_STATUS_FLAGS Algorithm</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Repeat the test for each object type that support this fault algorithm,
	<b>Testing Hints</b>	

### 5.3.24 Implements the CHANGE\_OF\_RELIABILITY - FAULT\_LISTED Algorithm

The IUT contains, or can be made to contain, an object that can generate EventNotifications with an Event\_Type of CHANGE\_OF\_RELIABILITY and supports the specified algorithm.

Contact BTL for interim tests for this algorithm.

### 5.3.25 Supports CHANGE\_OF\_RELIABILITY in the Event Enrollment Object

The IUT contains, or can be made to contain, an Event Enrollment object that can generate EventNotifications with an Event\_Type of CHANGE\_OF\_RELIABILITY.

<b>BTL - 8.5.X9.7.1 - Internal Faults Take Precedence Over Monitored Object Faults</b>		
	<b>Test Conditionality</b>	If the IUT does not support an Event Enrollment object which can detect internal faults and monitor an object which detects faults, then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 8.5.X9.7.2 - Monitored Object Faults Take Precedence Over Fault Algorithms</b>		
	<b>Test Conditionality</b>	If the IUT does not support an Event Enrollment object which monitors an object which detects faults and which applies a fault algorithm, then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 8.5.X9.7.3 - Internal Faults Take Precedence Over Fault Algorithms</b>		
	<b>Test Conditionality</b>	If the IUT does not support an Event Enrollment object which can detect internal faults and which applies a fault algorithm, then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 8.5.X9.8 - CHANGE_OF_RELIABILITY of Event Enrollment Object, Monitored Object Fault</b>		
	<b>Test Conditionality</b>	If the IUT has no Event Enrollment object where the Monitored_Object that can transition to fault, this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 8.5.X9.9 - CHANGE OF RELIABILITY of Event Enrollment Object Fault</b>		

	<b>Test Conditionality</b>	If the IUT has no Event Enrollment object that detects an internal unreliable operational fault, this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 5.3.26 Implements the CHANGE\_OF\_LIFE\_SAFETY Algorithm

Contact the BTL for interim tests for this algorithm.

### 5.3.27 Implements the ACCESS\_EVENT Algorithm

Contact the BTL for interim tests for this algorithm.

### 5.3.28 Implements the CHANGE\_OF\_DISCRETE\_VALUE Algorithm

The IUT contains, or can be made to contain, an object that can generate EventNotifications with an Event\_Type of CHANGE\_OF\_DISCRETE\_VALUE.

<b>BTL - 8.4.X10 - CHANGE OF DISCRETE VALUE Test (ConfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test must be repeated for each datatype, BOOLEAN, Unsigned, Integer, Enumerated, CharacterString, Octet String, Date, Time, BACnetObjectIdentifier, or BACnetDateTime supported by the Object Property Reference property of the Event Enrollment object.
	<b>Testing Hints</b>	
<b>BTL - 8.5.X10 - CHANGE OF DISCRETE VALUE Test (UnconfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test must be repeated for each datatype, BOOLEAN, Unsigned, Integer, Enumerated, CharacterString, Octet String, Date, Time, BACnetObjectIdentifier, or BACnetDateTime supported by the Object Property Reference property of the Event Enrollment object.
	<b>Testing Hints</b>	

### 5.3.29 Implements the CHANGE\_OF\_TIMER Algorithm

Contact the BTL for interim tests for this algorithm.

### 5.3.30 Implements the CHANGE\_OF\_RELIABILITY - FAULT\_OUT\_OF\_RANGE Algorithm

The IUT contains, or can be made to contain, an Event Enrollment object that can generate EventNotifications with an Event\_Type of CHANGE\_OF\_RELIABILITY and supports the specified algorithm.

<b>BTL - 8.4.X9 - CHANGE_OF_RELIABILITY with the FAULT_OUT_OF_RANGE Algorithm (ConfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed
	<b>Test Directives</b>	This test shall be executed with an Event Enrollment object that is configured to monitor a property in a device other than the IUT. This test shall be executed for different Reliability transitions which are supported in the IUT such as No_Fault_Detected -Under_Range, Under_Range - No_Fault_Detected, No_Fault_Detected - OverRange, Over_Range - No_Fault_Detected, No_Fault_Detected - Under_Range, Under_Range - Over_Range, Over_Range - Under_Range and Under_Range - No_Fault_Detected..

		This test must be repeated once for each object type that is capable of generating event notifications with an Event_Type of CHANGE OF RELIABILITY with the FAULT OUT OF RANGE
	<b>Testing Hints</b>	
<b>BTL - 8.5.X9 - CHANGE_OF_RELIABILITY with the FAULT_OUT_OF_RANGE Algorithm (UnconfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed
	<b>Test Directives</b>	<p>This test shall be executed with an Event Enrollment object that is configured to monitor a property in a device other than the IUT.</p> <p>This test shall be executed for different Reliability transitions which are supported in the IUT such as No_Fault_Detected -Under_Range, Under_Range - No_Fault_Detected, No_Fault_Detected - OverRange, Over_Range - No_Fault_Detected, No_Fault_Detected - Under_Range, Under_Range - Over_Range, Over_Range - Under_Range and Under_Range - No_Fault_Detected.</p> <p>This test must be repeated once for each object type that is capable of generating event notifications with an Event_Type of CHANGE OF RELIABILITY with the FAULT OUT OF RANGE.</p>
	<b>Testing Hints</b>	

### 5.3.31 Implements a Proprietary Algorithm

The IUT contains, or can be made to contain, an Event Enrollment object that can generate ConfirmedEventNotifications and UnconfirmedEventNotifications with an Event\_Type of Extended.

<b>BTL - 8.4.X6 - Proprietary Algorithm Tests (ConfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test shall be executed with an Event Enrollment object that is configured to monitor a property in a device other than the IUT.
	<b>Testing Hints</b>	
<b>BTL - 8.5.X6 - Proprietary Algorithm Tests (UnconfirmedEventNotification)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test shall be executed with an Event Enrollment object that is configured to monitor a property in a device other than the IUT.
	<b>Testing Hints</b>	

### 5.3.32 Supports Event\_Message\_Texts Property

The IUT contains one or more objects that support the Event\_Message\_Texts property.

<b>BTL - 7.3.1.X4 - Event_Message_Texts Tests</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Repeat test once for each object type in the IUT that contains an Event_Message_Texts property.
	<b>Testing Hints</b>	

## 5.4 Alarm and Event Management - Acknowledge - A

### 5.4.1 Base Requirements

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

BTL - 8.1 - AcknowledgeAlarm Service Initiation Tests		
	Test Conditionality	Must be executed once to acknowledge a ConfirmedEventNotification, and again to acknowledge an UnconfirmedEventNotification.
	Test Directives	
	Testing Hints	
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	

### 5.4.2 Generates AcknowledgeAlarm Requests with Acknowledge Timestamps of the BACnetDateTime Form

The IUT can generate AcknowledgeAlarm requests with a Time of Acknowledgement of the BACnetDateTime form.

BTL - 8.1 - AcknowledgeAlarm Service Initiation Tests		
	Test Conditionality	Must be executed. Verify that the Time of Acknowledgement parameter is of the BACnetDateTime form.
	Test Directives	
	Testing Hints	

### 5.4.3 Generates AcknowledgeAlarm Requests with Acknowledge Timestamps of the Time Form

The IUT can generate AcknowledgeAlarm requests with a Time of Acknowledgement of the Time form.

BTL - 8.1 - AcknowledgeAlarm Service Initiation Tests		
	Test Conditionality	Must be executed. Verify that the Time of Acknowledgement parameter is of the Time form.
	Test Directives	
	Testing Hints	

### 5.4.4 Generates AcknowledgeAlarm Requests with Acknowledge Timestamps of the Sequence Number Form

The IUT can generate AcknowledgeAlarm requests with a Time of Acknowledgement of the Sequence Number form.

BTL - 8.1 - AcknowledgeAlarm Service Initiation Tests		
	Test Conditionality	Must be executed. Verify that the Time of Acknowledgement parameter is of the Sequence Number form.
	Test Directives	
	Testing Hints	

## 5.5 Alarm and Event Management - Acknowledge - B

### 5.5.1 Base Requirements

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

<b>BTL - 9.1.1.1 - Successful Alarm Acknowledgment of Confirmed Event Notifications Using the Time Form of the 'Time of Acknowledgment' Parameter</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.1.1.2 - Successful Alarm Acknowledgment of Confirmed Event Notifications using the Sequence Number Form of the 'Time of Acknowledgment' Parameter</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.1.1.3 - Successful Alarm Acknowledgment of Confirmed Event Notifications Using the Date Time Form of the 'Time of Acknowledgment' Parameter</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.1.1.4 - Successful Alarm Acknowledgment of Unconfirmed Event Notifications Using the Time Form of the 'Time of Acknowledgment' Parameter</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.1.1.5 - Successful Alarm Acknowledgment of Unconfirmed Event Notifications Using the Sequence Number Form of the 'Time of Acknowledgment' Parameter</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.1.1.6 - Successful Alarm Acknowledgment of Unconfirmed Event Notifications Using the Date Time Form of the 'Time of Acknowledgment' Parameter</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.1.1.8 - Successful Alarm Acknowledgment of Confirmed Event Notifications Using an Unknown 'Acknowledging Process Identifier' Parameter</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.1.1.9 - Successful Alarm Acknowledgment of Unconfirmed Event Notifications Using an Unknown 'Acknowledging Process Identifier' Parameter</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.1.2.1 - Unsuccessful Alarm Acknowledgment of Confirmed Event Notifications Because the 'Time Stamp' is Too Old</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	The test should be run multiple times varying the timestamp each time. When selecting a timestamp, the following are considered good choices:

		<p>A timestamp from a previous transition of the same type (if acknowledging a To-Normal transition, choose a timestamp from a previous To-Normal transition.)</p> <p>A timestamp that differs only in one of the time fields (hundredths, seconds, minutes, or hours.)</p>
<b>BTL - 9.1.2.3 - Unsuccessful Alarm Acknowledgment of Confirmed Event Notifications Because the 'Event Object Identifier' is Invalid</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute the test multiple times, with the 'Event Object Identifier' specifying an object that does not support or is not configured for alarming, again for an object which does not exist, and if the device can be configured with an object where Event_Detection_Enable is equal to FALSE, at least once for an object with Event_Detection_Enable equal to FALSE.
	<b>Testing Hints</b>	
<b>BTL - 9.1.2.4 - Unsuccessful Alarm Acknowledgment of Confirmed Event Notifications Because the 'Event State Acknowledged' is Invalid</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.1.2.5 - Unsuccessful Alarm Acknowledgment of Unconfirmed Event Notifications Because the 'Time Stamp' is Too Old</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.1.2.6 - Unsuccessful Alarm Acknowledgment of Unconfirmed Event Notifications Because the Referenced Object Does Not Exist</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.1.2.7 - Unsuccessful Alarm Acknowledgment of Unconfirmed Event Notifications Because the 'Event State Acknowledged' is Invalid</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.1.1.X3 - Successful Alarm Acknowledgment of Confirmed Event Notifications when 'To State' is either High-Limit or Low-Limit</b>		
	<b>Test Conditionality</b>	If the IUT supports LOW_LIMIT or HIGH_LIMIT transitions this test must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.1.1.10 - Successful Alarm Re-Acknowledgment of Confirmed Event Notifications</b>		
	<b>Test Conditionality</b>	If the IUT supports revision 7 or higher this test must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.1.1.11 - Successful Alarm Re-Acknowledgment of Unconfirmed Event Notifications</b>		
	<b>Test Conditionality</b>	If the IUT supports revision 7 or higher this test must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.11 - Acked Transitions Tests</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.X6.3 - Event_Algorithm_Inhibit Acknowledgement Test</b>		



	<b>Test Conditionality</b>	If the IUT has no object in which the Event_Algorithm_Inhibit property is present or has no object in which Event_Detection_Enable can be made TRUE, this test shall be skipped. If the IUT cannot be configured to contain any object with an unacknowledged event, then this test shall be skipped.
	<b>Test Directives</b>	The object types selected by the tester should include all variants that differ in the set of supported alarming properties, or the writability of any of those properties. At least one instance of each variant should be selected.
	<b>Testing Hints</b>	
<b>BTL - 9.1.X1 - Unsupported Acknowledgment Source Character Set AcknowledgeAlarm Test</b>		
	<b>Test Conditionality</b>	If the IUT supports all character sets, this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

---

## 5.6 Alarm and Event Management - Alarm Summary - A

---

### 5.6.1 Base Requirements

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

135.1-2013 - 8.6 - GetAlarmSummary Service Initiation Tests		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

## 5.7 Alarm and Event Management - Alarm Summary - B

### 5.7.1 Base Requirements

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

<b>135.1-2013 - 9.6.1 - Alarm Summaries with no Active Alarms</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.6.2 - Alarm Summaries with One Active Alarm</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.6.3 - Alarm Summaries with Multiple Active Alarms</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

## 5.8 Alarm and Event Management - Enrollment Summary - A

### 5.8.1 Base Requirements

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

135.1-2013 - 8.7.1 - Acknowledgment Filter		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 5.8.2 Generates Requests with an Enrollment Filter

The IUT can generate GetEnrollmentSummary requests that contain an Enrollment filter.

135.1-2013 - 8.7.2 - Enrollment Filter		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 5.8.3 Generates Requests with an Event State Filter

The IUT can generate GetEnrollmentSummary requests that contain an Event State filter.

135.1-2013 - 8.7.3 - Event State Filter		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 5.8.4 Generates Requests with an Event Type Filter

The IUT can generate GetEnrollmentSummary requests that contain an Event Type filter.

135.1-2013 - 8.7.4 - Event Type Filter		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 5.8.5 Generates Requests with a Priority Filter

The IUT can generate GetEnrollmentSummary requests that contain a Priority filter.

135.1-2013 - 8.7.5 - Priority Filter		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 5.8.6 Generates Requests with a Notification Class Filter

The IUT can generate GetEnrollmentSummary requests that contain a Notification Class filter.

135.1-2013 - 8.7.6 - Notification Class Filter		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 5.8.7 Generates Requests with Multiple Filters

The IUT can generate GetEnrollmentSummary requests that contain multiple filters.

135.1-2013 - 8.7.7 - Multiple Filters		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

## 5.9 Alarm and Event Management - Enrollment Summary - B

### 5.9.1 Base Requirements

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

<b>BTL - 9.7.1.1 - Enrollment Summary with Zero Summaries</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.7.1.2 - ACKED</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.7.1.3 - NOT ACKED</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.7.1.4 - All</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.7.2.1 - Enrollment Filter</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	It is advisable that the pre-tester: - repeat this test with a (BACnet Recipient Process, Process Identifier) pair that the IUT is not configured with. - repeat this test with a BACnetRecipient of the BACnetObjectIdentifier form, and again with a BACnetRecipient of the BACnetAddress form.
<b>135.1-2013 - 9.7.2.2 - Event State Filter</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.7.2.3 - Event Type Filter</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.7.2.4 - Priority Filter</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.7.2.5 - Notification Class Filter</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.7.2.6 - A Combination of Filters</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	Each enrollment filter is applied individually. Only those objects that match each and every filter in the GetEnrollmentSummary-Request shall be returned.

## BTL Test Plan

		Given that any combination of filters can be chosen by the BTL Tester, it is advisable that the pre-tester test numerous, if not all, combinations.
--	--	---

---

## 5.10 Alarm and Event Management - Information - A

---

### 5.10.1 Base Requirements

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

<b>135.1-2013 - 8.8.1 - Without Chaining</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 8.8.2 - With Chaining</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	



## 5.11 Alarm and Event Management - Information - B

### 5.11.1 Base Requirements

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

<b>135.1-2013 - 9.8.1 - Event Information with no Active Events</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.8.2 - Event Information with one Active Event</b>		
	<b>Test Conditionality</b>	If the IUT cannot be configured to contain any object with an active event, then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.8.3 - Event Information with Multiple Active Events</b>		
	<b>Test Conditionality</b>	Must be executed for devices that contain more than one object that can detect alarms. If the IUT cannot be configured to contain multiple active events which can be transmitted in a single GetEventInformation service response without 'More Events' = TRUE, then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.8.4 - Event Information Based on Event State</b>		
	<b>Test Conditionality</b>	If the IUT cannot be configured to contain any object with an active event, then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.8.6 - Chaining Test</b>		
	<b>Test Conditionality</b>	Must be executed for devices that contain more than one object that can detect alarms. If the IUT cannot be configured to contain enough active events to trigger chaining, then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.X6.2 - Event Algorithm Inhibit Summarization Test</b>		
	<b>Test Conditionality</b>	If the IUT has no object in which the Event_Algorithm_Inhibit property is present or has no object in which Event_Detection_Enable can be made TRUE, this test shall be skipped. If the IUT cannot be configured to contain any object with an active event, then this test shall be skipped.
	<b>Test Directives</b>	The object types selected by the tester should include all variants that differ in the set of supported alarming properties, or the writability of any of those properties. At least one instance of each variant should be selected.
	<b>Testing Hints</b>	

### 5.11.2 Supports AE-ACK-B

The IUT initiates EventNotifications with service parameter AckRequired = True.

<b>135.1-2013 - 9.8.5 - Event Information Based on Acknowledged Transitions</b>		
	<b>Test Conditionality</b>	If the IUT cannot be configured to contain any object with an unacknowledged event, then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

## 5.12 Alarm and Event Management - Event Log View - A

### 5.12.1 Base Requirements

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

BTL - 8.21.9 - Presents Log Records		
	<b>Test Conditionality</b>	Must be executed
	<b>Test Directives</b>	The information conveyed to the user shall at minimum be as described in the AE-ELV-A BIBB or as described in AE-VN-A. Any other information conveyed to the user shall be consistent with the data contained in the notification.
	<b>Testing Hints</b>	

### 5.12.2 Initiates ReadRange

The IUT is able to initiate the ReadRange service.

Verify Checklist		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support for Initiates ReadRange in the Checklist.
	<b>Testing Hints</b>	

### 5.12.3 Supports Reading Items by Time with a Positive Count

The IUT can initiate one or more ReadRange requests that specify the Time form and a Positive 'Count' that access a tester-specified portion of log records.

135.1-2013 - 8.21.8 - Reading a Range of Items Using Any Valid Range		
	<b>Test Conditionality</b>	Must be executed
	<b>Test Directives</b>	This test shall be executed with a reference server device claiming Protocol Revision 7 or higher.
	<b>Testing Hints</b>	The 'Range' value in Steps 2 and 4 must include at least one occurrence of the Time form of the ReadRange service, with a Positive 'Count'. The IUT shall be capable of presenting any portion of records selected by the tester. The set of records in this test shall contain at least one log-status entry.

### 5.12.4 Supports Reading Items by Time with a Negative Count

The IUT can initiate one or more ReadRange requests that specify the Time form and a Negative 'Count' that access a tester-specified portion of log records

135.1-2013 - 8.21.8 - Reading a Range of Items Using Any Valid Range		
	<b>Test Conditionality</b>	Must be executed
	<b>Test Directives</b>	This test shall be executed with a reference server device claiming Protocol Revision 7 or higher.
	<b>Testing Hints</b>	The 'Range' value in Steps 2 and 4 must include at least one occurrence of the Time form of the ReadRange service, with a Negative 'Count'. The IUT shall be capable of presenting any portion of records selected by the tester. The set of records in this test shall contain at least one log-status entry.

### 5.12.5 Supports Reading Items by Position with a Positive Count

The IUT can initiate one or more ReadRange requests that specify the Position form and a Positive 'Count' that access a tester-specified portion of log records.

135.1-2013 - 8.21.8 - Reading a Range of Items Using Any Valid Range		
	<b>Test Conditionality</b>	Must be executed
	<b>Test Directives</b>	This test shall be executed with a reference server device claiming Protocol Revision 7 or higher.
	<b>Testing Hints</b>	The 'Range' value in Steps 2 and 4 must include at least one occurrence of the Position form of the ReadRange service, with a Positive 'Count'. The IUT shall be capable of presenting any portion of records selected by the tester. The set of records in this test shall contain at least one log-status entry.

### 5.12.6 Supports Reading Items by Position with a Negative Count

The IUT can initiate one or more ReadRange requests that specify the Position form and a Negative 'Count' that access a tester-specified portion of log records.

135.1-2013 - 8.21.8 - Reading a Range of Items Using Any Valid Range		
	<b>Test Conditionality</b>	Must be executed
	<b>Test Directives</b>	This test shall be executed with a reference server device claiming Protocol Revision 7 or higher.
	<b>Testing Hints</b>	The 'Range' value in Steps 2 and 4 must include at least one occurrence of the Position form of the ReadRange service, with a Negative 'Count'. The IUT shall be capable of presenting any portion of records selected by the tester. The set of records in this test shall contain at least one log-status entry.

### 5.12.7 Supports Reading Items by Sequence Number with a Positive Count

The IUT can initiate one or more ReadRange requests that specify the Sequence Number form and a Positive 'Count' that access a tester-specified portion of log records.

135.1-2013 - 8.21.8 - Reading a Range of Items Using Any Valid Range		
	<b>Test Conditionality</b>	Must be executed
	<b>Test Directives</b>	This test shall be executed with a reference server device claiming Protocol Revision 7 or higher.
	<b>Testing Hints</b>	The 'Range' value in Steps 2 and 4 must include at least one occurrence of the Sequence Number form of the ReadRange service, with a Positive 'Count'. The IUT shall be capable of presenting any portion of records selected by the tester. The set of records in this test shall contain at least one log-status entry.

### 5.12.8 Supports Reading Items by Sequence Number with a Negative Count

The IUT can initiate one or more ReadRange requests that specify the Sequence Number form and a Negative 'Count' that access a tester-specified portion of log records.

135.1-2013 - 8.21.8 - Reading a Range of Items Using Any Valid Range		
	<b>Test Conditionality</b>	Must be executed
	<b>Test Directives</b>	This test shall be executed with a reference server device claiming Protocol Revision 7 or higher.
	<b>Testing Hints</b>	The 'Range' value in Steps 2 and 4 must include at least one occurrence of the Sequence Number form of the ReadRange service, with a Negative 'Count'. The IUT shall be capable of presenting any portion of records selected by the tester. The set of records in this test shall contain at least one log-status entry.

### 5.12.9 Supports Reading Items with no Range

The IUT can initiate one or more ReadRange requests that specify no range that access a tester-specified portion of log records.

135.1-2013 - 8.21.8 - Reading a Range of Items Using Any Valid Range		
	<b>Test Conditionality</b>	Must be executed
	<b>Test Directives</b>	This test shall be executed with a reference server device claiming Protocol Revision 7 or higher.
	<b>Testing Hints</b>	The 'Range' value in Steps 2 and 4 must specify no range The set of records in this test shall contain at least one log-status entry.

## 5.13 Alarm and Event Management - Event Log View and Modify - A

### 5.13.1 Base Requirements

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

135.1-2013 - 8.18.3 - Reading and Presenting Properties		
	<b>Test Conditionality</b>	Must be executed
	<b>Test Directives</b>	Repeat the test for each of the properties listed in the table in the BIBB definition.
	<b>Testing Hints</b>	
135.1-2013 - 8.22.4 - Accepting Input and Modifying Properties		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Repeat the test for each of the properties listed in the table in the BIBB definition.
	<b>Testing Hints</b>	Repeat the test for a variety of values that cover the range of values required by the BIBB.

### 5.13.2 Supports AE-ELV-A

The IUT shall support AE-ELV-A in order to display log data from Event Log objects to the user.

Verify Checklist		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support for AE-ELV-A.
	<b>Testing Hints</b>	

### 5.13.3 Supports DS-RP-A

The IUT shall support DS-RP-A in order to display Event Log property values to the user.

Verify Checklist		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support for DS-RP-A.
	<b>Testing Hints</b>	

### 5.13.4 Supports DS-WP-A

The IUT shall support DS-WP-A in order to configure Event Log property values modified by the user.

Verify Checklist		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support for DS-WP-A.
	<b>Testing Hints</b>	

## 5.14 Alarm and Event Management - Event Log - Internal - B

### 5.14.1 Base Requirements

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

<b>135.1-2013 - 9.21.1.1 - Reading All Items in the List</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.24.1 - Enable Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.24.6.1 - Stop When Full TRUE Test</b>		
	<b>Test Conditionality</b>	This shall be executed only if the property is configurable or equal to TRUE
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.24.6.2 - Stop When Full FALSE Test</b>		
	<b>Test Conditionality</b>	Only applicable if this property is writable.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.24.7 - Buffer Size Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.24.8 - Record Count Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.24.9 - Total Record Count Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.24.13 - Log-Status Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.25.1 - Internal Logging of Notifications</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	REPEAT for both Confirmed and UnconfirmedEventNotifications.
	<b>Testing Hints</b>	REPEAT for events with optional message text present, and with message text not present if the device supports both.

### 5.14.2 Supports all forms of ReadRange

The IUT can accept any of the ReadRange options and respond appropriately.

<b>135.1-2013 - 9.21.1.2 - Reading Items by Position with Positive Count</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

<b>135.1-2013 - 9.21.1.3 - Reading Items by Position with Negative Count</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.21.1.4 - Reading Items by Time</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.21.1.4.1 - Reading Items by Time with Negative Count</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.21.1.9 - Reading Items by Sequence with Positive Count</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.21.1.10 - Reading Items by Sequence with Negative Count</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.21.1.7 - Reading a Range of Items that do not Exist (by Sequence)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.21.1.8 - Reading a Range of Items that do not Exist (by Time)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 5.14.3Executes ReadRange

The IUT is able to execute the ReadRange service.

<b>Verify Checklist</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support for Executes ReadRange in the Checklist.
	<b>Testing Hints</b>	

### 5.14.4Supports logging of ACK\_NOTIFICATION

The IUT can be made to log event notifications with AckRequired, From State, and Event Values absent.

<b>BTL - 7.3.2.27 - Internal Logging of ACK_NOTIFICATION</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 5.14.5Supports Start\_Time and Stop\_Time Properties

The IUT can be made to start and stop logging using these properties.

These properties are required to be present and writable in Event Log objects, if either is present.

<b>135.1-2013 - 7.3.2.24.2 - Start Time Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.24.3 - Stop Time Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.2.X6 - DateTime Non-Pattern Properties Test</b>		
	<b>Test Conditionality</b>	Must be executed if the IUT claims Protocol_Revision 11 or greater.
	<b>Test Directives</b>	Apply to the Start_Time and again to the Stop_Time properties in an Event Log object.
	<b>Testing Hints</b>	
<b>BTL - 9.23.2.X11 - DateTime Non-Pattern Properties Test using WritePropertyMultiple Service</b>		
	<b>Test Conditionality</b>	This test shall only be applied to devices claiming Protocol_Revision 11 or higher and which supports execution of WritePropertyMultiple.
	<b>Test Directives</b>	Apply to the Start_Time and again to the Stop_Time properties in an Event Log object.
	<b>Testing Hints</b>	



## 5.15 Alarm and Event Management - Event Log - External - B

### 5.15.1 Base Requirements

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

<b>135.1-2013 - 9.21.1.1 - Reading All Items in the List</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.24.1 - Enable Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.24.6.1 - Stop When Full TRUE Test</b>		
	<b>Test Conditionality</b>	Only applicable if this property is writable.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.24.6.2 - Stop When Full FALSE Test</b>		
	<b>Test Conditionality</b>	Only applicable if this property is writable.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.24.7 - Buffer Size Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.24.8 - Record Count Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.24.9 - Total Record Count Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.24.13 - Log-Status Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 5.15.2 Supports all forms of ReadRange

The IUT can accept any of the ReadRange options and respond appropriately.

<b>135.1-2013 - 9.21.1.2 - Reading Items by Position with Positive Count</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.21.1.3 - Reading Items by Position with Negative Count</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

<b>135.1-2013 - 9.21.1.4 - Reading Items by Time</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.21.1.4.1 - Reading Items by Time with Negative Count</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.21.1.9 - Reading Items by Sequence with Positive Count</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.21.1.10 - Reading Items by Sequence with Negative Count</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.21.1.7 - Reading a Range of Items that do not Exist (by Sequence)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.21.1.8 - Reading a Range of Items that do not Exist (by Time)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 5.15.3 Executes ReadRange

The IUT is able to execute the ReadRange service.

<b>Verify Checklist</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support for Executes ReadRange in the Checklist.
	<b>Testing Hints</b>	

### 5.15.4 Supports Logging all Notifications

The IUT can be made to log all Notifications. After a notification is sent to the device, it is required to be present in an Event Log object, and to represent the Notification completely and accurately.

<b>BTL - 7.3.2.26 - Remote Logging of Notifications</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	REPEAT for both Confirmed and UnconfirmedEventNotifications. REPEAT for events which contain the optional message text parameter and for those which don't. REPEAT for all BACnetEventTypes up to the Protocol_Revision claimed by the IUT. TRANSMIT EventNotifications of a size that the IUT is capable of logging.
	<b>Testing Hints</b>	This does include CHANGE_OF_LIFE_SAFETY, BUFFER_READY, and complex event types.
<b>BTL - 7.3.2.28 - Remote Logging of ACK NOTIFICATIONS</b>		
	<b>Test Conditionality</b>	Must be executed.

	<b>Test Directives</b>	TRANSMIT EventNotifications of a size that the IUT is capable of logging.
	<b>Testing Hints</b>	

### 5.15.5 Supports Start\_Time and Stop\_Time Properties

The IUT can be made to start and stop logging using these properties.

These properties are required to be present and writable in Event Log objects, if either is present.

<b>135.1-2013 - 7.3.2.24.2 - Start Time Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.24.3 - Stop Time Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

## 5.16 Alarm and Event Management - View Notifications - A

### 5.16.1 Base Requirements

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

<b>BTL - 9.4.5 - ConfirmedEventNotification Simple Presentation</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Repeat the test for each of the standard event types, including EXTENDED if IUT claims Protocol_Revision 13 or higher, and each of the transitions defined for those event types. Execute at least once with a Message_Text 32 or more characters in length.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.5.1 - UnconfirmedEventNotification Simple Presentation</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	Repeat the test for each of the standard event types, including EXTENDED if IUT claims Protocol_Revision 13 or higher, and each of the transitions defined for those event types. Execute at least once with a Message_Text 32 or more characters in length.

### 5.16.2 Supports AE-N-A

The IUT shall support AE-N-A in order to receive notification for presentation.

<b>Verify Checklist</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support for AE-N-A.
	<b>Testing Hints</b>	

## 5.17 Alarm and Event Management - View Modify - A

### 5.17.1 Base Requirements

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

135.1-2013 - 8.18.3 - Reading and Presenting Properties		
	<b>Test Conditionality</b>	Must be executed. If the IUT also claims support for AE-AVM-A, this test may be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	Repeat for each standard property, in each standard object type, which represent parameters to an event or fault algorithm excluding those for algorithms excluded by the BIBB definition.
135.1-2013 - 8.22.4 - Accepting Input and Modifying Properties		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	Repeat for each standard property, in each standard object type, which represent parameters to an event or fault algorithm excluding those for algorithms excluded by the BIBB definition. Repeat the test for a variety of values that cover the range of values required by the BIBB.

### 5.17.2 Supports DS-RP-A

The IUT shall support DS-RP-A in order to receive alarm parameters for presentation to the user.

Verify Checklist		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support for DS-RP-A.
	<b>Testing Hints</b>	

### 5.17.3 Supports DS-WP-A

The IUT shall support DS-WP-A in order to update alarm parameters modified by the user.

Verify Checklist		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support for DS-WP-A.
	<b>Testing Hints</b>	

## 5.18 Alarm and Event Management - Advanced View Notifications - A

### 5.18.1 Base Requirements

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

<b>BTL - 9.4.6 - ConfirmedEventNotification Full Presentation</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Repeat the test for each of the standard event types, including EXTENDED if IUT claims Protocol_Revision 13 or higher, and each of the transitions defined for those event types. For notifications using the CHOICE format of EXTENDED, the presentation must show all the fields which were in the notification. Execute at least once with a Message_Text 256 or more characters in length.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.5.2 - UnconfirmedEventNotification Full Presentation</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	Repeat the test for each of the standard event types, including EXTENDED if IUT claims Protocol_Revision 13 or higher, and each of the transitions defined for those event types. For notifications using the CHOICE format of EXTENDED, the presentation must show all the fields which were in the notification. Execute at least once with a Message_Text 256 or more characters in length.

### 5.18.2 Supports AE-VN-A

The IUT shall support AE-VN-A in order to receive notification for presentation.

<b>Verify Checklist</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support for AE-VN-A.
	<b>Testing Hints</b>	

## 5.19 Alarm and Event Management - Advanced View Modify - A

### 5.19.1 Base Requirements

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

135.1-2013 - 8.18.3 - Reading and Presenting Properties		
	Test Conditionality	Must be executed. Note: If the IUT also claims support for AE-AVM-A, this test may be skipped.
	Test Directives	
	Testing Hints	Repeat for each standard property, in each standard object type, which represent parameters to an event or fault algorithm excluding those for algorithms excluded by the BIBB definition. Repeat for each property listed in the tables in the BIBB, in each standard object type, excluding those for algorithms excluded by the BIBB definition.
135.1-2013 - 8.22.4 - Accepting Input and Modifying Properties		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	Repeat for each standard property, in each standard object type, which represents parameters to an event or fault algorithm excluding those for algorithms excluded by the BIBB definition. Repeat for each property listed in the tables in the BIBB, in each standard object type, excluding those for algorithms excluded by the BIBB definition. Repeat the test for a variety of values that cover the range of values required by the BIBB.

### 5.19.2 Supports DS-RP-A

The IUT shall support DS-RP-A in order to receive alarm parameters for presentation to the user.

Verify Checklist		
	Test Conditionality	Must be executed.
	Test Directives	Verify that the IUT claims support for DS-RP-A.
	Testing Hints	

### 5.19.3 Supports DS-WP-A

The IUT shall support DS-WP-A in order to update alarm parameters modified by the user.

Verify Checklist		
	Test Conditionality	Must be executed.
	Test Directives	Verify that the IUT claims support for DS-WP-A.
	Testing Hints	

### 5.19.4 Supports DM-OCD-A

The IUT shall support DS-OCD-A in order to allow the user to create Event Enrollment and Notification Class objects.

Verify Checklist		
	Test Conditionality	Must be executed.
	Test Directives	Verify that the IUT claims support for DM-OCD-A and that it claims the ability to create and delete Event Enrollment, Notification Class, and Notification Forwarder objects.
	Testing Hints	

## 5.20 Alarm and Event Management - Alarm Summary View - A

### 5.20.1 Base Requirements

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

135.1-2013 - 8.6.2 - Updating Alarm Summary Information with GetAlarmSummary		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.8.3 - Updating Alarm Summary Information with GetEventInformation Without Chaining		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.8.4 - Updating Alarm Summary Information with GetEventInformation With Chaining		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 5.20.2 Supports Initiation of GetEventInformation Service

The IUT shall support GetEventInformation service in order to receive alarm parameters for presentation to the user.

135.1-2013 - 8.8.1 - Without Chaining		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.8.2 - With Chaining		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 5.20.3 Supports Initiation of GetAlarmSummary Service

The IUT shall support AE-ASUM-A in order to update alarm parameters modified by the user.

135.1-2013 - 8.6 - GetAlarmSummary Service Initiation Tests		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 5.20.4 Supports Initiation of GetEnrollmentSummary Service

The IUT shall support GetEnrollmentSummary service in order to update alarm parameters modified by the user.

135.1-2013 - 8.7.1 - Acknowledgment Filter		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	



### 5.20.5 Generates GetEnrollmentSummary Requests with an Enrollment Filter

The IUT can generate GetEnrollmentSummary requests that contain an Enrollment filter.

135.1-2013 - 8.7.2 - Enrollment Filter		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 5.20.6 Generates GetEnrollmentSummary Requests with an Event State Filter

The IUT can generate GetEnrollmentSummary requests that contain an Event State filter.

135.1-2013 - 8.7.3 - Event State Filter		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 5.20.7 Generates GetEnrollmentSummary Requests with an Event Type Filter

The IUT can generate GetEnrollmentSummary requests that contain an Event Type filter.

135.1-2013 - 8.7.4 - Event Type Filter		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 5.20.8 Generates GetEnrollmentSummary Requests with a Priority Filter

The IUT can generate GetEnrollmentSummary requests that contain a Priority filter.

135.1-2013 - 8.7.5 - Priority Filter		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 5.20.9 Generates GetEnrollmentSummary Requests with a Notification Class Filter

The IUT can generate GetEnrollmentSummary requests that contain a Notification Class filter.

135.1-2013 - 8.7.6 - Notification Class Filter		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 5.20.10 Generates GetEnrollmentSummary Requests with Multiple Filters

The IUT can generate GetEnrollmentSummary requests that contain multiple filters.

135.1-2013 - 8.7.7 - Multiple Filters		
	Test Conditionality	Must be executed.

## BTL Test Plan

	<b>Test Directives</b>	
	<b>Testing Hints</b>	

## 5.21 Alarm and Event Management - LifeSafety - A

### 5.21.1 Base Requirements

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

BTL - 9.4.X1 - Unsupported Message Text Character Set ConfirmedEventNotification Test		
	Test Conditionality	If the IUT supports all character sets, this test shall be skipped.
	Test Directives	
	Testing Hints	
BTL - 9.5.X1 - Unsupported Message Text Character Set UnconfirmedEventNotification Test		
	Test Conditionality	If the IUT supports all character sets, this test shall be skipped.
	Test Directives	
	Testing Hints	

### 5.21.2 Initiates LifeSafetyOperation Requests

135.1-2013 - 8.9.1 - LifeSafetyOperation Service Initiation Tests to an Object		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.9.2 - LifeSafetyOperation Service Initiation Tests to all Objects in a Device		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 5.21.3 Executes ConfirmedEventNotifications

The IUT is capable of executing ConfirmedEventNotifications. This functionality will be covered by the testing of the individual algorithms.

No Specific Test		
	Test Conditionality	Must be executed.
	Test Directives	Verify that the IUT's EPICS claims that it supports the ConfirmedEventNotification service.
	Testing Hints	

### 5.21.4 Executes UnconfirmedEventNotifications

The IUT is capable of executing UnconfirmedEventNotifications. There are currently no tests defined for this functional item.

No Specific Test		
	Test Conditionality	Must be executed.
	Test Directives	Verify that the IUT's EPICS claims that it supports the UnconfirmedEventNotification service.
	Testing Hints	

### 5.21.5 Processes Intrinsically Generated Notifications

The IUT is capable of executing ConfirmedEventNotifications that reference an object type other than Event Enrollment.

<b>135.1-2013 - 9.4.1 - ConfirmedEventNotification Using the Time Form of the 'Timestamp' Parameter and Conveying a Text Message,</b> <b>135.1-2013 - 9.4.2 - ConfirmedEventNotification Using the DateTime Form of the 'Timestamp' Parameter and no Text Message, or</b> <b>135.1-2013 - 9.4.3 - ConfirmedEventNotification Using the Sequence Number Form of the 'Timestamp' Parameter and no Text Message</b>		
	<b>Test Conditionality</b>	At least one of the tests must be executed with the Event Object Identifier referencing a BACnet object other than an Event Enrollment object.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 5.21.6 Processes Algorithmically Generated Notifications

The IUT is capable of executing ConfirmedEventNotifications that reference an Event Enrollment object.

<b>135.1-2013 - 9.4.1 - ConfirmedEventNotification Using the Time Form of the 'Timestamp' Parameter and Conveying a Text Message,</b> <b>135.1-2013 - 9.4.2 - ConfirmedEventNotification Using the DateTime Form of the 'Timestamp' Parameter and no Text Message, or</b> <b>135.1-2013 - 9.4.3 - ConfirmedEventNotification Using the Sequence Number Form of the 'Timestamp' Parameter and no Text Message</b>		
	<b>Test Conditionality</b>	At least one of the tests must be executed with the Event Object Identifier referencing an Event Enrollment object.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 5.21.7 Processes Event Notifications with Timestamps of the BACnetDateTime Form

The IUT is capable of executing ConfirmedEventNotifications that contain a timestamp of the BACnetDateTime form.

<b>135.1-2013 - 9.4.2 - ConfirmedEventNotification Using the DateTime Form of the 'Timestamp' Parameter and no Text Message</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 5.21.8 Processes Event Notifications with Timestamps of the Time Form

The IUT is capable of executing ConfirmedEventNotifications that contain a timestamp of the Time form.

<b>135.1-2013 - 9.4.1 - ConfirmedEventNotification Using the Time Form of the 'Timestamp' Parameter and Conveying a Text Message</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 5.21.9 Processes Event Notifications with Timestamps of the Sequence Number Form

The IUT is capable of executing ConfirmedEventNotifications that contain a timestamp of the Sequence Number form.

<b>135.1-2013 - 9.4.3 - ConfirmedEventNotification Using the Sequence Number Form of the 'Timestamp' Parameter and no Text Message</b>		
--	--	--

	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 5.21.10 Supports AE-ACK-A

The IUT must support AE-ACK-A if it claims support for AE-LS-A.

Verify Checklist		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support for AE-ACK-A in the Checklist.
	<b>Testing Hints</b>	

### 5.21.11 Supports AE-INFO-A

The IUT must support AE-INFO-A if it claims support for AE-LS-A.

Verify Checklist		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support for AE-INFO-A in the Checklist.
	<b>Testing Hints</b>	

### 5.21.12 Supports AE-AS-A

The IUT must support AE-AS-A if it claims support for AE-LS-A.

Verify Checklist		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support for AE-AS-A in the Checklist.
	<b>Testing Hints</b>	

## 5.22 Alarm and Event Management - LifeSafety - B

### 5.22.1 Base Requirements

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

<b>BTL - 7.3.1.10.1 - Event Enable Tests for TO_OFFNORMAL and TO_NORMAL</b>		
	<b>Test Conditionality</b>	If the IUT cannot be configured to meet the configuration requirements then this test shall be skipped.
	<b>Test Directives</b>	If Event Enrollment objects are supported, ensure this functionality is tested on Event Enrollment objects.
	<b>Testing Hints</b>	The BTL will apply this to a single object. The pretester should apply it to all objects that support alarm generation.
<b>135.1-2013 - 7.3.1.12 - Notify Type Test</b>		
	<b>Test Conditionality</b>	If the IUT cannot be configured to meet the 135.1-2013 configuration requirements then this test shall be skipped.
	<b>Test Directives</b>	If Event Enrollment objects are supported, ensure this functionality is tested on Event Enrollment objects.
	<b>Testing Hints</b>	
<b>135.1-2013 - 8.4 - ConfirmedEventNotification Service Initiation Tests</b>		
	<b>Test Conditionality</b>	Must be executed. Any of the 8.4 tests can be used to ensure that the IUT properly generates ConfirmedEventNotification requests. The specific tests that can be executed are detailed under the test cases for the specific algorithms. As long as one of the tests is executed using ConfirmedEventNotifications, then this test case shall be satisfied.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 8.5 - UnconfirmedEventNotification Service Initiation Tests</b>		
	<b>Test Conditionality</b>	Must be executed. Any of the 8.5 tests can be used to ensure that the IUT properly generates UnconfirmedEventNotification requests. The specific tests that can be executed are detailed under the test cases for the specific algorithms. As long as one of the tests is executed using UnconfirmedEventNotifications, then this test case shall be satisfied.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.X9.1 - Event Detection Enable Inhibits Event Generation</b>		
	<b>Test Conditionality</b>	If Protocol Revision < 13, then this test shall be skipped.
	<b>Test Directives</b>	The objects selected by the tester should include all variants that differ in the set of supported alarming properties, or the writability of any of those properties. At least one instance of each variant should be selected.
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.X9.2 - Event Detection Enable Inhibits FAULT</b>		
	<b>Test Conditionality</b>	If Protocol Revision < 13, then this test shall be skipped.
	<b>Test Directives</b>	The objects selected by the tester should include all variants that differ in the set of supported alarming properties, or the writability of any of those properties. At least one instance of each variant should be selected.
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.X6.1 - Event Algorithm Inhibit Test</b>		
	<b>Test Conditionality</b>	If the IUT has no object in which the Event_Algorithm_Inhibit property is present and does not support the Event_Algorithm_Inhibit_Ref property, or has no object in which Event_Detection_Enable can be

		made TRUE, this test shall be skipped. If the IUT cannot be configured to contain any object capable of an event transition, then this test shall be skipped.
	<b>Test Directives</b>	The object types selected by the tester should include all variants that differ in the set of supported alarming properties, or the writability of any of those properties. At least one instance of each variant should be selected.
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.X7.1 - Event_Algorithm_Inhibit_Ref Test</b>		
	<b>Test Conditionality</b>	If the IUT has no object in which the Event_Algorithm_Inhibit_Ref property is present or has no object in which Event_Detection_Enable can be made TRUE, this test shall be skipped.
	<b>Test Directives</b>	The object types selected by the tester should include all variants that differ in the set of supported alarming properties, or the writability of any of those properties. At least one instance of each variant should be selected.
	<b>Testing Hints</b>	
<b>BTL - 7.3.1.X7.2 - Event_Algorithm_Inhibit Writable Test</b>		
	<b>Test Conditionality</b>	If the IUT has no object in which the Event_Algorithm_Inhibit_Ref property is absent or can be made uninitialized or has no object in which Event_Detection_Enable can be made TRUE, this test shall be skipped.
	<b>Test Directives</b>	The object types selected by the tester should include all variants that differ in the set of supported alarming properties, or the writability of any of those properties. At least one instance of each variant should be selected.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.9.1 - Reset Single Object Execution Tests</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.9.2 - Reset Multiple Object Execution Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.9.3 - Silencing Execution Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 5.22.2 Supports the Notification Class Object

The IUT supports the Notification Class object in order to send notifications.

<b>Verify Checklist</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support for the Notification Class Object in the Checklist.
	<b>Testing Hints</b>	

### 5.22.3 Supports AE-INFO-B

The IUT must support AE-INFO-B if it claims support for AE-LS-B.

<b>Verify Checklist</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support for AE-INFO-B in the Checklist.

	Testing Hints	
--	---------------	--

### 5.22.4 Implements Intrinsic Alarming

The IUT contains, or can be made to contain, an object other than an Event Enrollment object that can generate ConfirmedEventNotifications and UnconfirmedEventNotifications.

Verify Checklist		
	Test Conditionality	Must be executed.
	Test Directives	This functionality will be tested by the clause 8.4 or 8.5 algorithm tests listed later in this section.
	Testing Hints	

### 5.22.5 Supports the Event Enrollment Object

The IUT contains, or can be made to contain an Event Enrollment object that can generate ConfirmedEventNotifications and UnconfirmedEventNotifications.

Verify Checklist		
	Test Conditionality	Must be executed.
	Test Directives	Ensure this functionality is tested on Event Enrollment objects by the clause 8.4 or 8.5 algorithm tests listed later in this section.
	Testing Hints	

### 5.22.6 Supports the CHANGE\_OF\_LIFE\_SAFETY Algorithm in Event\_Parameters

The IUT contains, or can be made to contain an Event Enrollment object that can generate CHANGE\_OF\_LIFE\_SAFETY ConfirmedEventNotifications and UnconfirmedEventNotifications.

Verify Checklist		
	Test Conditionality	Must be executed.
	Test Directives	Ensure this functionality is tested on Event Enrollment objects by the clause 8.4.8 or 8.5.8 tests in that section.
	Testing Hints	

### 5.22.7 Supports AE-ACK-B

The IUT initiates EventNotifications with service parameter AckRequired = True.

135.1-2013 - 9.8.5 - Event Information Based on Acknowledged Transitions		
	Test Conditionality	If the IUT cannot be configured to contain any object with an unacknowledged event, then this test shall be skipped.
	Test Directives	
	Testing Hints	
Verify Checklist		
	Test Conditionality	Must be executed.
	Test Directives	If the IUT cannot be configured to contain any object with an unacknowledged event, then this test shall be skipped.
	Testing Hints	

### 5.22.8 Generates Event Notifications with Timestamps of the BACnetDateTime Form

The IUT generates, or can be made to generate, ConfirmedEventNotifications with the Time Stamp parameter taking the BACnetDateTime form.



<b>135.1-2013 - 8.4 - ConfirmedEventNotification Service Initiation Tests</b>		
	<b>Test Conditionality</b>	If the IUT supports AE-N-I-B, these tests may be skipped. Any of the 8.4 tests can be used to ensure that the IUT properly generates ConfirmedEventNotification requests using the BACnetDateTime form. The specific tests that can be executed are detailed under the test cases for the specific algorithms. As long as one of the tests is executed using ConfirmedEventNotifications and the notification that is generated contains a timestamp of the BACnetDateTime form, then this test case shall be satisfied.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 5.22.9 Generates Event Notifications with Timestamps of the Time Form

The IUT generates, or can be made to generate, ConfirmedEventNotifications with the Time Stamp parameter taking the Time form.

<b>135.1-2013 - 8.4 - ConfirmedEventNotification Service Initiation Tests</b>		
	<b>Test Conditionality</b>	If the IUT supports AE-N-I-B, these tests may be skipped. Any of the 8.4 tests can be used to ensure that the IUT properly generates ConfirmedEventNotification requests using the Time form. The specific tests that can be executed are detailed under the test cases for the specific algorithms. As long as one of the tests is executed using ConfirmedEventNotifications and the notification that is generated contains a timestamp of the Time form, then this test case shall be satisfied.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 5.22.10 Generates Event Notifications with Timestamps of the Sequence Number Form

The IUT generates, or can be made to generate, ConfirmedEventNotifications with the Time Stamp parameter taking the Sequence Number form.

<b>135.1-2013 - 8.4 - ConfirmedEventNotification Service Initiation Tests</b>		
	<b>Test Conditionality</b>	If the IUT supports AE-N-I-B, these tests may be skipped. Any of the 8.4 tests can be used to ensure that the IUT properly generates ConfirmedEventNotification requests using the Sequence Number form. The specific tests that can be executed are detailed under the test cases for the specific algorithms. As long as one of the tests is executed using ConfirmedEventNotifications and the notification that is generated contains a timestamp of the Sequence Number form, then this test case shall be satisfied.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 5.22.11 Mode Transition Tests when Event State is Maintained

<b>135.1-2013 - 8.4.8.7 - Mode Transition Tests when Event State is Maintained</b>		
	<b>Test Conditionality</b>	Must be executed
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 5.22.12 Supports Event\_Message\_Texts Property

The IUT contains one or more objects that support the Event\_Message\_Texts property.

<b>BTL - 7.3.1.X4 - Event_Message_Texts Tests</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Repeat test once for each object type in the IUT that contains an Event_Message_Texts property.
	<b>Testing Hints</b>	

### 5.22.13 Supports Event\_Message\_Texts\_Config Property

The IUT contains one or more objects that support the Event\_Message\_Texts\_Config property.

<b>BTL - 7.3.1.X5 - Event_Message_Texts_Config Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Repeat for each supported transition type (TO_OFFNORMAL, TO_FAULT, TO_NORMAL). Different objects may be selected for different transitions.
	<b>Testing Hints</b>	

## 5.23 Alarm and Event Management - Notification Forwarder - B

### 5.23.1 Base Requirements

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

<b>135.1-2013q - 7.3.2.30.2 - Recipient_List Forwarding Test</b>		
	Test Conditionality	Must be executed with Local_Forwarding_Only = FALSE.
	Test Directives	
	Testing Hints	
<b>135.1-2013q - 7.3.2.30.3 - Subscribed_Recipients Forwarding Test</b>		
	Test Conditionality	Must be executed with Local_Forwarding_Only = FALSE.
	Test Directives	Perform this using base Setup 2.
	Testing Hints	
<b>135.1-2013q - 7.3.2.30.7.1 - Destination Date Filtering Test</b>		
	Test Conditionality	Must be executed.
	Test Directives	Perform this using base Setup 1.
	Testing Hints	
<b>135.1-2013q - 7.3.2.30.7.2 - Destination Time Filtering Test</b>		
	Test Conditionality	Must be executed.
	Test Directives	Perform this using base Setup 1.
	Testing Hints	
<b>135.1-2013q - 7.3.2.30.7.3 - Process Identifier Test</b>		
	Test Conditionality	Must be executed.
	Test Directives	Perform this using base Setup 1.
	Testing Hints	
<b>135.1-2013q - 7.3.2.30.7.4 - Destination Transition Filtering Test</b>		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
<b>135.1-2013q - 7.3.2.30.11.2 - Forwards Locally and Remotely When False</b>		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
<b>135.1-2013q - 7.3.2.30.5 - Character Encoding Test</b>		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
<b>135.1-2013q - 7.3.2.30.12.1 - Local Broadcast To Receiving Port Restriction Test</b>		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
<b>135.1-2013q - 7.3.2.30.12.2 - Globally Broadcast Event Notification Received Restriction Test</b>		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
<b>135.1-2013q - 7.3.2.30.12.3 - Forwarding As Global Broadcast Restriction Test</b>		
	Test Conditionality	Must be executed.
	Test Directives	Perform this using base Setup 1.
	Testing Hints	
<b>135.1-2013q - 7.3.2.30.12.4 - Directed Broadcast Received Forwarding To BACnetAddress Restriction Test</b>		
	Test Conditionality	Must be executed.
	Test Directives	Perform this using base Setup 1.

	<b>Testing Hints</b>	
<b>135.1-2013q - 7.3.2.30.12.5. - Directed Broadcast Received Forwarding To Object Identifier Restriction Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Perform this using base Setup 1.
	<b>Testing Hints</b>	
<b>135.1-2013q - 7.3.2.30.12.6 - Port Restriction Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</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.

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

## 5.24 Alarm and Event Management - Notification Forwarder - I - B

### 5.24.1 Base Requirements

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

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

	Testing Hints	
--	---------------	--

### 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	

## 5.25 Alarm and Event Management - Configurable Recipient Lists - B

### 5.25.1 Base Requirements

There are no base requirements tests for this section. Existing tests for Notification Class object ensure Recipient\_List supports writing all forms.

### 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		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	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.
	<b>Testing Hints</b>	

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

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

### 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		
	<b>Test Conditionality</b>	Must be executed.

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

### 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.

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



## 5.26 Alarm and Event Management - Temporary Event Subscription - A

### 5.26.1 Base Requirements

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

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

### 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		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify DM-LM-A for the Subscribed Recipients property is supported.
	<b>Testing Hints</b>	

---

## **5.27 Alarm and Event Management - Life Safety View Notifications - A**

---

### **5.27.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

---

## **5.28 Alarm and Event Management - Life Safety Advanced View Notifications - A**

---

### **5.28.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

---

## **5.29 Alarm and Event Management - Life Safety View and Modify - A**

---

### **5.29.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

---

## **5.30 Alarm and Event Management - Life Safety Advanced View and Modify - A**

---

### **5.30.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

---

## **5.31 Alarm and Event Management - Access Control - A**

---

### **5.31.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

---

## **5.32 Alarm and Event Management - Access Control - B**

---

### **5.32.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

---

## **5.33 Alarm and Event Management - Access Control Advanced View Notifications - A**

---

### **5.33.1 Base Requirements**

Contact BTL for interim tests for this BIBB.



---

## **5.34 Alarm and Event Management - Access Control View and Modify - A**

---

### **5.34.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

---

## **5.35 Alarm and Event Management - Access Control Advanced View and Modify - A**

---

### **5.35.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

---

## **5.36 Alarm and Event Management - Elevator View Notifications - A**

---

### **5.36.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

---

## **5.37 Alarm and Event Management - Elevator Advanced View Notifications - A**

---

### **5.37.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

---

## **5.38 Alarm and Event Management - Elevator View and Modify - A**

---

### **5.38.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

---

## **5.39 Alarm and Event Management - Elevator Advanced View and Modify - A**

---

### **5.39.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

---

## **6 Scheduling BIBBs**

---

## 6.1 Scheduling - Advanced View Modify - A

### 6.1.1 Base Requirements

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

135.1-2013 - 8.18.3 - Reading and Presenting Properties		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	Verify for Weekly_Schedule, Exception_Schedule, Effective_Period , Date_List, Schedule_Default, List_Of_Object_Property_References, Priority_For_Writing, Out_Of_Service properties. See Addendum L for requirements details. The reference schedule used during this test should include an Exception_Schedule that contains 255 entries and contain 12 BACnetTimeValue tuples per entry. The reference schedule should also contain a Weekly_Schedule which contains 6 BACnetTimeValue tuples per day. The Calendar Date_List used in this test should contain 32 calendar entries.
135.1-2013 - 8.22.4 - Accepting Input and Modifying Properties		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	Verify for Weekly_Schedule, Exception_Schedule, Effective_Period , Date_List , Schedule_Default, List_Of_Object_Property_References, Priority_For_Writing, and Out_Of_Service properties. See Addendum L for requirements details. The reference schedule used during this test should include an Exception_Schedule that contains 255 entries and contain 12 BACnetTimeValue tuples per entry. The reference schedule should also contain a Weekly_Schedule which contains 6 BACnetTimeValue tuples per day. The Calendar Date_List used in this test should contain 32 calendar entries.

### 6.1.2 Supports SCHED-VM-A

The IUT supports SCHED-VM-A.

Verify Checklist		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify the IUT claims support for the BIBB SCHED-VM-A.
	<b>Testing Hints</b>	

### 6.1.3 Supports DM-OCD-A

The IUT supports DM-OCD-A for the Calendar and Schedule objects.

Verify Checklist		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify the IUT claims support for the BIBB DM-OCD-A and that the Calendar and Schedule objects can be created and deleted.
	<b>Testing Hints</b>	



### 6.1.4 Is able to Read a Schedule Object that is Self-inconsistent with regard to the Scheduled Datatype, and Modify it to be Consistent

The IUT is capable of modifying schedule objects that are self-inconsistent with regards to the scheduled datatype.

135.1-2013 - 13.10.6 - Modify a Self-inconsistent Schedule to be Consistent		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	The IUT is not required to present the self-inconsistent schedule, only that the IUT write out a consistent schedule after the modification. Ideally, the IUT should involve the user in the process of modifying the schedule to be consistent, presenting the full data of the schedule and allowing the user to edit the data to correct the inconsistencies. If the IUT modifies the schedule automatically to make it consistent, it should at least notify the user that the schedule has been modified. It is not acceptable for the IUT to modify the schedule without any indication to the user that this was done.

### 6.1.5 Is Able to Change the Datatype that a Schedule Object Schedules

The IUT supports the ability for the user to change the datatype of the schedule object.

135.1-2013 - 13.10.7 - Is able to change the datatype that a Schedule object schedules		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	The tester should attempt to change the datatype of the Schedule to each schedule datatype supported by the IUT.

## 6.2 Scheduling - View Modify - A

### 6.2.1 Base Requirements

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

135.1-2013 - 8.18.3 - Reading and Presenting Properties		
	<b>Test Conditionality</b>	This test may be skipped if the IUT claims support for SCHED-AVM-A.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	Verify for Weekly_Schedule, Exception_Schedule, Effective_Period and Date_List properties. See Addendum L for requirements details. The reference schedule used during this test should include an Exception_Schedule that contains 255 entries and contain 12 BACnetTimeValue tuples per entry. The reference schedule should also contain a Weekly_Schedule which contains 6 BACnetTimeValue tuples per day. The Calendar Date_List used in this test should contain 32 calendar entries.
135.1-2013 - 8.22.4 - Accepting Input and Modifying Properties		
	<b>Test Conditionality</b>	This test may be skipped if the IUT claims support for SCHED-AVM-A.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	Verify for Weekly_Schedule, Exception_Schedule, Effective_Period and Date_List properties. See Addendum L for requirements details. The reference schedule used during this test should include an Exception_Schedule that contains 255 entries and contain 12 BACnetTimeValue tuples per entry. The reference schedule should also contain a Weekly_Schedule which contains 6 BACnetTimeValue tuples per day. The Calendar Date_List used in this test should contain 32 calendar entries.

### 6.2.2 Supports DS-RP-A

The IUT supports DS-RP-A.

Verify Checklist		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify the IUT claims support for the BIBB DS-RP-A.
	<b>Testing Hints</b>	

### 6.2.3 Supports DS-WP-A

The IUT supports DS-WP-A

Verify Checklist		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify the IUT claims support for the BIBB DS-WP-A.
	<b>Testing Hints</b>	

## 6.2.4 Is able to Schedule any B-Side Device regardless of the claimed Protocol\_Revision in the B-side device

The IUT is able to schedule any server device that supports schedules regardless of the BACnet revision claimed by either device.

135.1-2013 - 13.10 - Workstation Scheduling Tests		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	A reference device with schedules supporting Protocol_Revision = 3 and a reference device with schedules supporting Protocol_Revision >= 4 should be used. Any of the tests in this section may be executed.

## 6.2.5 Is able to Present and Modify Schedules that do not include the Exception\_Schedule Property

The IUT is able to present and modify a schedule object that does not include the Exception\_Schedule property.

135.1-2013 - 13.10.2.2 - Modify a Weekly_Schedule by Changing the Value of a BACnetTimeValue		
	Test Conditionality	Must be executed.
	Test Directives	Schedule in the reference server (S2) shall not include the Exception_Schedule property.
	Testing Hints	Must be executed, but using reference schedule S2 which does not include the Exception_Schedule property.

## 6.2.6 Is able to Present and Modify a Calendar Object

135.1-2013 - 13.10.5.1 - Modify a Calendar Object by deleting a BACnetCalendarEntry from the Date_List		
	Test Conditionality	
	Test Directives	
	Testing Hints	
135.1-2013 - 13.10.5.2 - Modify a Calendar Object by adding a BACnetCalendarEntry of choice Date to the Date_List		
	Test Conditionality	
	Test Directives	
	Testing Hints	The tester should try different Date representations that leave one or more of each of the following fields unspecified: year, month, dayOfMonth, dayOfWeek. If the IUT supports devices with Protocol_Revision >= 4, the tester should test with the special values for month of 13 (all odd) and 14 (all even), and the special value of 32 (last day) for the dayOfMonth.
135.1-2013 - 13.10.5.3 - Modify a Calendar Object by adding a BACnetCalendarEntry of choice DateRange to the Date_List		
	Test Conditionality	
	Test Directives	
	Testing Hints	
135.1-2013 - 13.10.5.4 - Modify a Calendar Object by adding a BACnetCalendarEntry of choice WeekNDay to the Date_List		
	Test Conditionality	
	Test Directives	

	<b>Testing Hints</b>	The tester shall use different WeekNDay representations that leave one or more of each of the following fields unspecified: month, weekOfMonth, dayOfWeek. If the IUT supports devices with Protocol_Revision >= 4, the tester should run this test using the special values for month of 13 (all odd) and 14 (all even).
--	----------------------	---

## 6.2.7 Is able to Present and Modify Schedules of Enumerated Type

The IUT can read, present and modify schedules that contain time value pairs of the type Enumerated.

<b>135.1-2013 - 13.10.1 - Read and Present a Weekly Schedule</b>		
	<b>Test Conditionality</b>	The reference server schedule (S1) shall contain BACnetTimeValue of type Enumerated.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 13.10.2.1 - Modify a Weekly Schedule by Changing the Time of a BACnetTimeValue</b>		
	<b>Test Conditionality</b>	The reference server schedule (S1) shall contain BACnetTimeValue of type Enumerated.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	The tester should attempt to change the reference schedule by selecting several of the existing BACnetTimeValue pairs and modifying the time only.
<b>135.1-2013 - 13.10.2.2 - Modify a Weekly Schedule by Changing the Value of a BACnetTimeValue</b>		
	<b>Test Conditionality</b>	The reference server schedule (S1) shall contain BACnetTimeValue of type Enumerated.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	The tester should attempt to change the reference schedule by selecting several of the existing BACnetTimeValue pairs and modifying the value only. The tester should try Enumerated type as well as NULL data type values.
<b>135.1-2013 - 13.10.2.3 - Modify a Weekly Schedule by Deleting a BACnetTimeValue</b>		
	<b>Test Conditionality</b>	The reference server schedule (S1) shall contain BACnetTimeValue of type Enumerated.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 13.10.2.4 - Modify a Weekly Schedule by Adding a BACnetTimeValue</b>		
	<b>Test Conditionality</b>	The reference server schedule (S1) shall contain BACnetTimeValue of type Enumerated.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	The tester should attempt to add a BACnetTimeValue which contains a value of NULL, as well as other Enumerated values.
<b>135.1-2013 - 13.10.3 - Read and Present a Complex Schedule</b>		
	<b>Test Conditionality</b>	The reference server schedule (S1) shall contain BACnetTimeValue of type Enumerated.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 13.10.4.1 - Modify an Exception_Schedule by changing the time of a BACnetTimeValue in the listofTimeValues of a BACnetSpecialEvent with period of choice calendarEntry</b>		
	<b>Test Conditionality</b>	The reference server schedule (S1) shall contain BACnetTimeValue of type Enumerated.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	The tester should attempt to change an existing BACnetTimeValue pair in the reference schedule such that the time is different.

<b>135.1-2013 - 13.10.4.2 - Modify an Exception_Schedule by changing the value of a BACnetTimeValue in the listofTimeValues of a BACnetSpecialEvent with period of choice calendarEntry</b>		
	<b>Test Conditionality</b>	The reference server schedule (S1) shall contain BACnetTimeValue of type Enumerated.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	The tester should attempt to change an existing BACnetTimeValue pair in the reference schedule such that the value is different.
<b>135.1-2013 - 13.10.4.3 - Modify an Exception_Schedule by deleting a BACnetTimeValue from the listofTimeValues of a BACnetSpecialEvent with period of choice calendarEntry</b>		
	<b>Test Conditionality</b>	The reference server schedule (S1) shall contain BACnetTimeValue of type Enumerated.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	The tester should attempt to delete an existing BACnetTimeValue pair in the reference schedule.
<b>135.1-2013 - 13.10.4.4 - Modify an Exception_Schedule by adding a BACnetTimeValue to the listOfTimeValues of a BACnetSpecialEvent with period of choice calendarEntry</b>		
	<b>Test Conditionality</b>	The reference server schedule (S1) shall contain BACnetTimeValue of type Enumerated.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	The tester should attempt to add a BACnetTimeValue pair to the reference schedule.
<b>135.1-2013 - 13.10.4.5 - Modify an Exception_Schedule by changing the eventPriority of a BACnetSpecialEvent with period of choice calendarEntry</b>		
	<b>Test Conditionality</b>	The reference server schedule (S1) shall contain BACnetTimeValue of type Enumerated.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	The tester should attempt to change the eventPriority of an existing BACnetSpecialEvent in the reference schedule.
<b>135.1-2013 - 13.10.4.6 - Modify an Exception_Schedule by deleting a BACnetSpecialEvent with period of choice calendarEntry</b>		
	<b>Test Conditionality</b>	The reference server schedule (S1) shall contain BACnetTimeValue of type Enumerated.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	The tester should attempt to delete an entire BACnetSpecialEvent from the reference schedule.
<b>135.1-2013 - 13.10.4.7 - Modify an Exception_Schedule by adding a BACnetSpecialEvent with period of choice calendarEntry of choice Date</b>		
	<b>Test Conditionality</b>	The reference server schedule (S1) shall contain BACnetTimeValue of type Enumerated.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	The tester should attempt to add an entire BACnetSpecialEvent of choice Date to the reference schedule.
<b>135.1-2013 - 13.10.4.8 - Modify an Exception_Schedule by adding a BACnetSpecialEvent with period of choice calendarEntry of choice DateRange</b>		
	<b>Test Conditionality</b>	The reference server schedule (S1) shall contain BACnetTimeValue of type Enumerated.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	The tester should attempt to add an entire BACnetSpecialEvent of choice DateRange to the reference schedule.
<b>135.1-2013 - 13.10.4.9 - Modify an Exception_Schedule by adding a BACnetSpecialEvent with period of choice calendarEntry of choice WeekNDay</b>		
	<b>Test Conditionality</b>	The reference server schedule (S1) shall contain BACnetTimeValue of type Enumerated.

	<b>Test Directives</b>	
	<b>Testing Hints</b>	The tester should attempt to add an entire BACnetSpecialEvent of choice WeekNDay to the reference schedule.
<b>135.1-2013 - 13.10.4.10 - Modify an Exception_Schedule by adding a BACnetSpecialEvent with period of choice calendarReference</b>		
	<b>Test Conditionality</b>	The reference server schedule (S1) shall contain BACnetTimeValue of type Enumerated.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	The tester should attempt to add an entire BACnetSpecialEvent of choice calendarReference to the reference schedule.
<b>135.1-2013 - 13.10.4.11 - Modify an Exception_Schedule by changing the time of a BACnetTimeValue in the listofTimeValues of a BACnetSpecialEvent with period of choice calendarReference</b>		
	<b>Test Conditionality</b>	The reference server schedule (S1) shall contain BACnetTimeValue of type Enumerated.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	The tester should attempt to change an existing BACnetTimeValue pair in the reference schedule such that the time is different.
<b>135.1-2013 - 13.10.4.12 - Modify an Exception_Schedule by changing the value of a BACnetTimeValue in the listofTimeValues of a BACnetSpecialEvent with period of choice calendarReference</b>		
	<b>Test Conditionality</b>	The reference server schedule (S1) shall contain BACnetTimeValue of type Enumerated.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	The tester should attempt to change an existing BACnetTimeValue pair in the reference schedule such that the value is different.
<b>135.1-2013 - 13.10.4.13 - Modify an Exception_Schedule by deleting a BACnetTimeValue from the listofTimeValues of a BACnetSpecialEvent with period of choice calendarReference</b>		
	<b>Test Conditionality</b>	The reference server schedule (S1) shall contain BACnetTimeValue of type Enumerated.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	The tester should attempt to delete an existing BACnetTimeValue pair in the reference schedule.
<b>135.1-2013 - 13.10.4.14 - Modify an Exception_Schedule by adding a BACnetTimeValue to the listofTimeValues of a BACnetSpecialEvent with period of choice calendarReference</b>		
	<b>Test Conditionality</b>	The reference server schedule (S1) shall contain BACnetTimeValue of type Enumerated.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	The tester should attempt to add a BACnetTimeValue pair to the reference schedule.
<b>135.1-2013 - 13.10.4.15 - Modify an Exception_Schedule by deleting a BACnetSpecialEvent with period of choice calendarReference</b>		
	<b>Test Conditionality</b>	The reference server schedule (S1) shall contain BACnetTimeValue of type Enumerated.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	The tester should attempt to delete an entire BACnetSpecialEvent from the reference schedule.

## 6.2.8 Is able to Present and Modify Schedules of REAL Type

The IUT can read, present and modify schedules that contain time value pairs of the type REAL.

Execute all of the tests in section 6.2.7 above, but with the scheduled datatype changed to be type REAL.

### **6.2.9 Is able to Present and Modify Schedules of Unsigned32 Type**

The IUT can read, present and modify schedules that contain time value pairs of the type Unsigned32.

Execute all of the tests in section 6.2.7 above, but with the scheduled datatype changed to be type Unsigned32.

### **6.2.10 Is able to Present and Modify Schedules of BOOLEAN Type**

The IUT can read, present and modify schedules that contain time value pairs of the type BOOLEAN.

Execute all of the tests in section 6.2.7 above, but with the scheduled datatype changed to be type BOOLEAN.

### **6.2.11 Is able to Present and Modify Schedules of INTEGER (Signed) Type**

The IUT can read, present and modify schedules that contain time value pairs of the type INTEGER.

Execute all of the tests in section 6.2.7 above, but with the scheduled datatype changed to be type INTEGER.

### **6.2.12 Is able to Present and Modify Schedules of Double Type**

The IUT can read, present and modify schedules that contain time value pairs of the type Double.

Execute all of the tests in section 6.2.7 above, but with the scheduled datatype changed to be type Double.

### **6.2.13 Is able to Present and Modify Schedules of Octet String Type**

The IUT can read, present and modify schedules that contain time value pairs of the type Octet String.

Execute all of the tests in section 6.2.7 above, but with the scheduled datatype changed to be type Octet String.

### **6.2.14 Is able to Present and Modify Schedules of Character String Type**

The IUT can read, present and modify schedules that contain time value pairs of the type Character String.

Execute all of the tests in section 6.2.7 above, but with the scheduled datatype changed to be type Character String.

### **6.2.15 Is able to Present and Modify Schedules of Bit String Type**

The IUT can read, present and modify schedules that contain time value pairs of the type Bit String.

Execute all of the tests in section 6.2.7 above, but with the scheduled datatype changed to be type Bit String.

### **6.2.16 Is able to Present and Modify Schedules of Date Type**

The IUT can read, present and modify schedules that contain time value pairs of the type Date.

Execute all of the tests in section 6.2.7 above, but with the scheduled datatype changed to be type Date.

### **6.2.17 Is able to Present and Modify Schedules of Time Type**

The IUT can read, present and modify schedules that contain time value pairs of the type Time.

Execute all of the tests in section 6.2.7 above, but with the scheduled datatype changed to be type Time.

### **6.2.18 Is able to Present and Modify Schedules of BACnetObjectIdentifier Type**

The IUT can read, present and modify schedules that contain time value pairs of the type BACnetObjectIdentifier.

Execute all of the tests in section 6.2.7 above, but with the scheduled datatype changed to be type BACnetObjectIdentifier.



## 6.3 Scheduling - Weekly Schedule - A

### 6.3.1 Base Requirements

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

135.1-2013 - 8.18.3 - Reading and Presenting Properties		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	Verify for Weekly_Schedule and Schedule_Default properties.
135.1-2013 - 8.22.4 - Accepting Input and Modifying Properties		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	Verify for Weekly_Schedule and Schedule_Default properties.

### 6.3.2 Supports DS-RP-A

The IUT supports DS-RP-A.

Verify Checklist		
	Test Conditionality	Must be executed.
	Test Directives	Verify the IUT claims support for the BIBB DS-RP-A.
	Testing Hints	

### 6.3.3 Supports DS-WP-A

The IUT supports DS-WP-A

Verify Checklist		
	Test Conditionality	Must be executed.
	Test Directives	Verify the IUT claims support for the BIBB DS-WP-A.
	Testing Hints	

### 6.3.4 Is able to Schedule any B-Side Device with a Protocol\_Revision Less Than or Equal to its Own Protocol\_Revision

The IUT is able to schedule any server device that supports schedules from the same or previous revisions of BACnet.

135.1-2013 - 13.10 - Workstation Scheduling Tests		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	A reference device with schedules supporting Protocol_Revision = 3 and a reference device with schedules supporting Protocol_Revision >= 4 should be used. Any of the tests in this section may be executed.

### 6.3.5 Is able to Present and Modify Weekly\_Schedule of Enumerated Type

The IUT can read, present and modify schedules that contain time value pairs of the type Enumerated.

135.1-2013 - 13.10.1 - Read and Present a Weekly_Schedule		
	Test Conditionality	The reference server schedule (S1) shall contain BACnetTimeValue of type Enumerated.
	Test Directives	
	Testing Hints	

<b>135.1-2013 - 13.10.2.1 - Modify a Weekly Schedule by Changing the Time of a BACnetTimeValue</b>		
	<b>Test Conditionality</b>	The reference server schedule (S1) shall contain BACnetTimeValue of type Enumerated.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	The tester should attempt to change the reference schedule by selecting several of the existing BACnetTimeValue pairs and modifying the time only.
<b>135.1-2013 - 13.10.2.2 - Modify a Weekly Schedule by Changing the Value of a BACnetTimeValue</b>		
	<b>Test Conditionality</b>	The reference server schedule (S1) shall contain BACnetTimeValue of type Enumerated.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	The tester should attempt to change the reference schedule by selecting several of the existing BACnetTimeValue pairs and modifying the value only. The tester should try Enumerated type as well as NULL data type values.
<b>135.1-2013 - 13.10.2.3 - Modify a Weekly Schedule by Deleting a BACnetTimeValue</b>		
	<b>Test Conditionality</b>	The reference server schedule (S1) shall contain BACnetTimeValues of type Enumerated.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 13.10.2.4 - Modify a Weekly Schedule by Adding a BACnetTimeValue</b>		
	<b>Test Conditionality</b>	The reference server schedule (S1) shall contain BACnetTimeValue of type Enumerated.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	The tester should attempt to add a BACnetTimeValue which contains a value of NULL, as well as other Enumerated values.

### 6.3.6 Is able to Present and Modify Weekly\_Schedule of REAL Type

The IUT can read, present and modify schedules that contain time value pairs of the type REAL.

Execute all of the tests in section 6.3.5 above, but with the scheduled datatype changed to be type REAL.

### 6.3.7 Is able to Present and Modify Weekly\_Schedule of BOOLEAN Type

The IUT can read, present and modify schedules that contain time value pairs of the type BOOLEAN.

Execute all of the tests in section 6.3.5 above, but with the scheduled datatype changed to be type BOOLEAN.

### 6.3.8 Is able to Present and Modify Weekly\_Schedule of Unsigned Type

The IUT can read, present and modify schedules that contain time value pairs of the type Unsigned.

Execute all of the tests in section 6.3.5 above, but with the scheduled datatype changed to be type Unsigned.

### 6.3.9 Is able to Present and Modify Weekly\_Schedule of INTEGER (Signed) Type

The IUT can read, present and modify schedules that contain time value pairs of the type INTEGER.

Execute all of the tests in section 6.3.5 above, but with the scheduled datatype changed to be type INTEGER.

### **6.3.10 Is able to Present and Modify Weekly\_Schedule of Double Type**

The IUT can read, present and modify schedules that contain time value pairs of the type Double.

Execute all of the tests in section 6.3.5 above, but with the scheduled datatype changed to be type Double.

### **6.3.11 Is able to Present and Modify Weekly\_Schedule of Octet String Type**

The IUT can read, present and modify schedules that contain time value pairs of the type Octet String.

Execute all of the tests in section 6.3.5 above, but with the scheduled datatype changed to be type Octet String.

### **6.3.12 Is able to Present and Modify Weekly\_Schedule of Character String Type**

The IUT can read, present and modify schedules that contain time value pairs of the type Character String.

Execute all of the tests in section 6.3.5 above, but with the scheduled datatype changed to be type Character String.

### **6.3.13 Is able to Present and Modify Weekly\_Schedule of Bit String Type**

The IUT can read, present and modify schedules that contain time value pairs of the type Bit String.

Execute all of the tests in section 6.3.5 above, but with the scheduled datatype changed to be type Bit String.

### **6.3.14 Is able to Present and Modify Weekly\_Schedule of Date Type**

The IUT can read, present and modify schedules that contain time value pairs of the type Date.

Execute all of the tests in section 6.3.5 above, but with the scheduled datatype changed to be type Date.

### **6.3.15 Is able to Present and Modify Weekly\_Schedule of Time Type**

The IUT can read, present and modify schedules that contain time value pairs of the type Time.

Execute all of the tests in section 6.3.5 above, but with the scheduled datatype changed to be type Time.

### **6.3.16 Is able to Present and Modify Weekly\_Schedule of BACnetObjectIdentifier Type**

The IUT can read, present and modify schedules that contain time value pairs of the type BACnetObjectIdentifier.

Execute all of the tests in section 6.3.5 above, but with the scheduled datatype changed to be type BACnetObjectIdentifier.

## 6.4 Scheduling - Internal - B

### 6.4.1 Base Requirements

Base requirements must be met by any IUT claiming conformance to this BIBB. (The BIBB requires, among other things, support for either TimeSynchronization-Request or UTCTimeSynchronization-Request execution; these are tested by the Device Management tests.)

<b>135.1-2013 - 7.3.2.23.2 - Weekly Schedule Property Test</b>		
	<b>Test Conditionality</b>	As per <i>ASHRAE 135.1-2013</i> if and only if the IUT is prior to protocol revision 4. If the IUT is of the correct Protocol_Revision, the IUT is required to be configurable such that this test can be run. This test may not be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.23.10.2 - Revision 4 Weekly Schedule Property Test</b>		
	<b>Test Conditionality</b>	As per <i>ASHRAE 135.1-2013</i> if and only if the IUT is protocol revision 4 or higher. If the IUT is of the correct Protocol_Revision, the IUT is required to be configurable such that this test can be run. This test may not be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.23.6 - Weekly Schedule Restoration Test</b>		
	<b>Test Conditionality</b>	The IUT is required to be configurable such that this test can be run. This test may not be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.23.3.8 - Event Priority Test</b>		
	<b>Test Conditionality</b>	As per <i>ASHRAE 135.1-2013</i> if and only if the IUT is prior to protocol revision 4. If the IUT is of the correct Protocol_Revision, the IUT is required to be configurable such that this test can be run. This test may not be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.23.10.3.8 - Revision 4 Event Priority Test</b>		
	<b>Test Conditionality</b>	If the IUT does not support enough exception schedule entries to execute this test, the test shall be skipped, otherwise the test shall be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.23.3.9 - List of BACnetTimeValue Test</b>		
	<b>Test Conditionality</b>	As per <i>ASHRAE 135.1-2013</i> if and only if the IUT is prior to protocol revision 4. If the IUT is of the correct Protocol_Revision, the IUT is required to be configurable such that this test can be run. This test may not be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.23.10.3.9 - Revision 4 List of BACnetTimeValue Test</b>		
	<b>Test Conditionality</b>	As per <i>ASHRAE 135.1-2013</i> if and only if the IUT is protocol revision 4 or higher. If the IUT is of the correct Protocol_Revision, the IUT is required to be configurable such that this test can be run. This test may not be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.23.5 - Exception Schedule Restoration Test</b>		

	<b>Test Conditionality</b>	The IUT is required to be configurable such that this test can be run. This test may not be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.23.3.2 - Calendar Entry Date Test</b>		
	<b>Test Conditionality</b>	As per <i>ASHRAE 135.1-2013</i> if and only if the IUT is prior to protocol revision 4. If the IUT is of the correct Protocol_Revision, the IUT is required to be configurable such that this test can be run. This test may not be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.23.10.3.2 - Revision 4 Calendar Entry Date Test</b>		
	<b>Test Conditionality</b>	As per <i>ASHRAE 135.1-2013</i> if and only if the IUT is protocol revision 4 or higher. If the IUT is of the correct Protocol_Revision, the IUT is required to be configurable such that this test can be run. This test may not be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.23.3.3 - Calendar Entry DateRange Test</b>		
	<b>Test Conditionality</b>	As per <i>ASHRAE 135.1-2013</i> if and only if the IUT is prior to protocol revision 4. If the IUT is of the correct Protocol_Revision, the IUT is required to be configurable such that this test can be run. This test may not be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.23.10.3.3 - Revision 4 Calendar Entry DateRange Test</b>		
	<b>Test Conditionality</b>	As per <i>ASHRAE 135.1-2013</i> if and only if the IUT is protocol revision 4 or higher. If the IUT is of the correct Protocol_Revision, the IUT is required to be configurable such that this test can be run. This test may not be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.23.3.4 - Calendar Entry WeekNDay Month Test</b>		
	<b>Test Conditionality</b>	As per <i>ASHRAE 135.1-2013</i> if and only if the IUT is prior to protocol revision 4. If the IUT is of the correct Protocol_Revision, the IUT is required to be configurable such that this test can be run. This test may not be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.23.10.3.4 - Revision 4 Calendar Entry WeekNDay Month Test</b>		
	<b>Test Conditionality</b>	As per <i>ASHRAE 135.1-2013</i> if and only if the IUT is protocol revision 4 or higher. If the IUT is of the correct Protocol_Revision, the IUT is required to be configurable such that this test can be run. This test may not be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.23.3.5 - Calendar Entry WeekNDay Week Of Month Test</b>		
	<b>Test Conditionality</b>	As per <i>ASHRAE 135.1-2013</i> if and only if the IUT is prior to protocol revision 4. If the IUT is of the correct Protocol_Revision, the IUT is required to be configurable such that this test can be run. This test may not be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.23.10.3.5 - Revision 4 Calendar Entry WeekNDay Week Of Month Test</b>		

	<b>Test Conditionality</b>	As per <i>ASHRAE 135.1-2013</i> if and only if the IUT is protocol revision 4 or higher. If the IUT is of the correct Protocol_Revision, the IUT is required to be configurable such that this test can be run. This test may not be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.23.3.6 - Calendar Entry WeekNDay Last Week Of Month Test</b>		
	<b>Test Conditionality</b>	As per <i>ASHRAE 135.1-2013</i> if and only if the IUT is prior to protocol revision 4. If the IUT is of the correct Protocol_Revision, the IUT is required to be configurable such that this test can be run. This test may not be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.23.10.3.6 - Revision 4 Calendar Entry WeekNDay Last Week Of Month Test</b>		
	<b>Test Conditionality</b>	As per <i>ASHRAE 135.1-2013</i> if and only if the IUT is protocol revision 4 or higher. If the IUT is of the correct Protocol_Revision, the IUT is required to be configurable such that this test can be run. This test may not be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.23.3.7 - Calendar Entry WeekNDay Day Of Week Test</b>		
	<b>Test Conditionality</b>	As per <i>ASHRAE 135.1-2013</i> if and only if the IUT is prior to protocol revision 4. If the IUT is of the correct Protocol_Revision, the IUT is required to be configurable such that this test can be run. This test may not be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.23.10.3.7 - Revision 4 Calendar Entry WeekNDay Day Of Week Test</b>		
	<b>Test Conditionality</b>	As per <i>ASHRAE 135.1-2013</i> if and only if the IUT is protocol revision 4 or higher. If the IUT is of the correct Protocol_Revision, the IUT is required to be configurable such that this test can be run. This test may not be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.23.10.3.10 - Revision 4 Calendar Entry WeekNDay Odd-Numbered Month Test</b>		
	<b>Test Conditionality</b>	As per <i>ASHRAE 135.1-2013</i> if and only if the IUT is protocol revision 4 or higher. If the IUT is of the correct Protocol_Revision, the IUT is required to be configurable such that this test can be run. This test may not be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.23.10.3.11 - Revision 4 Calendar Entry WeekNDay Even-Numbered Month Test</b>		
	<b>Test Conditionality</b>	As per <i>ASHRAE 135.1-2013</i> if and only if the IUT is protocol revision 4 or higher. If the IUT is of the correct Protocol_Revision, the IUT is required to be configurable such that this test can be run. This test may not be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.23.10.3.12 - Revision 4 Lower Event Priority Change Test</b>		
	<b>Test Conditionality</b>	If the IUT does not support enough exception schedule entries to execute this test, the test shall be skipped, otherwise the test shall be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.23.10.3.13 - Revision 4 Schedule_Default Test</b>		

# BTL Test Plan

	<b>Test Conditionality</b>	As per <i>ASHRAE 135.1-2013</i> if and only if the IUT is protocol revision 4 or higher. If the IUT is of the correct Protocol_Revision, the IUT is required to be configurable such that this test can be run. This test may not be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.23.12 - Revision 4 Midnight Evaluation Test</b>		
	<b>Test Conditionality</b>	This test shall be executed if and only if the IUT is protocol revision 4 or higher. If the IUT is of the correct Protocol_Revision, the IUT is required to be configurable such that this test can be run. This test may not be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.2.X1 - Date Pattern Properties Test</b>		
	<b>Test Conditionality</b>	This test shall only be applied to devices claiming Protocol_Revision 11 or higher.
	<b>Test Directives</b>	Apply to BACnetCalendarEntry in the Exception_Schedule property in the BACnet Date form.
	<b>Testing Hints</b>	
<b>BTL - 7.2.X5 - Time Non-Pattern Properties Test</b>		
	<b>Test Conditionality</b>	This test shall only be applied to devices claiming Protocol_Revision 11 or higher.
	<b>Test Directives</b>	Apply to the time portion of BACnetTimeValues in the Exception_Schedule property, then apply to the time portion of BACnetTimeValues in the Weekly_Schedule property.
	<b>Testing Hints</b>	
<b>BTL - 7.2.X7 - BACnetDateRange Non-Pattern Properties Test</b>		
	<b>Test Conditionality</b>	This test shall only be applied to devices claiming Protocol_Revision 11 or higher.
	<b>Test Directives</b>	Apply to BACnetCalendarEntry in the Exception_Schedule property in the BACnetDateRange form.
	<b>Testing Hints</b>	
<b>BTL - 7.2.X8 - BACnetDateRange Open-Ended Pattern Properties Test</b>		
	<b>Test Conditionality</b>	This test shall only be applied to devices claiming Protocol_Revision 11 or higher.
	<b>Test Directives</b>	Apply to BACnetCalendarEntry in the Exception_Schedule property in the BACnetDateRange form.
	<b>Testing Hints</b>	
<b>BTL - 9.23.2.X10 - Time Non-Pattern Properties Test using WritePropertyMultiple Service</b>		
	<b>Test Conditionality</b>	This test shall only be applied to devices claiming Protocol_Revision 11 or higher and which supports execution of WritePropertyMultiple.
	<b>Test Directives</b>	Apply to the time portion of BACnetTimeValues in the Exception_Schedule property, then apply to the time portion of BACnetTimeValues in the Weekly_Schedule property.
	<b>Testing Hints</b>	
<b>BTL - 9.23.2.X12 - BACnetDateRange Non-Pattern Properties Test using WritePropertyMultiple Service</b>		
	<b>Test Conditionality</b>	This test shall only be applied to devices claiming Protocol_Revision 11 or higher and which supports execution of WritePropertyMultiple.
	<b>Test Directives</b>	Apply to BACnetCalendarEntry in the Exception_Schedule property in the BACnetDateRange form.
	<b>Testing Hints</b>	
<b>135.1-2013r - 7.3.2.23.X - Forbid Duplicate Time Values</b>		
	<b>Test Conditionality</b>	If Protocol_Revision < 16, then this test shall be skipped.

	<b>Test Directives</b>	Apply to a writable Weekly_Schedule and then to an Exception_Schedule property of a schedule object
	<b>Testing Hints</b>	

## 6.4.2 Supports Concurrent Weekly and Exception Schedules

The IUT supports Schedule objects that support both Weekly and Exception Schedules concurrently, as is required for a BTL Listing.

135.1-2013 - 7.3.2.23.4 - Weekly Schedule and Exception Schedule Interaction Test		
	<b>Test Conditionality</b>	This test shall be executed if and only if the IUT is prior to protocol revision 4. If the IUT is of the correct Protocol_Revision, the IUT is required to be configurable such that this test can be run. This test may not be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
135.1-2013 - 7.3.2.23.10.4 - Revision 4 Weekly Schedule and Exception Schedule Interaction Test		
	<b>Test Conditionality</b>	This test shall be executed if and only if the IUT is protocol revision 4 or higher. If the IUT is of the correct Protocol_Revision, the IUT is required to be configurable such that this test can be run. This test may not be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

## 6.4.3 Supports Reference to Calendar Object

The IUT supports Schedule objects that support references to Calendar objects, as required for a BTL listing.

135.1-2013 - 7.3.2.23.3.1 - Calendar Reference Test		
	<b>Test Conditionality</b>	As per <i>ASHRAE 135.1-2013</i> if and only if the IUT is prior to protocol revision 4. If the IUT is of the correct Protocol_Revision, the IUT is required to be configurable such that this test can be run. This test may not be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
135.1-2013 - 7.3.2.23.10.3.1 - Revision 4 Calendar Reference Test		
	<b>Test Conditionality</b>	As per <i>ASHRAE 135.1-2013</i> if and only if the IUT is protocol revision 4 or higher. If the IUT is of the correct Protocol_Revision, the IUT is required to be configurable such that this test can be run. This test may not be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

## 6.4.4 Supports Configurable Effective\_Period

The IUT supports the Effective\_Period property and it is configurable.

135.1-2013 - 7.3.2.23.1 - Effective_Period Test		
	<b>Test Conditionality</b>	This test shall be executed if and only if the IUT is prior to protocol revision 4. If the IUT is of the correct Protocol_Revision, the IUT is required to be configurable such that this test can be run. This test may not be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
135.1-2013 - 7.3.2.23.10.1 - Revision 4 Effective_Period Test		



	<b>Test Conditionality</b>	This test shall be executed if and only if the IUT is protocol revision 4 or higher. If the IUT is of the correct Protocol_Revision, the IUT is required to be configurable such that this test can be run. This test may not be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.2.X7 - BACnetDateRange Non-Pattern Properties Test</b>		
	<b>Test Conditionality</b>	This test shall only be applied to devices claiming Protocol_Revision 11 or higher.
	<b>Test Directives</b>	Apply to the Effective_Period property.
	<b>Testing Hints</b>	
<b>BTL - 7.2.X8 - BACnetDateRange Open-Ended Pattern Properties Test</b>		
	<b>Test Conditionality</b>	This test shall only be applied to devices claiming Protocol_Revision 11 or higher.
	<b>Test Directives</b>	Apply to the Effective_Period property.
	<b>Testing Hints</b>	
<b>BTL - 9.23.2.X12 - BACnetDateRange Non-Pattern Properties Test using WritePropertyMultiple Service</b>		
	<b>Test Conditionality</b>	This test shall only be applied to devices claiming Protocol_Revision 11 or higher and which supports execution of WritePropertyMultiple.
	<b>Test Directives</b>	Apply to the Effective_Period property.
	<b>Testing Hints</b>	

#### 6.4.5 Is Able to Schedule Values of Different Datatypes (not Simultaneously)

The IUT supports Schedule objects that are capable of being configured to schedule values of different datatypes. An example would be a Schedule object that was originally configured to write values of datatype REAL to an Analog Value object's Present\_Value property, but is later reconfigured to write Enumerated values to a Binary Value object's Present\_Value.

Such a reconfiguration involves changing the List\_Of\_Object\_Property\_References property to refer to properties of a specific datatype, and changing the 'value' members of the BACnetTimeValue pairs in the Weekly\_Schedule and Exception\_Schedule properties to have values in the same datatype as the properties referenced by the List\_Of\_Object\_Property\_References property.

<b>BTL - 7.3.2.23.11.1 - Internally Written Datatypes Test, non-NULL values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.23.11.2 - Internally Written Datatypes Test, NULL values and Priority Arrays</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 6.4.6 Supports Non-Empty List\_Of\_Object\_Property\_References Property

The IUT supports a non-empty List\_Of\_Object\_Property\_Reference property.

<b>135.1-2013 - 7.3.2.23.7 - List Of Object Property Reference Internal Test</b>		
	<b>Test Conditionality</b>	This test shall be executed if and only if the IUT is prior to protocol revision 4. If the IUT is of the correct Protocol_Revision, the IUT is required to be configurable such that this test can be run. This test may not be skipped.

	<b>Test Directives</b>	
	<b>Testing Hints</b>	
	<b>135.1-2013 - 7.3.2.23.10.7 - Revision 4 List Of Object Property Reference Internal Test</b>	
	<b>Test Conditionality</b>	This test shall be executed if and only if the IUT is protocol revision 4 or later. If the IUT is of the correct Protocol_Revision, the IUT is required to be configurable such that this test can be run. This test may not be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 6.4.7 Is Able to Schedule NULL Values

The IUT supports a Schedule object that is able to schedule (write) NULL with a priority.

<b>135.1-2013 - 7.3.2.23.11.2 - Internally Written Datatypes Test, NULL values and Priority Arrays</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 6.4.8 Is Able to Schedule BOOLEAN Values

The IUT supports a Schedule object that is able to schedule (write) BOOLEAN values.

<b>BTL - 7.3.2.23.11.1 - Internally Written Datatypes Test, non-NULL values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 6.4.9 Is Able to Schedule Unsigned Values

The IUT supports a Schedule object that is able to schedule (write) Unsigned values.

<b>BTL - 7.3.2.23.11.1 - Internally Written Datatypes Test, non-NULL values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 6.4.10 Is Able to Schedule INTEGER (Signed) Values

The IUT supports a Schedule object that is able to schedule (write) INTEGER values.

<b>BTL - 7.3.2.23.11.1 - Internally Written Datatypes Test, non-NULL values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 6.4.11 Is Able to Schedule REAL Values

The IUT supports a Schedule object that is able to schedule (write) REAL values.

<b>BTL - 7.3.2.23.11.1 - Internally Written Datatypes Test, non-NULL values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

#### 6.4.12 Is Able to Schedule Double Values

The IUT supports a Schedule object that able to schedule (write) Double values.

BTL - 7.3.2.23.11.1 - Internally Written Datatypes Test, non-NULL values		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

#### 6.4.13Is Able to Schedule Octet String Values

The IUT supports a Schedule object that is able to schedule (write) Octet String values.

BTL - 7.3.2.23.11.1 - Internally Written Datatypes Test, non-NULL values		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

#### 6.4.14Is Able to Schedule Character String Values

The IUT supports a Schedule object that is able to schedule (write) Character String values.

BTL - 7.3.2.23.11.1 - Internally Written Datatypes Test, non-NULL values		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

#### 6.4.15Is Able to Schedule Bit String Values

The IUT supports a Schedule object that is able to schedule (write) Bit String values.

BTL - 7.3.2.23.11.1 - Internally Written Datatypes Test, non-NULL values		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

#### 6.4.16Is Able to Schedule Enumerated Values

The IUT supports a Schedule object that is able to schedule (write) Enumerated values.

BTL - 7.3.2.23.11.1 - Internally Written Datatypes Test, non-NULL values		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

#### 6.4.17Is Able to Schedule Date Values

The IUT supports a Schedule object that is able to schedule (write) Date values.

BTL - 7.3.2.23.11.1 - Internally Written Datatypes Test, non-NULL values		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

#### 6.4.18Is Able to Schedule Time Values

The IUT supports a Schedule object that is able to write schedule (write) Time values.

BTL - 7.3.2.23.11.1 - Internally Written Datatypes Test, non-NULL values		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 6.4.19 Is Able to Schedule BACnetObjectIdentifier Values

The IUT supports a Schedule object that is able to schedule (write) BACnetObjectIdentifier values.

BTL - 7.3.2.23.11.1 - Internally Written Datatypes Test, non-NULL values		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 6.4.20 Supports DM-TS-B

The IUT supports DM-TS-B.

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

### 6.4.21 Supports DM-UTC-B

The IUT supports DM-UTC-B.

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

## 6.5 Scheduling - External - B

### 6.5.1 Base Requirements

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

All SCHED-I-B tests must also be executed.

Base requirements must be met by any IUT claiming conformance to this BIBB. (The BIBB requires, among other things, support for either TimeSynchronization-Request or UTCTimeSynchronization-Request execution; these are tested by the Device Management tests. This BIBB also requires support for the SCHED-I-B BIBB.)

135.1-2013 - 7.3.2.23.8 - List Of Object Property Reference External Test		
	<b>Test Conditionality</b>	This test shall be executed if and only if the IUT is prior to protocol revision 4. If the IUT is of the correct Protocol_Revision, the IUT is required to be configurable such that this test can be run. This test may not be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
135.1-2013 - 7.3.2.23.10.8 - Revision 4 List Of Object Property Reference External Test		
	<b>Test Conditionality</b>	This test shall be executed if and only if the IUT is protocol revision 4 or higher. If the IUT is of the correct Protocol_Revision, the IUT is required to be configurable such that this test can be run. This test may not be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 6.5.2 Supports DS-WP-A

The IUT supports DS-WP-A.

Verify Checklist		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support for DS-WP-A.
	<b>Testing Hints</b>	

### 6.5.3 Supports SCHED-I-B

The IUT supports SCHED-I-B.

Verify Checklist		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support for SCHED-I-B.
	<b>Testing Hints</b>	

### 6.5.4 Supports Writable List\_Of\_Object\_Property\_References Property

The IUT supports a writable List\_Of\_Object\_Property\_Reference property.

No Specific Test		
	<b>Test Conditionality</b>	
	<b>Test Directives</b>	Verify that the EPICS declares that this property is writable.
	<b>Testing Hints</b>	

### 6.5.5 Is Able to Schedule NULL Values

The IUT supports a Schedule object that is able to schedule (write) NULL with a priority.

135.1-2013 - 7.3.2.23.11.4 - Externally Written Datatypes Test, NULL values and Priority Arrays		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 6.5.6 Is Able to Schedule BOOLEAN Values

The IUT supports a Schedule object that is able to schedule (write) BOOLEAN values.

135.1-2013 - 7.3.2.23.11.3 - Externally Written Datatypes Test, non-NULL values		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 6.5.7 Is Able to Schedule Unsigned Values

The IUT supports a Schedule object that is able to schedule (write) Unsigned values.

135.1-2013 - 7.3.2.23.11.3 - Externally Written Datatypes Test, non-NULL values		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 6.5.8 Is Able to Schedule REAL Values

The IUT supports a Schedule object that is able to schedule (write) REAL values.

135.1-2013 - 7.3.2.23.11.3 - Externally Written Datatypes Test, non-NULL values		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 6.5.9 Is Able to Schedule Enumerated Values

The IUT supports a Schedule object that is able to schedule (write) Enumerated values.

135.1-2013 - 7.3.2.23.11.3 - Externally Written Datatypes Test, non-NULL values		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 6.5.10 Is Able to Schedule Values of Different Data Types (Not Simultaneously)

The IUT supports Schedule objects that are capable of being configured to schedule values of different datatypes. An example would be a Schedule object that was originally configured to write values of datatype REAL to an Analog Value object's Present\_Value property, but is later reconfigured to write Enumerated values to a Binary Value object's Present\_Value.

Such a reconfiguration involves changing the List\_Of\_Object\_Property\_References property to refer to properties of a specific datatype, and changing the 'value' members of the BACnetTimeValue pairs in the Weekly\_Schedule and Exception\_Schedule properties to have values in the same datatype as the properties referenced by the List\_Of\_Object\_Property\_References property.

135.1-2013 - 7.3.2.23.11.3 - Externally Written Datatypes Test, non-NULL values		
---	--	--

	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.23.11.4 - Externally Written Datatypes Test, NULL values and Priority Arrays</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 6.5.11Is Able to Schedule INTEGER (Signed) Values

The IUT supports a Schedule object that is able to schedule (write) INTEGER values.

<b>135.1-2013 - 7.3.2.23.11.3 - Externally Written Datatypes Test, non-NULL values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 6.5.12Is Able to Schedule Double Values

The IUT supports a Schedule object that able to schedule (write) Double values.

<b>135.1-2013 - 7.3.2.23.11.3 - Externally Written Datatypes Test, non-NULL values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 6.5.13Is Able to Schedule Character String Values

The IUT supports a Schedule object that is able to schedule (write) Character String values.

<b>135.1-2013 - 7.3.2.23.11.3 - Externally Written Datatypes Test, non-NULL values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 6.5.14Is Able to Schedule Bit String Values

The IUT supports a Schedule object that is able to schedule (write) Bit String values.

<b>135.1-2013 - 7.3.2.23.11.3 - Externally Written Datatypes Test, non-NULL values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 6.5.15Is Able to Schedule Octet String Values

The IUT supports a Schedule object that is able to schedule (write) Octet String values.

<b>135.1-2013 - 7.3.2.23.11.3 - Externally Written Datatypes Test, non-NULL values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 6.5.16Is Able to Schedule Date Values

The IUT supports a Schedule object that is able to schedule (write) Date values.

<b>135.1-2013 - 7.3.2.23.11.3 - Externally Written Datatypes Test, non-NULL values</b>		
	<b>Test Conditionality</b>	Must be executed.

	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 6.5.17Is Able to Schedule Time Values

The IUT supports a Schedule object that is able to write schedule (write) Time values.

135.1-2013 - 7.3.2.23.11.3 - Externally Written Datatypes Test, non-NULL values		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 6.5.18Is Able to Schedule BACnetObjectIdentifier Values

The IUT supports a Schedule object that is able to schedule (write) BACnetObjectIdentifier values.

135.1-2013 - 7.3.2.23.11.3 - Externally Written Datatypes Test, non-NULL values		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	



## 6.6 Scheduling - Weekly Schedule - Internal - B

### 6.6.1 Base Requirements

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

<b>135.1-2013 - 7.3.2.23.2 - Weekly Schedule Property Test</b>		
	<b>Test Conditionality</b>	As per <i>ASHRAE 135.1-2013</i> if and only if the IUT is prior to protocol revision 4. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.23.10.2 - Revision 4 Weekly Schedule Property Test</b>		
	<b>Test Conditionality</b>	As per <i>ASHRAE 135.1-2013</i> if and only if the IUT is protocol revision 4 or higher. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.23.6 - Weekly Schedule Restoration Test</b>		
	<b>Test Conditionality</b>	If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Testing Hints</b>	
	<b>Test Directives</b>	
<b>135.1-2013 - 7.3.2.23.10.7 - Revision 4 List Of Object Property Reference Internal Test</b>		
	<b>Test Conditionality</b>	This test shall be executed if and only if the IUT is protocol revision 4 or higher and if the object has or can be made to have a non-empty List_Of_Object_PropertyReference property. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.23.10.3.13 - Revision 4 Schedule Default Test</b>		
	<b>Test Conditionality</b>	As per <i>ASHRAE 135.1-2013</i> if and only if the IUT is protocol revision 4 or higher. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.23.12 - Revision 4 Midnight Evaluation Test</b>		
	<b>Test Conditionality</b>	This test shall be executed if and only if the IUT is protocol revision 4 or higher. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>Verify Checklist</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the BIBB SCHED-I-B is not claimed in the checklist
	<b>Testing Hints</b>	
<b>135.1-2013r - 7.3.2.23.X - Forbid Duplicate Time Values</b>		
	<b>Test Conditionality</b>	If Protocol_Revision < 16, then this test shall be skipped.
	<b>Test Directives</b>	Apply to a writable Weekly Schedule property of a schedule object
	<b>Testing Hints</b>	

### 6.6.2 Priority\_For\_Writing is Writable

The Priority\_For\_Writing property in the schedule object shall be writable.

Verify EPICS		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the Priority_For_Writing property is writable in the IUT.
	<b>Testing Hints</b>	

### 6.6.3 Exception\_Schedule Property Shall Not Be Present

The IUT must not contain the Exception\_Schedule property

Verify EPICS		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT does not contain the Exception_Schedule property.
	<b>Testing Hints</b>	

### 6.6.4 Supports DM-TS-B

The IUT supports DM-TS-B.

Verify Checklist		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support for DM-TS-B.
	<b>Testing Hints</b>	

### 6.6.5 Supports DM-UTC-B

The IUT supports DM-UTC-B.

Verify Checklist		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support for DM-UTC-B.
	<b>Testing Hints</b>	

### 6.6.6 Supports Configurable Effective\_Period

The IUT supports the Effective\_Period property and it is configurable.

135.1-2013 - 7.3.2.23.1 - Effective_Period Test		
	<b>Test Conditionality</b>	This test shall be executed if and only if the IUT is prior to protocol revision 4. If the IUT is of the correct protocol revision and the vendor has selected this optional functionality, this test may not be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
135.1-2013 - 7.3.2.23.10.1 - Revision 4 Effective_Period Test		
	<b>Test Conditionality</b>	This test shall be executed if and only if the IUT is protocol revision 4 or higher. If the IUT is of the correct protocol revision and the vendor has selected this optional functionality, this test may not be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
BTL - 7.2.X7 - BACnetDateRange Non-Pattern Properties Test		
	<b>Test Conditionality</b>	This test shall only be applied to devices claiming Protocol_Revision 11 or higher.
	<b>Test Directives</b>	Apply to the Effective_Period property.

	<b>Testing Hints</b>	
<b>BTL - 7.2.X8 - BACnetDateRange Open-Ended Pattern Properties Test</b>		
	<b>Test Conditionality</b>	This test shall only be applied to devices claiming Protocol_Revision 11 or higher.
	<b>Test Directives</b>	Apply to the Effective Period property.
	<b>Testing Hints</b>	
<b>BTL - 9.23.2.X12 - BACnetDateRange Non-Pattern Properties Test using WritePropertyMultiple Service</b>		
	<b>Test Conditionality</b>	This test shall only be applied to devices claiming Protocol_Revision 11 or higher and which supports execution of WritePropertyMultiple.
	<b>Test Directives</b>	Apply to the Effective Period property.
	<b>Testing Hints</b>	

### 6.6.7 Is Able to Schedule Values of Different Datatypes (not simultaneously)

The IUT supports Schedule objects that are capable of being configured to schedule values of different datatypes. An example would be a Schedule object that was originally configured to write values of datatype REAL to an Analog Value object's Present\_Value property, but is later reconfigured to write Enumerated values to a Binary Value object's Present\_Value.

Such a reconfiguration involves changing the List\_Of\_Object\_Property\_References property to refer to properties of a specific datatype, and changing the 'value' members of the BACnetTimeValue pairs in the Weekly\_Schedule and Exception\_Schedule properties to have values in the same datatype as the properties referenced by the List\_Of\_Object\_Property\_References property.

<b>BTL - 7.3.2.23.11.1 - Internally Written Datatypes Test, non-NULL values</b>		
	<b>Test Conditionality</b>	Must be executed. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.23.11.2 - Internally Written Datatypes Test, NULL values and Priority Arrays</b>		
	<b>Test Conditionality</b>	Must be executed. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 6.6.8 Is Able to Schedule NULL Values

The IUT supports a Schedule object that is able to schedule (write) NULL with a priority.

<b>135.1-2013 - 7.3.2.23.11.2 - Internally Written Datatypes Test, NULL values and Priority Arrays</b>		
	<b>Test Conditionality</b>	Must be executed. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 6.6.9 Is Able to Schedule BOOLEAN Values

The IUT supports a Schedule object that is able to schedule (write) BOOLEAN values.

<b>BTL - 7.3.2.23.11.1 - Internally Written Datatypes Test, non-NULL values</b>		
	<b>Test Conditionality</b>	Must be executed. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.

	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 6.6.10Is Able to Schedule Unsigned Values

The IUT supports a Schedule object that is able to schedule (write) Unsigned values.

<b>BTL - 7.3.2.23.11.1 - Internally Written Datatypes Test, non-NULL values</b>		
	<b>Test Conditionality</b>	Must be executed. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 6.6.11Is Able to Schedule INTEGER (Signed) Values

The IUT supports a Schedule object that is able to schedule (write) INTEGER values.

<b>BTL - 7.3.2.23.11.1 - Internally Written Datatypes Test, non-NULL values</b>		
	<b>Test Conditionality</b>	Must be executed. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 6.6.12Is Able to Schedule REAL Values

The IUT supports a Schedule object that is able to schedule (write) REAL values.

<b>BTL - 7.3.2.23.11.1 - Internally Written Datatypes Test, non-NULL values</b>		
	<b>Test Conditionality</b>	Must be executed. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 6.6.13Is Able to Schedule Double Values

The IUT supports a Schedule object that able to schedule (write) Double values.

<b>BTL - 7.3.2.23.11.1 - Internally Written Datatypes Test, non-NULL values</b>		
	<b>Test Conditionality</b>	Must be executed. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 6.6.14Is Able to Schedule Octet String Values

The IUT supports a Schedule object that is able to schedule (write) Octet String values.

<b>BTL - 7.3.2.23.11.1 - Internally Written Datatypes Test, non-NULL values</b>		
	<b>Test Conditionality</b>	Must be executed. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 6.6.15Is Able to Schedule Character String Values

The IUT supports a Schedule object that is able to schedule (write) Character String values.

<b>BTL - 7.3.2.23.11.1 - Internally Written Datatypes Test, non-NULL values</b>		
---	--	--

	<b>Test Conditionality</b>	Must be executed. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 6.6.16Is Able to Schedule Bit String Values

The IUT supports a Schedule object that is able to schedule (write) Bit String values.

<b>BTL - 7.3.2.23.11.1 - Internally Written Datatypes Test, non-NULL values</b>		
	<b>Test Conditionality</b>	Must be executed. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 6.6.17Is Able to Schedule Enumerated Values

The IUT supports a Schedule object that is able to schedule (write) Enumerated values.

<b>BTL - 7.3.2.23.11.1 - Internally Written Datatypes Test, non-NULL values</b>		
	<b>Test Conditionality</b>	Must be executed. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 6.6.18Is Able to Schedule Date Values

The IUT supports a Schedule object that is able to schedule (write) Date values.

<b>BTL - 7.3.2.23.11.1 - Internally Written Datatypes Test, non-NULL values</b>		
	<b>Test Conditionality</b>	Must be executed. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 6.6.19Is Able to Schedule Time Values

The IUT supports a Schedule object that is able to write schedule (write) Time values.

<b>BTL - 7.3.2.23.11.1 - Internally Written Datatypes Test, non-NULL values</b>		
	<b>Test Conditionality</b>	Must be executed. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 6.6.20Is Able to Schedule BACnetObjectIdentifier Values

The IUT supports a Schedule object that is able to schedule (write) BACnetObjectIdentifier values.

<b>BTL - 7.3.2.23.11.1 - Internally Written Datatypes Test, non-NULL values</b>		
	<b>Test Conditionality</b>	Must be executed. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

## 6.7 Scheduling - Readonly - B

### 6.7.1 Base Requirements

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

<b>135.1-2013 - 7.3.2.23.1 - Effective Period Test</b>		
	<b>Test Conditionality</b>	This test shall be executed if and only if the IUT is prior to protocol revision 4. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.23.10.1 - Revision 4 Effective Period Test</b>		
	<b>Test Conditionality</b>	This test shall be executed if and only if the IUT is protocol revision 4 or higher. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.23.7 - List Of Object Property Reference Internal Test</b>		
	<b>Test Conditionality</b>	This test shall be executed if and only if the IUT is prior to protocol revision 4. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.23.10.7 - Revision 4 List Of Object Property Reference Internal Test</b>		
	<b>Test Conditionality</b>	This test shall be executed if and only if the IUT is protocol revision 4 or higher. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.23.10.3.13 - Revision 4 Schedule Default Test</b>		
	<b>Test Conditionality</b>	As per <i>ASHRAE 135.1-2013</i> if and only if the IUT is protocol revision 4 or higher. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.23.12 - Revision 4 Midnight Evaluation Test</b>		
	<b>Test Conditionality</b>	This test shall be executed if and only if the IUT is protocol revision 4 or higher. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>Verify Checklist</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT does not claim Weekly_Schedule or Exception_Schedule to be writable in the EPICS.
	<b>Testing Hints</b>	
<b>Verify Checklist</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the BIBB SCHED-I-B is not claimed in the checklist
	<b>Testing Hints</b>	

### 6.7.2 Supports DM-TS-B

The IUT supports DM-TS-B.

Verify Checklist		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support for DM-TS-B.
	<b>Testing Hints</b>	

### 6.7.3 Supports DM-UTC-B

The IUT supports DM-UTC-B.

Verify Checklist		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support for DM-UTC-B.
	<b>Testing Hints</b>	

### 6.7.4 Can Be Made to Contain a Schedule That Schedules BOOLEAN Values

The IUT can be configured to contain a Schedule object that is able to schedule (write) BOOLEAN values.

BTL - 7.3.2.23.11.1 - Internally Written Datatypes Test, non-NULL values		
	<b>Test Conditionality</b>	Must be executed. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 6.7.5 Can Be Made to Contain a Schedule That Schedules Unsigned Values

The IUT can be configured to contain a Schedule object that is able to schedule (write) Unsigned values.

BTL - 7.3.2.23.11.1 - Internally Written Datatypes Test, non-NULL values		
	<b>Test Conditionality</b>	Must be executed. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 6.7.6 Can Be Made to Contain a Schedule that Schedules INTEGER (signed) Values

The IUT can be configured to contain a Schedule object that is able to schedule (write) INTEGER (signed) values.

BTL - 7.3.2.23.11.1 - Internally Written Datatypes Test, non-NULL values		
	<b>Test Conditionality</b>	Must be executed. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 6.7.7 Can Be Made to Contain a Schedule that Schedules Real Values

The IUT can be configured to contain a Schedule object that is able to schedule (write) Real values.

<b>BTL - 7.3.2.23.11.1 - Internally Written Datatypes Test, non-NULL values</b>		
	<b>Test Conditionality</b>	Must be executed. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 6.7.8 Can Be Made to Contain a Schedule that Schedules Double Values

The IUT can be configured to contain a Schedule object that is able to schedule (write) Double values.

<b>BTL - 7.3.2.23.11.1 - Internally Written Datatypes Test, non-NULL values</b>		
	<b>Test Conditionality</b>	Must be executed. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 6.7.9 Can Be Made to Contain a Schedule that Schedules Octet String Values

The IUT can be configured to contain a Schedule object that is able to schedule (write) Octet String values.

<b>BTL - 7.3.2.23.11.1 - Internally Written Datatypes Test, non-NULL values</b>		
	<b>Test Conditionality</b>	Must be executed. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 6.7.10 Can Be Made to Contain a Schedule That Schedules Character String Values

The IUT can be configured to contain a Schedule object that is able to schedule (write) Character String values.

<b>BTL - 7.3.2.23.11.1 - Internally Written Datatypes Test, non-NULL values</b>		
	<b>Test Conditionality</b>	Must be executed. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 6.7.11 Can Be Made to Contain a Schedule That Schedules Bit String Values

The IUT can be configured to contain a Schedule object that is able to schedule (write) Bit String values.

<b>BTL - 7.3.2.23.11.1 - Internally Written Datatypes Test, non-NULL values</b>		
	<b>Test Conditionality</b>	Must be executed. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	



### 6.7.12 Can Be Made to Contain a Schedule That Schedules Enumerated Values

The IUT can be configured to contain a Schedule object that is able to schedule (write) Enumerated values.

BTL - 7.3.2.23.11.1 - Internally Written Datatypes Test, non-NULL values		
	Test Conditionality	Must be executed. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	Test Directives	
	Testing Hints	

### 6.7.13 Can Be Made to Contain a Schedule That Schedules Date Values

The IUT can be configured to contain a Schedule object that is able to schedule (write) Date values.

BTL - 7.3.2.23.11.1 - Internally Written Datatypes Test, non-NULL values		
	Test Conditionality	Must be executed. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	Test Directives	
	Testing Hints	

### 6.7.14 Can Be Made to Contain a Schedule That Schedules Time Values

The IUT can be configured to contain a Schedule object that is able to schedule (write) Time values.

BTL - 7.3.2.23.11.1 - Internally Written Datatypes Test, non-NULL values		
	Test Conditionality	Must be executed. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	Test Directives	
	Testing Hints	

### 6.7.15 Can Be Made to Contain a Schedule That Schedules BACnetObjectIdentifier Values

The IUT can be configured to contain a Schedule object that is able to schedule (write) BACnetObjectIdentifier values.

BTL - 7.3.2.23.11.1 - Internally Written Datatypes Test, non-NULL values		
	Test Conditionality	Must be executed. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	Test Directives	
	Testing Hints	

### 6.7.16 Can Be Made to Contain a Non-empty Weekly\_Schedule Property

The IUT can be configured to contain a Schedule object that has a non-empty Weekly\_Schedule property.

135.1-2013 - 7.3.2.23.2 - Weekly Schedule Property Test		
	Test Conditionality	Must be executed if and only if the IUT is prior to protocol revision 4. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	Test Directives	
	Testing Hints	
135.1-2013 - 7.3.2.23.10.2 - Revision 4 Weekly_Schedule Property Test		

	<b>Test Conditionality</b>	Must be executed if and only if the IUT is protocol revision 4 or higher. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.23.6 - Weekly Schedule Restoration Test</b>		
	<b>Test Conditionality</b>	If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 6.7.17 Can Be Made to Contain a Non-empty Exception\_Schedule

The IUT can be configured to contain a Schedule object that contains a non-empty Exception\_Schedule property.

<b>135.1-2013 - 7.3.2.23.3.9 - List of BACnetTimeValue Test</b>		
	<b>Test Conditionality</b>	Must be executed. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.23.10.3.9 - Revision 4 List of BACnetTimeValue Test</b>		
	<b>Test Conditionality</b>	Must be executed if and only if the IUT is protocol revision 4 or higher. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.23.5 - Exception_Schedule Restoration Test</b>		
	<b>Test Conditionality</b>	If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.23.10.3.12 - Revision 4 Lower Event Priority Change Test</b>		
	<b>Test Conditionality</b>	Must be executed if and only if the IUT is protocol revision 4 or higher. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 6.7.18 Can Be Made to Contain an Exception\_Schedule Property with a Calendar Reference

The IUT can be configured to contain a Schedule object that has an Exception\_Schedule that contains a calendar reference.

<b>135.1-2013 - 7.3.2.23.3.1 - Calendar Reference Test</b>		
	<b>Test Conditionality</b>	Must be executed if and only if the IUT is prior to protocol revision 4. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.23.10.3.1 - Revision 4 Calendar Reference Test</b>		
	<b>Test Conditionality</b>	Must be executed if and only if the IUT is protocol revision 4 or higher. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 6.7.19 Can Be Made to Contain an Exception\_Schedule Property with a WeekNDay

The IUT can be configured to contain a Schedule object that has an Exception\_Schedule that contains a calendar entry that uses a WeekNDay.

<b>135.1-2013 - 7.3.2.23.3.4 - Calendar Entry WeekNDay Month Test</b>		
	<b>Test Conditionality</b>	Must be executed if and only if the IUT is prior to protocol revision 4. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.23.10.3.4 - Revision 4 Calendar Entry WeekNDay Month Test</b>		
	<b>Test Conditionality</b>	Must be executed (see clause 7.3.2.22.X.3) if and only if the IUT is protocol revision 4 or higher. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.23.3.5 - Calendar Entry WeekNDay Week Of Month Test</b>		
	<b>Test Conditionality</b>	Must be executed if and only if the IUT is prior to protocol revision 4. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.23.10.3.5 - Revision 4 Calendar Entry WeekNDay Week Of Month Test</b>		
	<b>Test Conditionality</b>	Must be executed (see clause 7.3.2.22.X.3) if and only if the IUT is protocol revision 4 or higher. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.23.3.6 - Calendar Entry WeekNDay Last Week Of Month Test</b>		
	<b>Test Conditionality</b>	Must be executed if and only if the IUT is prior to protocol revision 4. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.23.10.3.6 - Revision 4 Calendar Entry WeekNDay Last Week Of Month Test</b>		
	<b>Test Conditionality</b>	Must be executed if and only if the IUT is protocol revision 4 or higher. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.23.3.7 - Calendar Entry WeekNDay Day Of Week Test</b>		
	<b>Test Conditionality</b>	Must be executed if and only if the IUT is prior to protocol revision 4. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.23.10.3.7 - Revision 4 Calendar Entry WeekNDay Day Of Week Test</b>		
	<b>Test Conditionality</b>	Must be executed if and only if the IUT is protocol revision 4 or higher. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.

	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.23.10.3.10 - Revision 4 Calendar Entry WeekNDay Odd-Numbered Month Test</b>		
	<b>Test Conditionality</b>	Must be executed if and only if the IUT is protocol revision 4 or higher. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.23.10.3.11 - Revision 4 Calendar Entry WeekNDay Even-Numbered Month Test</b>		
	<b>Test Conditionality</b>	Must be executed if and only if the IUT is protocol revision 4 or higher. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 6.7.20 Can Be Made to Contain an Exception\_Schedule Property with a DateRange

The IUT can be configured to contain a Schedule object that has an Exception\_Schedule that contains a calendar entry that uses a DateRange.

<b>135.1-2013 - 7.3.2.23.3.3 - Calendar Entry DateRange Test</b>		
	<b>Test Conditionality</b>	Must be executed if and only if the IUT is prior to protocol revision 4. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.23.10.3.3 - Revision 4 Calendar Entry DateRange Test</b>		
	<b>Test Conditionality</b>	Must be executed if and only if the IUT is protocol revision 4 or higher. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 6.7.21 Can Be Made to Contain an Exception\_Schedule Property with a Date

The IUT can be configured to contain a Schedule object that has an Exception\_Schedule that contains a Calendar entry that uses a Date.

<b>135.1-2013 - 7.3.2.23.3.2 - Calendar Entry Date Test</b>		
	<b>Test Conditionality</b>	Must be executed if and only if the IUT is prior to protocol revision 4. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.23.10.3.2 - Revision 4 Calendar Entry Date Test</b>		
	<b>Test Conditionality</b>	Must be executed if and only if the IUT is protocol revision 4 or higher. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 6.7.22 Can Be Made to Contain a Schedule That Schedules NULL Values

The IUT can be configured to contain a Schedule object that schedules NULL values.

<b>135.1-2013 - 7.3.2.23.11.2 - Internally Written Datatypes Test, NULL values and Priority Arrays</b>		
	<b>Test Conditionality</b>	Must be executed. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 6.7.23 Can Be Made to Contain an Exception\_Schedule Property with Two or More BACnetSpecialEvents Entries

The IUT can be configured to contain an Exception\_Schedule that contains 2 or more BACnetSpecialEvent entries.

<b>135.1-2013 - 7.3.2.23.3.8 - Event Priority Test</b>		
	<b>Test Conditionality</b>	Must be executed if and only if the IUT is prior to protocol revision 4. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.23.10.3.8 - Revision 4 Event Priority Test</b>		
	<b>Test Conditionality</b>	Must be executed if and only if the IUT is protocol revision 4 or higher. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 6.7.24 Supports Concurrent Weekly and Exception Schedules

The IUT supports Schedule objects that support both Weekly and Exception Schedules concurrently.

<b>135.1-2013 - 7.3.2.23.4 - Weekly Schedule and Exception Schedule Interaction Test</b>		
	<b>Test Conditionality</b>	This test shall be executed if and only if the IUT is prior to protocol revision 4. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.23.10.4 - Revision 4 Weekly Schedule and Exception Schedule Interaction Test</b>		
	<b>Test Conditionality</b>	This test shall be executed if and only if the IUT is protocol revision 4 or higher. If it is impossible to configure the IUT in the manner specified by this test, the test shall be omitted.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

---

## **6.8 Scheduling - Schedule - A**

---

### **6.8.1 Base Requirements**

Base requirements must be met by any IUT claiming conformance to this BIBB. There are no base requirements tests for this section.

---

## **6.9 Scheduling - Timer - I - B**

---

### **6.9.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

---

## **6.10 Scheduling - Timer - E - B**

---

### **6.10.1 Base Requirements**

Contact BTL for interim tests for this BIBB.



---

## **7 Trending BIBBs**

---

## 7.1 Trending - Viewing - A

### 7.1.1 Base Requirements

Base requirements must be met by any IUT claiming conformance to this BIBB. There are no base requirements tests for this section.

### 7.1.2 Initiates ReadRange

The IUT is able to initiate the ReadRange service.

Verify Checklist		
	Test Conditionality	Must be executed.
	Test Directives	Verify that the IUT claims support for Initiates ReadRange in the Checklist.
	Testing Hints	

### 7.1.3 Interoperates with Trend Logs

The IUT can interoperate with Trend Log objects.

BTL - 8.21.9 - Presents Log Records		
	Test Conditionality	Must be executed
	Test Directives	This test shall be executed with a reference server device containing a Trend Log object. Repeat the test for Real, INTEGER, BOOLEAN, Bit-String, Enumerated, and NULL datatypes.
	Testing Hints	Schedule_Default and Present_Value of the Schedule Object, Alarm_Values and Fault_Values of the CharacterString Value Object and Low_Diff_Limit in the Loop Object are standard properties that should accept a written NULL.

### 7.1.4 Interoperates with Trend Log Multiple Objects

The IUT can interoperate with Trend Log Multiple objects in devices claiming Protocol\_Revision 7 or higher.

BTL - 8.21.9 - Presents Log Records		
	Test Conditionality	Must be executed
	Test Directives	This test shall be executed with a reference server device containing a Trend Log Multiple object claiming Protocol_Revision 7 or higher. Execute test against Trend Log Multiple that contains all required datatypes (Boolean, Real, Enumerated, Unsigned32, Integer32, Bit String, and NULL).
	Testing Hints	Schedule_Default and Present_Value of the Schedule Object, Alarm_Values and Fault_Values of the CharacterString Value Object and Low_Diff_Limit in the Loop Object are standard properties that should accept a written NULL.

### 7.1.5 Supports Reading Items by Time with a Positive Count

The IUT can initiate one or more ReadRange requests that specify the Time form and a Positive 'Count' that access a tester-specified portion of log records.

135.1-2013 - 8.21.8 - Reading a Range of Items Using Any Valid Range		
	Test Conditionality	Must be executed
	Test Directives	This test shall be executed with a reference server device claiming Protocol_Revision 3 or higher for Trend Logs and Protocol_Revision 7 or higher for Trend Log Multiple. Perform the test on the Trend Log

		object and repeat on Trend Log Multiple object if Protocol_Revision 7 or higher is supported.
	<b>Testing Hints</b>	The 'Range' value in Steps 2 and 4 must include at least one occurrence of the Time form of the ReadRange service, with a Positive 'Count'. The IUT shall be capable of presenting any portion of records selected by the tester. The set of records in this test shall contain at least one log-status entry.

### 7.1.6 Supports Reading Items by Time with a Negative Count

The IUT can initiate one or more ReadRange requests that specify the Time form and a Negative 'Count' that access a tester-specified portion of log records.

135.1-2013 - 8.21.8 - Reading a Range of Items Using Any Valid Range		
	<b>Test Conditionality</b>	Must be executed
	<b>Test Directives</b>	This test shall be executed with a reference server device claiming Protocol_Revision 3 or higher for Trend Logs and Protocol_Revision 7 or higher for Trend Log Multiple. Perform the test on the Trend Log object and repeat on Trend Log Multiple object if Protocol_Revision 7 or higher is supported.
	<b>Testing Hints</b>	The 'Range' value in Steps 2 and 4 must include at least one occurrence of the Time form of the ReadRange service, with a Negative 'Count'. The IUT shall be capable of presenting any portion of records selected by the tester. The set of records in this test shall contain at least one log-status entry.

### 7.1.7 Supports Reading Items by Position with a Positive Count

The IUT can initiate one or more ReadRange requests that specify the Position form and a Positive 'Count' that access a tester-specified portion of log records.

135.1-2013 - 8.21.8 - Reading a Range of Items Using Any Valid Range		
	<b>Test Conditionality</b>	Must be executed
	<b>Test Directives</b>	This test shall be executed with a reference server device claiming Protocol_Revision 3 or higher for Trend Logs and Protocol_Revision 7 or higher for Trend Log Multiple. Perform the test on the Trend Log object and repeat on Trend Log Multiple object if Protocol_Revision 7 or higher is supported.
	<b>Testing Hints</b>	The 'Range' value in Steps 2 and 4 must include at least one occurrence of the Position form of the ReadRange service, with a Positive 'Count'. The IUT shall be capable of presenting any portion of records selected by the tester. The set of records in this test shall contain at least one log-status entry.

### 7.1.8 Supports Reading Items by Position with a Negative Count

The IUT can initiate one or more ReadRange requests that specify the Position form and a Negative 'Count' that access a tester-specified portion of log records.

135.1-2013 - 8.21.8 - Reading a Range of Items Using Any Valid Range		
	<b>Test Conditionality</b>	Must be executed
	<b>Test Directives</b>	This test shall be executed with a reference server device claiming Protocol_Revision 3 or higher for Trend Logs and Protocol_Revision 7 or higher for Trend Log Multiple. Perform the test on the Trend Log object and repeat on Trend Log Multiple object if Protocol_Revision 7 or higher is supported.

	<b>Testing Hints</b>	The 'Range' value in Steps 2 and 4 must include at least one occurrence of the Position form of the ReadRange service, with a Negative 'Count'. The IUT shall be capable of presenting any portion of records selected by the tester. The set of records in this test shall contain at least one log-status entry.
--	----------------------	---

### 7.1.9 Supports Reading Items by Sequence Number with a Positive Count

The IUT can initiate one or more ReadRange requests that specify the Sequence Number form and a Positive 'Count' that access a tester-specified portion of log records.

135.1-2013 - 8.21.8 - Reading a Range of Items Using Any Valid Range		
	<b>Test Conditionality</b>	Must be executed
	<b>Test Directives</b>	This test shall be executed with a reference server device claiming Protocol_Revision 3 or higher for Trend Logs and Protocol_Revision 7 or higher for Trend Log Multiple. Perform the test on the Trend Log object and repeat on Trend Log Multiple object if Protocol_Revision 7 or higher is supported.
	<b>Testing Hints</b>	The 'Range' value in Steps 2 and 4 must include at least one occurrence of the Sequence Number form of the ReadRange service, with a Positive 'Count'. The IUT shall be capable of presenting any portion of records selected by the tester. The set of records in this test shall contain at least one log-status entry.

### 7.1.10 Supports Reading Items by Sequence Number with a Negative Count

The IUT can initiate one or more ReadRange requests that specify the Sequence Number form and a Negative 'Count' that access a tester-specified portion of log records.

135.1-2013 - 8.21.8 - Reading a Range of Items Using Any Valid Range		
	<b>Test Conditionality</b>	Must be executed
	<b>Test Directives</b>	This test shall be executed with a reference server device claiming Protocol_Revision 3 or higher for Trend Logs and Protocol_Revision 7 or higher for Trend Log Multiple. Perform the test on the Trend Log object and repeat on Trend Log Multiple object if Protocol_Revision 7 or higher is supported.
	<b>Testing Hints</b>	The 'Range' value in Steps 2 and 4 must include at least one occurrence of the Sequence Number form of the ReadRange service, with a Negative 'Count'. The IUT shall be capable of presenting any portion of records selected by the tester. The set of records in this test shall contain at least one log-status entry.

### 7.1.11 Supports Reading Items with no Range

The IUT can initiate one or more ReadRange requests that specify no range that access a tester-specified portion of log records.

135.1-2013 - 8.21.8 - Reading a Range of Items Using Any Valid Range		
	<b>Test Conditionality</b>	Must be executed
	<b>Test Directives</b>	This test shall be executed with a reference server device claiming Protocol_Revision 3 or higher for Trend Logs and Protocol_Revision 7 or higher for Trend Log Multiple. Perform the test on the Trend Log

		object and repeat on Trend Log Multiple object if Protocol_Revision 7 or higher is supported.
	<b>Testing Hints</b>	The 'Range' value in Steps 2 and 4 specify no range The set of records in this test shall contain at least one log-status entry.

### 7.1.12 Is able to present Double datatypes in trend logging objects

The IUT can present optional datatypes.

<b>BTL - 8.21.9 - Presents Log Records</b>		
	<b>Test Conditionality</b>	Must be executed and repeated if IUT claims Protocol_Revision 7 or higher.
	<b>Test Directives</b>	For Trend Log objects, this test shall be executed with a reference server device claiming Protocol_Revision 3 or higher and for Trend Log Multiple objects, this test shall be executed with a reference server device claiming Protocol_Revision 7 or higher. For Protocol_Revision 3 or higher, perform the test using Double datatype and for Protocol_Revision 7 or higher, repeat the test using a Trend Log Multiple that contains a Double datatype.
	<b>Testing Hints</b>	

### 7.1.13 Is able to present Octet String datatypes in trend logging objects

The IUT can present optional datatypes.

<b>BTL - 8.21.9 - Presents Log Records</b>		
	<b>Test Conditionality</b>	Must be executed and repeated if IUT claims Protocol_Revision 7 or higher.
	<b>Test Directives</b>	For Trend Log objects, this test shall be executed with a reference server device claiming Protocol_Revision 3 or higher and for Trend Log Multiple objects, this test shall be executed with a reference server device claiming Protocol_Revision 7 or higher. For Protocol_Revision 3 or higher, perform the test using Octet String datatype and for Protocol_Revision 7 or higher, repeat the test using a Trend Log Multiple that contains a Octet String datatype.
	<b>Testing Hints</b>	

### 7.1.14 Is able to present Character String datatypes in trend logging objects

The IUT can present optional datatypes.

<b>BTL - 8.21.9 - Presents Log Records</b>		
	<b>Test Conditionality</b>	Must be executed and repeated if IUT claims Protocol_Revision 7 or higher.
	<b>Test Directives</b>	For Trend Log objects, this test shall be executed with a reference server device claiming Protocol_Revision 3 or higher and for Trend Log Multiple objects, this test shall be executed with a reference server device claiming Protocol_Revision 7 or higher. For Protocol_Revision 3 or higher, perform the test using Character String datatype and for Protocol_Revision 7 or higher, repeat the test using a Trend Log Multiple that contains a Character String datatype.
	<b>Testing Hints</b>	

### 7.1.15 Is able to present Date datatypes in trend logging objects

The IUT can present optional datatypes.

<b>BTL - 8.21.9 - Presents Log Records</b>		
	<b>Test Conditionality</b>	Must be executed and repeated if IUT claims Protocol_Revision 7 or higher.
	<b>Test Directives</b>	For Trend Log objects, this test shall be executed with a reference server device claiming Protocol_Revision 3 or higher and for Trend Log Multiple objects, this test shall be executed with a reference server device claiming Protocol_Revision 7 or higher. For Protocol_Revision 3 or higher, perform the test using Date datatype and for Protocol_Revision 7 or higher, repeat the test using a Trend Log Multiple that contains a Date datatype.
	<b>Testing Hints</b>	

### 7.1.16Is able to present Time datatypes in trend logging objects

The IUT can present optional datatypes.

<b>BTL - 8.21.9 - Presents Log Records</b>		
	<b>Test Conditionality</b>	Must be executed and repeated if IUT claims Protocol_Revision 7 or higher.
	<b>Test Directives</b>	For Trend Log objects, this test shall be executed with a reference server device claiming Protocol_Revision 3 or higher and for Trend Log Multiple objects, this test shall be executed with a reference server device claiming Protocol_Revision 7 or higher. For Protocol_Revision 3 or higher, perform the test using Time datatype and for Protocol_Revision 7 or higher, repeat the test using a Trend Log Multiple that contains a Time datatype.
	<b>Testing Hints</b>	

### 7.1.17Is able to present BACnetObjectIdentifier datatypes in trend logging objects

The IUT can present optional datatypes.

<b>BTL - 8.21.9 - Presents Log Records</b>		
	<b>Test Conditionality</b>	Must be executed and repeated if IUT claims Protocol_Revision 7 or higher.
	<b>Test Directives</b>	For Trend Log objects, this test shall be executed with a reference server device claiming Protocol_Revision 3 or higher and for Trend Log Multiple objects, this test shall be executed with a reference server device claiming Protocol_Revision 7 or higher. For Protocol_Revision 3 or higher, perform the test using BACnet Object Identifier datatype and for Protocol_Revision 7 or higher, repeat the test using a Trend Log Multiple that contains a BACnet Object Identifier datatype.
	<b>Testing Hints</b>	

## 7.2 Trending - Advanced View and Modify - A

### 7.2.1 Base Requirements

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

<b>BTL - 8.21.9 - Presents Log Records</b>		
	<b>Test Conditionality</b>	Must be executed
	<b>Test Directives</b>	This test shall be executed with a reference server device claiming Protocol_Revision 3 or higher. Repeat the test for BOOLEAN, Real, Enumerated, Unsigned32, Integer23, Bit String, and Null datatypes
	<b>Testing Hints</b>	Schedule_Default and Present_Value of the Schedule Object, Alarm_Values and Fault_Values of the CharacterString Value Object and Low_Diff_Limit in the Loop Object are standard properties that should accept a written NULL.
<b>135.1-2013 - 8.18.3 - Reading and Presenting Properties</b>		
	<b>Test Conditionality</b>	Must be executed
	<b>Test Directives</b>	
	<b>Testing Hints</b>	Repeat the test for a Trend Log, Trend Log Multiple, Event Enrollment, and Notification Class object. Repeat the test for each of the properties listed in the table in the BIBB definition.
<b>135.1-2013 - 8.22.4 - Accepting Input and Modifying Properties</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	Repeat the test for a Trend Log, Trend Log Multiple, Event Enrollment, and Notification Class object. Repeat the test for each of the properties listed in the table in the BIBB definition. Repeat the test for a variety of values that cover the range of values required by the BIBB.

### 7.2.2 Supports T-V-A

The IUT shall support T-V-A in order to display Trend Log and Trend Log Multiple objects to the user.

<b>Verify Checklist</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support for T-V-A.
	<b>Testing Hints</b>	

### 7.2.3 Supports DS-RP-A

The IUT shall support DS-RP-A in order to display Trend Log, Trend Log Multiple, Event Enrollment, and Notification Class property values to the user.

<b>Verify Checklist</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support for DS-RP-A.
	<b>Testing Hints</b>	

### 7.2.4 Supports DS-WP-A

The IUT shall support DS-WP-A in order to configure Trend Log, Trend Log Multiple, Event Enrollment, and NotificationClass property values modified by the user.

Verify Checklist		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support for DS-WP-A.
	<b>Testing Hints</b>	

### 7.2.5 Supports DM-OCD-A

The IUT shall support DM-OCD-A in order to create Trend Log, Trend Log Multiple, Event Enrollment, and Notification Class objects.

Verify Checklist		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support for DM-OCD-A and that it claims the ability to create and delete Trend Log, Trend Log Multiple, Event Enrollment, and Notification Class objects.
	<b>Testing Hints</b>	



## 7.3 Trending - View and Modify Trends - I - B

### 7.3.1 Base Requirements

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

<b>135.1-2013 - 9.21.1.1 - Reading All Items in the List</b>		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
<b>135.1-2013 - 7.3.2.24.1 - Enable Test</b>		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
<b>135.1-2013 - 7.3.2.24.6.1 - Stop When Full TRUE Test</b>		
	Test Conditionality	Only applicable if the Stop_When_Full property is configurable or equal to TRUE.
	Test Directives	
	Testing Hints	
<b>135.1-2013 - 7.3.2.24.6.2 - Stop When Full FALSE Test</b>		
	Test Conditionality	Only applicable if the Stop_When_Full property is configurable or equal to FALSE.
	Test Directives	
	Testing Hints	
<b>135.1-2013 - 7.3.2.24.7 - Buffer Size Test</b>		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
<b>135.1-2013 - 7.3.2.24.8 - Record Count Test</b>		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
<b>135.1-2013 - 7.3.2.24.9 - Total Record Count Test</b>		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
<b>BTL - 7.3.2.24.13 - Log-Status Test</b>		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
<b>BTL - 7.3.2.24.14 - Time Change Test</b>		
	Test Conditionality	If the Device does not support Local_Time property this test may be skipped.
	Test Directives	
	Testing Hints	

### 7.3.2 Supports all forms of ReadRange

The IUT can accept any of the ReadRange options and respond appropriately.

<b>135.1-2013 - 9.21.1.2 - Reading Items by Position with Positive Count</b>		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
<b>135.1-2013 - 9.21.1.3 - Reading Items by Position with Negative Count</b>		

	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.21.1.4 - Reading Items by Time</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.21.1.4.1 - Reading Items by Time with Negative Count</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.21.1.9 - Reading Items by Sequence with Positive Count</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.21.1.10 - Reading Items by Sequence with Negative Count</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.21.1.7 - Reading a Range of Items that do not Exist (by Sequence)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.21.1.8 - Reading a Range of Items that do not Exist (by Time)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.21.1.13 - Reading Items with Negative Count and MOREITEMS</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 7.3.3 Executes ReadRange

The IUT is able to execute the ReadRange service.

Verify Checklist		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support Executes ReadRange in the Checklist.
	<b>Testing Hints</b>	

### 7.3.4 Supports Periodic Logging (Polling)

The IUT can be made to gather trend data at a rate specified by the Log\_Interval property.

BTL - 7.3.2.24.4 - Log_Interval Test		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 7.3.5 Supports COV Logging of Local Properties

The IUT can be made to gather trend data based change of value of the local property.

<b>BTL - 7.3.2.24.15 - COV-Sampling Verification Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 7.3.6 Supports Triggered Logging

The IUT can be made to gather trend data using the Trigger property.

<b>BTL - 7.3.2.24.19 - Trigger Verification Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 7.3.7 Supports Start\_Time and Stop\_Time Properties

The IUT can be made to start and stop logging using these properties.

These properties are required to be present and writable in trend log objects that are trending a BACnet property.

<b>135.1-2013 - 7.3.2.24.2 - Start Time Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.24.3 - Stop Time Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.2.X6 - DateTime Non-Pattern Properties Test</b>		
	<b>Test Conditionality</b>	Must be executed if the IUT claims Protocol Revision 11 or greater.
	<b>Test Directives</b>	Apply to the Start_Time and again to the Stop_Time properties in a Trend Log object.
	<b>Testing Hints</b>	
<b>BTL - 9.23.2.X11 - DateTime Non-Pattern Properties Test using WritePropertyMultiple Service</b>		
	<b>Test Conditionality</b>	This test shall only be applied to devices claiming Protocol Revision 11 or higher and which supports execution of WritePropertyMultiple.
	<b>Test Directives</b>	Apply to the Start_Time and again to the Stop_Time properties in a Trend Log object.
	<b>Testing Hints</b>	

### 7.3.8 Supports Clock-aligned Logging

The IUT can be made to gather trend data with clock-aligned Timestamps.

<b>BTL - 7.3.2.24.X8 - Clock-aligned logging</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.24.X9 - Logging Interval Offset</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

## 7.4 Trending - View and Modify Trends - E - B

### 7.4.1 Base Requirements

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

This BIBB requires support of T-VMT-I-B, DS-RP-A.

This BIBB requires that the Log\_DeviceObjectProperty and Log\_Interval properties must be present and writable.

Verify Checklist		
	Test Conditionality	Must be executed
	Test Directives	Verify that the IUT claims support for T-VMT-I-B in the Checklist.
	Testing Hints	
Verify Checklist		
	Test Conditionality	Must be executed
	Test Directives	Verify that the IUT claims support for DS-RP-A in the Checklist.
	Testing Hints	
135.1-2013 - 9.21.1.12 - Status/Failure Logging		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	Tester may want to try other error conditions.
135.1-2013 - 7.3.2.24.16 - Interval Gathering of External Trends Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 7.4.2 Is Able to Trend REAL Values

The IUT can be made to trend REAL type properties.

135.1-2013 - 9.21.1.11 - Data Type Verification Test		
	Test Conditionality	Must be executed.
	Test Directives	Set Log_DeviceObjectProperty to an external property of type REAL.
	Testing Hints	

### 7.4.3 Is Able to Trend Unsigned Values

The IUT can be made to trend Unsigned type properties.

135.1-2013 - 9.21.1.11 - Data Type Verification Test		
	Test Conditionality	Must be executed.
	Test Directives	Set Log_DeviceObjectProperty to an external property of type UNSIGNED
	Testing Hints	

### 7.4.4 Is Able to Trend INTEGER (Signed) Values

The IUT can be made to trend Integer type properties.

135.1-2013 - 9.21.1.11 - Data Type Verification Test		
	Test Conditionality	Must be executed.
	Test Directives	Set Log_DeviceObjectProperty to an external property of type INTEGER.
	Testing Hints	

### 7.4.5 Is Able to Trend BOOLEAN Values

The IUT can be made to trend BOOLEAN type properties.

135.1-2013 - 9.21.1.11 - Data Type Verification Test		
	Test Conditionality	Must be executed.
	Test Directives	Set Log_DeviceObjectProperty to an external property of type BOOLEAN.
	Testing Hints	

### 7.4.6 Is Able to Trend Bit String Values

The IUT can be made to trend Bit String type properties.

135.1-2013 - 9.21.1.11 - Data Type Verification Test		
	Test Conditionality	Must be executed.
	Test Directives	Set Log_DeviceObjectProperty to an external property of type Bit String.
	Testing Hints	

### 7.4.7 Is Able to Trend Enumerated Values

The IUT can be made to trend Enumerated type properties.

135.1-2013 - 9.21.1.11 - Data Type Verification Test		
	Test Conditionality	Must be executed.
	Test Directives	Set Log_DeviceObjectProperty to an external property of type Enumerated.
	Testing Hints	

### 7.4.8 Is Able to Trend NULL Values

The IUT can be made to trend any type property that may change to a value of NULL.

135.1-2013 - 9.21.1.11 - Data Type Verification Test		
	Test Conditionality	Must be executed.
	Test Directives	Set Log_DeviceObjectProperty to an external property that can be set to NULL value.
	Testing Hints	The best way to do this is to trend an entry in a priority array. Schedule_Default and Present_Value of the Schedule Object, Alarm_Values and Fault_Values of the CharacterString Value Object and Low_Diff_Limit in the Loop Object, are standard properties that should accept a written NULL.

### 7.4.9 Is Able to Trend Double Values

The IUT can be made to trend Double type properties.

135.1-2013 - 9.21.1.11 - Data Type Verification Test		
	Test Conditionality	Must be executed.
	Test Directives	Set Log_DeviceObjectProperty to an external property of type Double.
	Testing Hints	

### 7.4.10 Is Able to Trend Character String Values

The IUT can be made to trend Character String type properties.

135.1-2013 - 9.21.1.11 - Data Type Verification Test		
	Test Conditionality	Must be executed.

	<b>Test Directives</b>	Set Log_DeviceObjectProperty to an external property of type Character String.
	<b>Testing Hints</b>	

### 7.4.11Is Able to Trend Octet String Values

The IUT can be made to trend Octet String type properties.

135.1-2013 - 9.21.1.11 - Data Type Verification Test		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Set Log_DeviceObjectProperty to an external property of type Octet String.
	<b>Testing Hints</b>	

### 7.4.12Is Able to Trend Date Values

The IUT can be made to trend Date type properties.

135.1-2013 - 9.21.1.11 - Data Type Verification Test		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Set Log_DeviceObjectProperty to an external property of type Date.
	<b>Testing Hints</b>	

### 7.4.13Is Able to Trend Time Values

The IUT can be made to trend Time type properties.

135.1-2013 - 9.21.1.11 - Data Type Verification Test		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Set Log_DeviceObjectProperty to an external property of type Time.
	<b>Testing Hints</b>	

### 7.4.14Is Able to Trend BACnetObjectIdentifier Values

The IUT can be made to trend BACnetObjectIdentifier type properties.

135.1-2013 - 9.21.1.11 - Data Type Verification Test		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Set Log_DeviceObjectProperty to an external property of type BACnetObjectIdentifier.
	<b>Testing Hints</b>	

### 7.4.15Supports COV Logging of Remote Property Values

The IUT can be made to use COV logging.

Requires that COV\_Resubscription\_Interval be present.

BTL - 7.3.1.7.X1 - COV_Resubscription_Interval Test		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

## 7.5 Trending - Automated Trend Retrieval - A

### 7.5.1 Base Requirements

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

135.1-2013 - 8.21.6 - Reading a Range of Items Using Any Valid Range in Response to ConfirmedEventNotifications of the Buffer_Ready Event Type		
	Test Conditionality	Must be executed
	Test Directives	
	Testing Hints	Repeat the test for each form of the 'Time Stamp' parameter - BACnetDateTime, Time, and Sequence Number. For each test, the values in Steps 2 and 7 shall take the appropriate form.

### 7.5.2 Initiates ReadRange

The IUT is able to initiate the ReadRange service.

Verify Checklist		
	Test Conditionality	Must be executed.
	Test Directives	Verify that the IUT claims support for Initiates ReadRange in the Checklist.
	Testing Hints	

### 7.5.3 Processes BUFFER\_READY Event Notifications from Devices Claiming Protocol\_Revision 3 or Higher

The IUT can process BUFFER\_READY event notifications from devices claiming Protocol\_Revision 3 or higher.

No Specific Test		
	Test Conditionality	
	Test Directives	
	Testing Hints	This functionality will be verified by the other tests in this section.

### 7.5.4 Processes ConfirmedEventNotification of BUFFER\_READY Events

The IUT can process ConfirmedEventNotification messages indicating an 'Event Type' of Buffer\_Ready, and respond with a series of ReadRange requests that access records from the Trend Log referenced in the notifications.

135.1-2013 - 8.21.6 - Reading a Range of Items Using Any Valid Range in Response to ConfirmedEventNotifications of the Buffer_Ready Event Type		
	Test Conditionality	Must be executed
	Test Directives	This test shall be executed with a reference server device claiming Protocol Revision 3 or higher.
	Testing Hints	

### 7.5.5 Processes UnconfirmedEventNotification of BUFFER\_READY Events

The IUT can process UnconfirmedEventNotification messages indicating an 'Event Type' of Buffer\_Ready, and respond with a series of ReadRange requests that access records from the Trend Log referenced in the notifications.

<b>135.1-2013 - 8.21.7 - Reading a Range of Items Using Any Valid Range in Response to UnconfirmedEventNotifications of the Buffer Ready Event Type</b>		
	<b>Test Conditionality</b>	Must be executed
	<b>Test Directives</b>	This test shall be executed with a reference server device claiming Protocol Revision 3 or higher.
	<b>Testing Hints</b>	



## 7.6 Trending - Automated Trend Retrieval - B

### 7.6.1 Base Requirements

Base requirements must be met by any IUT claiming conformance to this BIBB. There are no base requirements tests for this section.

### 7.6.2 Supports T-VMT-I-B

The IUT supports T-VMT-I-B.

Verify Checklist		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support for T-VMT-I-B in the Checklist.
	<b>Testing Hints</b>	

### 7.6.3 Supports the Notification Class Object

The IUT supports the Notification Class Object in order to send Buffer\_Ready notifications.

Verify Checklist		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support for the Notification Class Object in the Checklist.
	<b>Testing Hints</b>	

### 7.6.4 Implements Intrinsic Alarming

The IUT contains, or can be made to contain, a Trend Log object that can generate BUFFER\_READY notifications.

Each Trend Log object must include all of the Trend Log properties required for intrinsic reporting, i.e. Notification\_Threshold, Records\_Since\_Notification, Last\_Notify\_Record, Notification\_Class, Event\_Enable, Acked\_Transitions, Notify\_Type, & Event\_Time\_Stamps.

BTL - 7.3.1.10.2 - Event Enable Tests for TO NORMAL only Algorithms		
	<b>Test Conditionality</b>	If the IUT cannot be configured to meet the configuration requirements then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
135.1-2013 - 7.3.1.12 - Notify Type Test		
	<b>Test Conditionality</b>	If the IUT cannot be configured to meet the 135.1-2013 configuration requirements then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
135.1-2013 - 7.3.2.24.10 - Notification Threshold Test		
	<b>Test Conditionality</b>	If the IUT cannot be configured to meet the 135.1-2013 configuration requirements then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
135.1-2013 - 7.3.2.24.17 - Last Notify Record Test		
	<b>Test Conditionality</b>	If the IUT cannot be configured to meet the configuration requirements then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
135.1-2013 - 7.3.2.24.18 - Records Since Notification		

	<b>Test Conditionality</b>	If the IUT cannot be configured to meet the 135.1-2013 configuration requirements then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 8.4.7 - BUFFER READY Tests</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	The 'Event Object Identifier' in this test must be a Trend Log object contained in the IUT.
<b>135.1-2013 - 8.5.7 - BUFFER READY Tests</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	The 'Event Object Identifier' in this test must be a Trend Log object contained in the IUT.

### 7.6.5 Implements Algorithmic Alarming

The IUT contains, or can be made to contain an Event Enrollment object that can generate a Buffer\_Ready ConfirmedEventNotification.

<b>Verify Checklist</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support for the Event Enrollment object in the checklist.
	<b>Testing Hints</b>	
<b>135.1-2013 - 8.4.7 - BUFFER READY Tests</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	The 'Event Object Identifier' in this test must be an Event Enrollment object contained in the IUT.
<b>135.1-2013 - 8.5.7 - BUFFER READY Tests</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	The 'Event Object Identifier' in this test must be an Event Enrollment object contained in the IUT.

### 7.6.6 Generates Event Notifications with Timestamps of the BACnetDateTime Form

The IUT generates, or can be made to generate, ConfirmedEventNotifications with the Time Stamp parameter taking the BACnetDateTime form.

<b>135.1-2013 - 8.4 - ConfirmedEventNotification Service Initiation Tests</b>		
	<b>Test Conditionality</b>	If the IUT supports AE-N-I-B, these tests may be skipped. Any of the 8.4 tests can be used to ensure that the IUT properly generates ConfirmedEventNotification requests using the BACnetDateTime form. The specific tests that can be executed are detailed under the test cases for the specific algorithms. As long as one of the tests is executed using ConfirmedEventNotifications and the notification that is generated contains a timestamp of the BACnetDateTime form, then this test case shall be satisfied.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 7.6.7 Generates Event Notifications with Timestamps of the Time Form

The IUT generates, or can be made to generate, ConfirmedEventNotifications with the Time Stamp parameter taking the Time form.

135.1-2013 - 8.4 - ConfirmedEventNotification Service Initiation Tests		
	<b>Test Conditionality</b>	If the IUT supports AE-N-I-B, these tests may be skipped. Any of the 8.4 tests can be used to ensure that the IUT properly generates ConfirmedEventNotification requests using the Time form. The specific tests that can be executed are detailed under the test cases for the specific algorithms. As long as one of the tests is executed using ConfirmedEventNotifications and the notification that is generated contains a timestamp of the Time form, then this test case shall be satisfied.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 7.6.8 Generates Event Notifications with Timestamps of the Sequence Number Form

The IUT generates, or can be made to generate, ConfirmedEventNotifications with the Time Stamp parameter taking the Sequence Number form.

135.1-2013 - 8.4 - ConfirmedEventNotification Service Initiation Tests		
	<b>Test Conditionality</b>	If the IUT supports AE-N-I-B, these tests may be skipped. Any of the 8.4 tests can be used to ensure that the IUT properly generates ConfirmedEventNotification requests using the Sequence Number form. The specific tests that can be executed are detailed under the test cases for the specific algorithms. As long as one of the tests is executed using ConfirmedEventNotifications and the notification that is generated contains a timestamp of the Sequence Number form, then this test case shall be satisfied.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

## 7.7 Trending - View and Modify Multiple Values - I - B

### 7.7.1 Base Requirements

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

<b>135.1-2013 - 9.21.1.1 - Reading All Items in the List</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test shall be executed with a reference server device claiming Protocol_Revision 3 or higher.
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.24.1 - Enable Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	The test must be repeated using all supported logging types.
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.24.6.1 - Stop When Full TRUE Test</b>		
	<b>Test Conditionality</b>	Only applicable if this property is writable.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.24.6.2 - Stop When Full FALSE Test</b>		
	<b>Test Conditionality</b>	Only applicable if this property is writable.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.24.7 - Buffer Size Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.24.8 - Record Count Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.2.24.9 - Total Record Count Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.24.13 - Log-Status Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.24.14 - Time Change Test</b>		
	<b>Test Conditionality</b>	If the Device does not support Local_Time property this test may be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 7.7.2 Supports all forms of ReadRange

The IUT can accept any of the ReadRange options and respond appropriately.

<b>135.1-2013 - 9.21.1.2 - Reading Items by Position with Positive Count</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.21.1.3 - Reading Items by Position with Negative Count</b>		

	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.21.1.4 - Reading Items by Time</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.21.1.4.1 - Reading Items by Time with Negative Count</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.21.1.9 - Reading Items by Sequence with Positive Count</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.21.1.10 - Reading Items by Sequence with Negative Count</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.21.1.7 - Reading a Range of Items that do not Exist (by Sequence)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.21.1.8 - Reading a Range of Items that do not Exist (by Time)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 7.7.3 Executes ReadRange

The IUT is able to execute the ReadRange service.

<b>Verify Checklist</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support for Executes ReadRange in the Checklist.
	<b>Testing Hints</b>	

### 7.7.4 Supports Periodic Logging (Polling)

The IUT can be made to gather trend data at a rate specified by the Log\_Interval property.

<b>BTL - 7.3.2.24.4 - Log Interval Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 7.7.5 Supports Triggered Logging

The IUT can be made to gather trend data using the Trigger property.

<b>BTL - 7.3.2.24.19 - Trigger Verification Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 7.7.6 Supports Clock-aligned Logging

The IUT can be made to gather trend data with clock-aligned Timestamps.

<b>BTL - 7.3.2.24.X8 - Clock-aligned logging</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.3.2.24.X9 - Logging Interval Offset</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 7.7.7 Supports Start\_Time and Stop\_Time Properties

The IUT can be made to start and stop logging using these properties.

If present these properties are required to be writable.

<b>135.1-2013 - 7.3.2.24.2 - Start Time Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 7.2.X6 - DateTime Non-Pattern Properties Test</b>		
	<b>Test Conditionality</b>	Must be executed if the IUT claims Protocol Revision 11 or greater.
	<b>Test Directives</b>	Apply to the Start_Time and again to the Stop_Time properties in a Trend Log object.
	<b>Testing Hints</b>	
<b>BTL - 9.23.2.X11 - DateTime Non-Pattern Properties Test using WritePropertyMultiple Service</b>		
	<b>Test Conditionality</b>	This test shall only be applied to devices claiming Protocol Revision 11 or higher and which supports execution of WritePropertyMultiple.
	<b>Test Directives</b>	Apply to the Start_Time and again to the Stop_Time properties in a Trend Log object.
	<b>Testing Hints</b>	

### 7.7.8 Is Able to Trend REAL Datatypes

The IUT is able to trend REAL data type in a Trend Log Multiple Object.

<b>135.1-2013 - 9.21.1.1 - Reading All Items in the List, or</b> <b>135.1-2013 - 9.21.1.2 - Reading Items by Position with Positive Count, or</b> <b>135.1-2013 - 9.21.1.3 - Reading Items by Position with Negative Count, or</b> <b>135.1-2013 - 9.21.1.4 - Reading Items by Time, or</b> <b>135.1-2013 - 9.21.1.4.1 - Reading Items by Time with Negative Count, or</b> <b>135.1-2013 - 9.21.1.9 - Reading Items by Sequence with Positive Count, or</b> <b>135.1-2013 - 9.21.1.10 - Reading Items by Sequence with Negative Count.</b>		
	<b>Test Conditionality</b>	At least one of the tests listed above shall be run.
	<b>Test Directives</b>	As per test description. The Trend Log Multiple object used for this test shall contain REAL data types. The TLM object may contain other data types as well.
	<b>Testing Hints</b>	

### 7.7.9 Is Able to Trend Unsigned Datatypes

The IUT is able to trend Unsigned data type in a Trend Log Multiple Object.

<b>135.1-2013 - 9.21.1.1 - Reading All Items in the List, or</b> <b>135.1-2013 - 9.21.1.2 - Reading Items by Position with Positive Count, or</b> <b>135.1-2013 - 9.21.1.3 - Reading Items by Position with Negative Count, or</b> <b>135.1-2013 - 9.21.1.4 - Reading Items by Time, or</b> <b>135.1-2013 - 9.21.1.4.1 - Reading Items by Time with Negative Count, or</b> <b>135.1-2013 - 9.21.1.9 - Reading Items by Sequence with Positive Count, or</b> <b>135.1-2013 - 9.21.1.10 - Reading Items by Sequence with Negative Count.</b>		
	<b>Test Conditionality</b>	At least one of the tests listed above shall be run.
	<b>Test Directives</b>	As per test description. The Trend Log Multiple object used for this test shall contain Unsigned data types. The TLM object may contain other data types as well.
	<b>Testing Hints</b>	

### 7.7.10Is Able to Trend INTEGER Datatypes

The IUT is able to trend INTEGER data type in a Trend Log Multiple Object.

<b>135.1-2013 - 9.21.1.1 - Reading All Items in the List, or</b> <b>135.1-2013 - 9.21.1.2 - Reading Items by Position with Positive Count, or</b> <b>135.1-2013 - 9.21.1.3 - Reading Items by Position with Negative Count, or</b> <b>135.1-2013 - 9.21.1.4 - Reading Items by Time, or</b> <b>135.1-2013 - 9.21.1.4.1 - Reading Items by Time with Negative Count, or</b> <b>135.1-2013 - 9.21.1.9 - Reading Items by Sequence with Positive Count, or</b> <b>135.1-2013 - 9.21.1.10 - Reading Items by Sequence with Negative Count.</b>		
	<b>Test Conditionality</b>	At least one of the tests listed above shall be run.
	<b>Test Directives</b>	As per test description. The Trend Log Multiple object used for this test shall contain INTEGER data types. The TLM object may contain other data types as well.
	<b>Testing Hints</b>	

### 7.7.11Is Able to Trend BOOLEAN Datatypes

The IUT is able to trend BOOLEAN data type in a Trend Log Multiple Object.

<b>135.1-2013 - 9.21.1.1 - Reading All Items in the List, or</b> <b>135.1-2013 - 9.21.1.2 - Reading Items by Position with Positive Count, or</b> <b>135.1-2013 - 9.21.1.3 - Reading Items by Position with Negative Count, or</b> <b>135.1-2013 - 9.21.1.4 - Reading Items by Time, or</b> <b>135.1-2013 - 9.21.1.4.1 - Reading Items by Time with Negative Count, or</b> <b>135.1-2013 - 9.21.1.9 - Reading Items by Sequence with Positive Count, or</b> <b>135.1-2013 - 9.21.1.10 - Reading Items by Sequence with Negative Count.</b>		
	<b>Test Conditionality</b>	At least one of the tests listed above shall be run.
	<b>Test Directives</b>	As per test description. The Trend Log Multiple object used for this test shall contain BOOLEAN data types. The TLM object may contain other data types as well.
	<b>Testing Hints</b>	

### 7.7.12Is Able to Trend Bit String Datatypes

The IUT is able to trend Bit String data type in a Trend Log Multiple Object.

<b>135.1-2013 - 9.21.1.1 - Reading All Items in the List, or</b> <b>135.1-2013 - 9.21.1.2 - Reading Items by Position with Positive Count, or</b> <b>135.1-2013 - 9.21.1.3 - Reading Items by Position with Negative Count, or</b> <b>135.1-2013 - 9.21.1.4 - Reading Items by Time, or</b> <b>135.1-2013 - 9.21.1.4.1 - Reading Items by Time with Negative Count, or</b> <b>135.1-2013 - 9.21.1.9 - Reading Items by Sequence with Positive Count, or</b> <b>135.1-2013 - 9.21.1.10 - Reading Items by Sequence with Negative Count.</b>		
	<b>Test Conditionality</b>	At least one of the tests listed above shall be run.
	<b>Test Directives</b>	As per test description. The Trend Log Multiple object used for this test shall contain Bit String data types. The TLM object may contain other data types as well.
	<b>Testing Hints</b>	

### 7.7.13Is Able to Trend Enumerated Datatypes

The IUT is able to trend Enumerated data type in a Trend Log Multiple Object.

<b>135.1-2013 - 9.21.1.1 - Reading All Items in the List, or</b> <b>135.1-2013 - 9.21.1.2 - Reading Items by Position with Positive Count, or</b> <b>135.1-2013 - 9.21.1.3 - Reading Items by Position with Negative Count, or</b> <b>135.1-2013 - 9.21.1.4 - Reading Items by Time, or</b> <b>135.1-2013 - 9.21.1.4.1 - Reading Items by Time with Negative Count, or</b> <b>135.1-2013 - 9.21.1.9 - Reading Items by Sequence with Positive Count, or</b> <b>135.1-2013 - 9.21.1.10 - Reading Items by Sequence with Negative Count.</b>		
	<b>Test Conditionality</b>	At least one of the tests listed above shall be run.
	<b>Test Directives</b>	As per test description. The Trend Log Multiple object used for this test shall contain Enumerated data types. The TLM object may contain other data types as well.
	<b>Testing Hints</b>	

### 7.7.14Is Able to Trend NULL Datatypes

The IUT is able to trend NULL data type in a Trend Log Multiple Object.

<b>135.1-2013 - 9.21.1.1 - Reading All Items in the List, or</b> <b>135.1-2013 - 9.21.1.2 - Reading Items by Position with Positive Count, or</b> <b>135.1-2013 - 9.21.1.3 - Reading Items by Position with Negative Count, or</b> <b>135.1-2013 - 9.21.1.4 - Reading Items by Time, or</b> <b>135.1-2013 - 9.21.1.4.1 - Reading Items by Time with Negative Count, or</b> <b>135.1-2013 - 9.21.1.9 - Reading Items by Sequence with Positive Count, or</b> <b>135.1-2013 - 9.21.1.10 - Reading Items by Sequence with Negative Count.</b>		
	<b>Test Conditionality</b>	At least one of the tests listed above shall be run.
	<b>Test Directives</b>	As per test description. The Trend Log Multiple object used for this test shall contain NULL data types. The TLM object may contain other data types as well.
	<b>Testing Hints</b>	Schedule_Default and Present_Value of the Schedule Object, Alarm_Values and Fault_Values of the CharacterString Value Object and Low_Diff_Limit in the Loop Object, are standard properties that should accept a written NULL.



## 7.8 Trending - View and Modify Multiple Values - E - B

### 7.8.1 Base Requirements

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

This BIBB requires support of T-VMMV-I-B, DS-RPM-A.

This BIBB requires that the Log\_DeviceObjectProperty and Log\_Interval properties must be present and writable.

Verify Checklist		
	<b>Test Conditionality</b>	Must be executed
	<b>Test Directives</b>	Verify that the IUT claims support for DS-RPM-A in the Checklist.
	<b>Testing Hints</b>	
135.1-2013 - 9.21.1.12 - Status/Failure Logging		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	Tester may want to try other error conditions.
135.1-2013 - 7.3.2.24.16 - Interval Gathering of External Trends Test		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 7.8.2 Supports T-VMMV-I-B

Verify Checklist		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support for T-VMMV-I-B.
	<b>Testing Hints</b>	

### 7.8.3 Is Able to Trend REAL Datatypes

The IUT is able to trend REAL data type in a Trend Log Multiple Object.

135.1-2013 - 9.21.1.1 - Reading All Items in the List, <b>or</b> 135.1-2013 - 9.21.1.2 - Reading Items by Position with Positive Count, <b>or</b> 135.1-2013 - 9.21.1.3 - Reading Items by Position with Negative Count, <b>or</b> 135.1-2013 - 9.21.1.4 - Reading Items by Time, <b>or</b> 135.1-2013 - 9.21.1.4.1 - Reading Items by Time with Negative Count, <b>or</b> 135.1-2013 - 9.21.1.9 - Reading Items by Sequence with Positive Count, <b>or</b> 135.1-2013 - 9.21.1.10 - Reading Items by Sequence with Negative Count.		
	<b>Test Conditionality</b>	At least one of the tests listed above shall be run.
	<b>Test Directives</b>	As per test description. The Trend Log Multiple object used for this test shall contain REAL data types. The TLM object may contain other data types as well.
	<b>Testing Hints</b>	

### 7.8.4 Is Able to Trend Unsigned Datatypes

The IUT is able to trend Unsigned data type in a Trend Log Multiple Object.

<b>135.1-2013 - 9.21.1.1 - Reading All Items in the List, or</b> <b>135.1-2013 - 9.21.1.2 - Reading Items by Position with Positive Count, or</b> <b>135.1-2013 - 9.21.1.3 - Reading Items by Position with Negative Count, or</b> <b>135.1-2013 - 9.21.1.4 - Reading Items by Time, or</b> <b>135.1-2013 - 9.21.1.4.1 - Reading Items by Time with Negative Count, or</b> <b>135.1-2013 - 9.21.1.9 - Reading Items by Sequence with Positive Count, or</b> <b>135.1-2013 - 9.21.1.10 - Reading Items by Sequence with Negative Count.</b>		
	<b>Test Conditionality</b>	At least one of the tests listed above shall be run.
	<b>Test Directives</b>	As per test description. The Trend Log Multiple object used for this test shall contain Unsigned data types. The TLM object may contain other data types as well.
	<b>Testing Hints</b>	

### 7.8.5 Is Able to Trend INTEGER Datatypes

The IUT is able to trend INTEGER data type in a Trend Log Multiple Object.

<b>135.1-2013 - 9.21.1.1 - Reading All Items in the List, or</b> <b>135.1-2013 - 9.21.1.2 - Reading Items by Position with Positive Count, or</b> <b>135.1-2013 - 9.21.1.3 - Reading Items by Position with Negative Count, or</b> <b>135.1-2013 - 9.21.1.4 - Reading Items by Time, or</b> <b>135.1-2013 - 9.21.1.4.1 - Reading Items by Time with Negative Count, or</b> <b>135.1-2013 - 9.21.1.9 - Reading Items by Sequence with Positive Count, or</b> <b>135.1-2013 - 9.21.1.10 - Reading Items by Sequence with Negative Count.</b>		
	<b>Test Conditionality</b>	At least one of the tests listed above shall be run.
	<b>Test Directives</b>	As per test description. The Trend Log Multiple object used for this test shall contain INTEGER data types. The TLM object may contain other data types as well.
	<b>Testing Hints</b>	

### 7.8.6 Is Able to Trend BOOLEAN Datatypes

The IUT is able to trend BOOLEAN data type in a Trend Log Multiple Object.

<b>135.1-2013 - 9.21.1.1 - Reading All Items in the List, or</b> <b>135.1-2013 - 9.21.1.2 - Reading Items by Position with Positive Count, or</b> <b>135.1-2013 - 9.21.1.3 - Reading Items by Position with Negative Count, or</b> <b>135.1-2013 - 9.21.1.4 - Reading Items by Time, or</b> <b>135.1-2013 - 9.21.1.4.1 - Reading Items by Time with Negative Count, or</b> <b>135.1-2013 - 9.21.1.9 - Reading Items by Sequence with Positive Count, or</b> <b>135.1-2013 - 9.21.1.10 - Reading Items by Sequence with Negative Count.</b>		
	<b>Test Conditionality</b>	At least one of the tests listed above shall be run.
	<b>Test Directives</b>	As per test description. The Trend Log Multiple object used for this test shall contain BOOLEAN data types. The TLM object may contain other data types as well.
	<b>Testing Hints</b>	

### 7.8.7 Is Able to Trend Bit String Datatypes

The IUT is able to trend Bit String data type in a Trend Log Multiple Object.

<b>135.1-2013 - 9.21.1.1 - Reading All Items in the List, or</b> <b>135.1-2013 - 9.21.1.2 - Reading Items by Position with Positive Count, or</b> <b>135.1-2013 - 9.21.1.3 - Reading Items by Position with Negative Count, or</b> <b>135.1-2013 - 9.21.1.4 - Reading Items by Time, or</b> <b>135.1-2013 - 9.21.1.4.1 - Reading Items by Time with Negative Count, or</b> <b>135.1-2013 - 9.21.1.9 - Reading Items by Sequence with Positive Count, or</b> <b>135.1-2013 - 9.21.1.10 - Reading Items by Sequence with Negative Count.</b>		
	<b>Test Conditionality</b>	At least one of the tests listed above shall be run.
	<b>Test Directives</b>	As per test description. The Trend Log Multiple object used for this test shall contain Bit String data types. The TLM object may contain other data types as well.
	<b>Testing Hints</b>	

### 7.8.8 Is Able to Trend Enumerated Datatypes

The IUT is able to trend Enumerated data type in a Trend Log Multiple Object.

<b>135.1-2013 - 9.21.1.1 - Reading All Items in the List, or</b> <b>135.1-2013 - 9.21.1.2 - Reading Items by Position with Positive Count, or</b> <b>135.1-2013 - 9.21.1.3 - Reading Items by Position with Negative Count, or</b> <b>135.1-2013 - 9.21.1.4 - Reading Items by Time, or</b> <b>135.1-2013 - 9.21.1.4.1 - Reading Items by Time with Negative Count, or</b> <b>135.1-2013 - 9.21.1.9 - Reading Items by Sequence with Positive Count, or</b> <b>135.1-2013 - 9.21.1.10 - Reading Items by Sequence with Negative Count.</b>		
	<b>Test Conditionality</b>	At least one of the tests listed above shall be run.
	<b>Test Directives</b>	As per test description. The Trend Log Multiple object used for this test shall contain Bit String data types. The TLM object may contain other data types as well.
	<b>Testing Hints</b>	

### 7.8.9 Is Able to Trend NULL Datatypes

The IUT is able to trend NULL data type in a Trend Log Multiple Object.

<b>135.1-2013 - 9.21.1.1 - Reading All Items in the List, or</b> <b>135.1-2013 - 9.21.1.2 - Reading Items by Position with Positive Count, or</b> <b>135.1-2013 - 9.21.1.3 - Reading Items by Position with Negative Count, or</b> <b>135.1-2013 - 9.21.1.4 - Reading Items by Time, or</b> <b>135.1-2013 - 9.21.1.4.1 - Reading Items by Time with Negative Count, or</b> <b>135.1-2013 - 9.21.1.9 - Reading Items by Sequence with Positive Count, or</b> <b>135.1-2013 - 9.21.1.10 - Reading Items by Sequence with Negative Count.</b>		
	<b>Test Conditionality</b>	At least one of the tests listed above shall be run.
	<b>Test Directives</b>	As per test description. The Trend Log Multiple object used for this test shall contain NULL data types. The TLM object may contain other data types as well.
	<b>Testing Hints</b>	Schedule_Default and Present_Value of the Schedule Object, Alarm_Values and Fault_Values of the CharacterString Value Object and Low_Diff_Limit in the Loop Object, are standard properties that should accept a written NULL.

## 7.9 Trending - Automated Multiple Value Retrieval - A

### 7.9.1 Base Requirements

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

135.1-2013 - 8.21.6 - Reading a Range of Items Using Any Valid Range in Response to ConfirmedEventNotifications of the Buffer_Ready Event Type		
	Test Conditionality	Must be executed
	Test Directives	
	Testing Hints	Repeat the test for each form of the 'Time Stamp' parameter - BACnetDateTime, Time, and Sequence Number. For each test, the values in Steps 2 and 7 shall take the appropriate form.

### 7.9.2 Initiates ReadRange

The IUT is able to initiate the ReadRange service.

Verify Checklist		
	Test Conditionality	Must be executed.
	Test Directives	Verify that the IUT claims support for Initiates ReadRange in the Checklist.
	Testing Hints	

### 7.9.3 Processes ConfirmedEventNotification of BUFFER\_READY Events

The IUT can process ConfirmedEventNotification messages indicating an 'Event Type' of Buffer\_Ready, and respond with a series of ReadRange requests that access records from the Trend Log Multiple referenced in the notifications.

135.1-2013 - 8.21.6 - Reading a Range of Items Using Any Valid Range in Response to ConfirmedEventNotifications of the Buffer_Ready Event Type		
	Test Conditionality	Must be executed
	Test Directives	
	Testing Hints	

### 7.9.4 Processes UnconfirmedEventNotification of BUFFER\_READY Events

The IUT can process UnconfirmedEventNotification messages indicating an 'Event Type' of Buffer\_Ready, and respond with a series of ReadRange requests that access records from the Trend Log Multiple referenced in the notifications.

135.1-2013 - 8.21.7 - Reading a Range of Items Using Any Valid Range in Response to UnconfirmedEventNotifications of the Buffer_Ready Event Type		
	Test Conditionality	Must be executed
	Test Directives	
	Testing Hints	

## 7.10 Trending - Automated Multiple Value Retrieval - B

### 7.10.1 Base Requirements

Base requirements must be met by any IUT claiming conformance to this BIBB. There are no base requirements tests for this section.

### 7.10.2 Supports T-VMMV-I-B

The IUT supports T-VMMV-I-B.

Verify Checklist		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support for T-VMMV-I-B in the Checklist.
	<b>Testing Hints</b>	

### 7.10.3 Supports the Notification Class Object

The IUT supports the Notification Class Object in order to send Buffer\_Ready notifications.

Verify Checklist		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support for the Notification Class Object in the Checklist.
	<b>Testing Hints</b>	

### 7.10.4 Implements Intrinsic Alarming

The IUT contains, or can be made to contain, a Trend Log Multiple object that can generate BUFFER\_READY notifications.

Each Trend Log Multiple object must include all of the properties required for intrinsic reporting, i.e. Notification\_Threshold, Records\_Since\_Notification, Last\_Notify\_Record, Notification\_Class, Event\_Enable, Acked\_Transitions, Notify\_Type, & Event\_Time\_Stamps.

BTL - 7.3.1.10.2 - Event Enable Tests for TO NORMAL only Algorithms		
	<b>Test Conditionality</b>	If the IUT cannot be configured to meet the configuration requirements then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
135.1-2013 - 7.3.1.12 - Notify Type Test		
	<b>Test Conditionality</b>	If the IUT cannot be configured to meet the 135.1-2013 configuration requirements then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
135.1-2013 - 7.3.2.24.10 - Notification Threshold Test		
	<b>Test Conditionality</b>	If the IUT cannot be configured to meet the 135.1-2013 configuration requirements then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
135.1-2013 - 7.3.2.24.17 - Last Notify Record Test		
	<b>Test Conditionality</b>	If the IUT cannot be configured to meet the configuration requirements then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
135.1-2013 - 7.3.2.24.18 - Records Since Notification		

	<b>Test Conditionality</b>	If the IUT cannot be configured to meet the 135.1-2013 configuration requirements then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 8.4.7 - BUFFER READY Tests</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	The 'Event Object Identifier' in this test must be a Trend Log Multiple object contained in the IUT.
<b>135.1-2013 - 8.5.7 - BUFFER READY Tests</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	The 'Event Object Identifier' in this test must be a Trend Log Multiple object contained in the IUT.

### 7.10.5 Implements Algorithmic Alarming

The IUT contains, or can be made to contain an Event Enrollment object that can generate a Buffer\_Ready ConfirmedEventNotification.

<b>Verify Checklist</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support for the Event Enrollment object in the checklist.
	<b>Testing Hints</b>	
<b>135.1-2013 - 8.4.7 - BUFFER READY Tests</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	The 'Event Object Identifier' in this test must be an Event Enrollment object contained in the IUT.
<b>135.1-2013 - 8.5.7 - BUFFER READY Tests</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	The 'Event Object Identifier' in this test must be an Event Enrollment object contained in the IUT.

### 7.10.6 Generates Event Notifications with Timestamps of the BACnetDateTime Form

The IUT generates, or can be made to generate, ConfirmedEventNotifications with the Time Stamp parameter taking the BACnetDateTime form.

<b>135.1-2013 - 8.4 - ConfirmedEventNotification Service Initiation Tests</b>		
	<b>Test Conditionality</b>	If the IUT supports AE-N-I-B, these tests may be skipped. Any of the 8.4 tests can be used to ensure that the IUT properly generates ConfirmedEventNotification requests using the BACnetDateTime form. The specific tests that can be executed are detailed under the test cases for the specific algorithms. As long as one of the tests is executed using ConfirmedEventNotifications and the notification that is generated contains a timestamp of the BACnetDateTime form, then this test case shall be satisfied.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 7.10.7 Generates Event Notifications with Timestamps of the Time Form

The IUT generates, or can be made to generate, ConfirmedEventNotifications with the Time Stamp parameter taking the Time form.

135.1-2013 - 8.4 - ConfirmedEventNotification Service Initiation Tests		
	<b>Test Conditionality</b>	If the IUT supports AE-N-I-B, these tests may be skipped. Any of the 8.4 tests can be used to ensure that the IUT properly generates ConfirmedEventNotification requests using the Time form. The specific tests that can be executed are detailed under the test cases for the specific algorithms. As long as one of the tests is executed using ConfirmedEventNotifications and the notification that is generated contains a timestamp of the Time form, then this test case shall be satisfied.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 7.10.8 Generates Event Notifications with Timestamps of the Sequence Number Form

The IUT generates, or can be made to generate, ConfirmedEventNotifications with the Time Stamp parameter taking the Sequence Number form.

135.1-2013 - 8.4 - ConfirmedEventNotification Service Initiation Tests		
	<b>Test Conditionality</b>	If the IUT supports AE-N-I-B, these tests may be skipped. Any of the 8.4 tests can be used to ensure that the IUT properly generates ConfirmedEventNotification requests using the Sequence Number form. The specific tests that can be executed are detailed under the test cases for the specific algorithms. As long as one of the tests is executed using ConfirmedEventNotifications and the notification that is generated contains a timestamp of the Sequence Number form, then this test case shall be satisfied.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

## 7.11 Trending - Archival - A

### 7.11.1 Base Requirements

Base requirements must be met by any IUT claiming conformance to this BIBB. There are no base requirements tests for this section.

Verify Non-Volatile Storage and Retrieval		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT store trend data in a non-volatile method and the data can be retrieved electronically.
	<b>Testing Hints</b>	

### 7.11.2 Supports T-ATR-A

The IUT supports T-ATR-A.

Verify Checklist		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support for T-ATR-A in the Checklist.
	<b>Testing Hints</b>	

### 7.11.3 Supports T-AMVR-A

The IUT supports T-AMVR-A.

Verify Checklist		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support for T-AMVR-A in the Checklist.
	<b>Testing Hints</b>	



---

## **7.12 Trending - View and Modify Trends - A**

---

### **7.12.1 Base Requirements**

Base requirements must be met by any IUT claiming conformance to this BIBB. There are no base requirements tests for this section.

---

## **7.13 Trending - View and Modify Multiple Values - A**

---

### **7.13.1 Base Requirements**

Base requirements must be met by any IUT claiming conformance to this BIBB. There are no base requirements tests for this section.

---

## **8 Device Management BIBBs**

---

## 8.1 Device Management - Dynamic Device Binding - A

### 8.1.1 Base Requirements

Base requirements must be met by any IUT claiming conformance to this BIBB. There are no base requirements tests for this section.

### 8.1.2 Initiates Who-Is Service Request with Range Parameters

The BTL requires support for initiating Who-Is service requests with range parameters, even though SSPC 135 does not. This is because the use of Who-Is service requests with no range parameters can cause I-Am “broadcast storms”, which can adversely affect network performance.

The IUT can initiate a Who-Is Service Request with device instance range parameters.

BTL - 8.34.2 - Who-Is Request with a Device Instance Range		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 8.1.3 Initiates Unicast Who-Is Service Request with no Range Parameters

The IUT can initiate a unicast Who-Is Service Request with no device instance range parameters.

135.1-2013 - 8.34.3 - Who-Is Request with no Device Instance Range		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 8.1.4 Initiates Broadcast Who-Is Service Request with no Range Parameters

The IUT can initiate a directed or global broadcast Who-Is Service Request with no device instance range parameters.

135.1-2013 - 8.34.1 - Who-Is Request with no Device Instance Range		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

## 8.2 Device Management - Dynamic Device Binding - B

### 8.2.1 Base Requirements

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

<b>135.1-2013 - 9.33.1.1 - Local Broadcast, General Inquiry</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.33.1.2 - Global Broadcast, General Inquiry</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.33.1.3 - Local Broadcast, Specific Device Inquiry with IUT Outside of the Device Range</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.33.1.4 - Local Broadcast, Specific Device Inquiry with IUT Device Instance Equal to Low Limit of Device Range</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.33.1.5 - Local Broadcast, Specific Device Inquiry with IUT Device Instance Equal to High Limit of Device Range</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.33.1.6 - Local Broadcast, Specific Device Inquiry with IUT Inside of the Device Range</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.33.2.1 - General Inquiry, Global Broadcast from a Remote Network</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.33.2.2 - General Inquiry, Remote Broadcast</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.33.2.3 - General Inquiry, Directed to a Remote Device</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

## 8.3 Device Management - Dynamic Object Binding - A

### 8.3.1 Base Requirements

Base requirements must be met by any IUT claiming conformance to this BIBB. There are no base requirements tests for this section.

### 8.3.2 Initiates Who-Has Service Request with Object Identifier Parameter and no Range Parameters

The IUT can initiate a Who-Has service request with an ObjectID and No Range Parameters.

BTL - 8.32.1 - Object Identifier Selection with no Device Instance Range		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 8.3.3 Initiates Who-Has Service Request with Object Identifier Parameter with Range Parameters

The IUT can initiate a Who-Has service request with an ObjectID and Range Parameters.

BTL - 8.32.3 - Object Identifier Selection with a Device Instance Range		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 8.3.4 Initiates Who-Has Service Request with Object Name Parameter and no Range Parameters

The IUT can initiate a Who-Has service request with an Object Name and No Range Parameters.

BTL - 8.32.2 - Object Name Selection with no Device Instance Range		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 8.3.5 Initiates Who-Has Service Request with Object Name Parameter with Range Parameters

The IUT can initiate a Who-Has service request with an Object Name and Range Parameters.

BTL - 8.32.4 - Object Name Selection with a Device Instance Range		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

## 8.4 Device Management - Dynamic Object Binding - B

### 8.4.1 Base Requirements

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

<b>BTL - 9.32.1.1 - Object ID Version with no Device Range</b>		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
<b>BTL - 9.32.1.2 - Object Name Version with no Device Range</b>		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
<b>BTL - 9.32.1.3 - Object ID Version with IUT Inside of the Device Range</b>		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
<b>BTL - 9.32.1.4 - Object ID Version with IUT Outside of the Device Range</b>		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
<b>BTL - 9.32.1.5 - Object Name Version with IUT Inside of the Device Range</b>		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
<b>135.1-2013 - 9.32.1.6 - Object Name Version with IUT Outside of the Device Range</b>		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
<b>BTL - 9.32.1.7 - Object ID Version with IUT Device Instance Equal to the High Limit of the Device Range</b>		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
<b>BTL - 9.32.1.8 - Object ID Version with IUT Device Instance Equal to the Low Limit of the Device Range</b>		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
<b>BTL - 9.32.1.9 - Object Name Version with IUT Device Instance Equal to the High Limit of the Device Range</b>		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
<b>BTL - 9.32.1.10 - Object Name Version with IUT Device Instance Equal to the Low Limit of the Device Range</b>		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
<b>BTL - 9.32.1.11 - Object Name Version, Directed to a Specific MAC Address</b>		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

<b>BTL - 9.32.2.1 - Object ID Version, Global Broadcast from a Remote Network</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.32.2.2 - Object ID Version, Remote Broadcast</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.32.1.12 - Who-Has After Object Name Changed</b>		
	<b>Test Conditionality</b>	If the IUT contains an object whose Object_Name can be changed, then this test shall be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.32.1.13 - Who-Has After Object Identifier Changed</b>		
	<b>Test Conditionality</b>	If the IUT contains an object whose Object_Identifier can be changed, then this test shall be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.32.2.X3 - Who-Has for Non-existent Object Name</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.32.2.X5 - Who-Has for Non-existent Object Identifier</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	



## 8.5 Device Management - Automatic Device Mapping - A

### 8.5.1 Base Requirements

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

135.1-2013 - 13.7 - Automatic Device Mapping		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	<p>4 types of reference servers to test against are:</p> <ul style="list-style-type: none"> <li>• S1 supports ReadProperty, supports segmentation, and contains an Object_List property that can be returned in a segmented APDU.</li> <li>• S2 supports ReadProperty, does not support segmentation, and contains an Object_List property that cannot be returned in a single APDU.</li> <li>• S3 supports ReadProperty and ReadPropertyMultiple, supports segmentation, and contains an Object_List property that can be returned in a segmented APDU.</li> <li>• S4 supports ReadProperty and ReadPropertyMultiple, does not support segmentation, and contains an Object_List property that cannot be returned in a single APDU.</li> </ul> <p>The server device shall be configured to contain a collection of standard and proprietary objects.</p> <p>Server supports segmentation but contains an object-list that cannot be returned in the number of segments supported by the server or client.</p> <p>Configuring reference servers with MaxAPDUs that do not match the underlying datalink size may provide more test benefit.</p>

### 8.5.2 Supports DS-RP-A

The IUT supports DS-RP-A.

Verify Checklist		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify that the IUT claims support for DS-RP-A in the checklist.
	<b>Testing Hints</b>	

## 8.6 Device Management - Automatic Network Mapping - A

### 8.6.1 Base Requirements

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

135.1-2013 - 13.6 - Automatic Network Mapping		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 8.6.2 Supports DM-DDB-A

The IUT supports DM-DDB-A.

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

## 8.7 Device Management - Time Synchronization - A

### 8.7.1 Base Requirements

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

135.1-2013 - 8.30 - TimeSynchronization Service Initiation Tests		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 8.7.2 Supports DM-UTC-A

The IUT supports DM-UTC-A.

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

## 8.8 Device Management - Time Synchronization - B

### 8.8.1 Base Requirements

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

<b>BTL - 9.30.1.1 - TimeSynchronization Local Broadcast</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Ensure to test with the following date values: Feb 28, Feb 29, Mar 1 Dec 31, Jan 1 of the following year 28-Feb-2100 if IUT supports Dates in the year 2100
	<b>Testing Hints</b>	
<b>BTL - 9.30.1.2 - TimeSynchronization Directed to the IUT</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

## 8.9 Device Management - UTC Time Synchronization - A

### 8.9.1 Base Requirements

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

135.1-2013 - 8.31 - UTCTimeSynchronization Service Initiation Tests		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 8.9.2 Supports DM-TS-A

The IUT supports DM-TS-A.

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

## 8.10 Device Management - UTC Time Synchronization - B

### 8.10.1 Base Requirements

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

<b>BTL - 9.31.1.1 - UTC TimeSynchronization Local Broadcast</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Values tested in the UTCTimeSynchronization-Request should include at least one which, in combination with the UTC Offset, shall cause the result to cross midnight so that the resulting date is different from the date in the UTCTimeSynchronization-Request. Ensure to test with the following date values: Feb 28, Feb 29, Mar 1 Dec 31, Jan 1 of the following year 28-Feb-2100 if IUT supports Dates in the year 2100
	<b>Testing Hints</b>	
<b>BTL - 9.31.1.2 - UTC TimeSynchronization Directed to the IUT</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

## 8.11 Device Management - Automatic Time Synchronization - A

### 8.11.1 Base Requirements

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

<b>135.1-2013 - 13.2.1 - TimeSynchronization Recipients Test, Protocol_Revision &lt; 7</b>		
	<b>Test Conditionality</b>	If Protocol_Revision >= 7, then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 13.2.2 - TimeSynchronization Recipients Test, Protocol_Revision &gt;= 7</b>		
	<b>Test Conditionality</b>	If Protocol_Revision < 7, then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 13.2.3 - UTCTimeSynchronization Recipients Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 13.2.4 - TimeSynchronization Interval Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 13.2.5 - UTCTimeSynchronization Interval Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 13.2.6 - Align_Intervals and Interval_Offset TimeSynchronization Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 13.2.7 - Align_Intervals and Interval_Offset UTCTimeSynchronization Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

---

## 8.12 Device Management - Manual Time Synchronization - A

---

### 8.12.1 Base Requirements

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

<b>135.1-2013 - 8.30 - TimeSynchronization Service Initiation Tests</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 8.31 - UTCTimeSynchronization Service Initiation Tests</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	



## 8.13 Device Management - Device Communication Control - A

### 8.13.1 Base Requirements

Base requirements must be met by any IUT claiming conformance to this BIBB. There are no base requirements tests for this section.

### 8.13.2 Supports Sending a DeviceCommunicationControl Service Request with an Arbitrary Password

The IUT is capable of initiating a DeviceCommunicationControl service request with an arbitrary password. The IUT must allow the user to specify the password. The IUT must not modify the password in any way, such as forcing the password to uppercase or to lowercase.

BTL - 8.24.3 - Time Duration, Disable, Password		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
BTL - 8.24.4 - Enable, Password		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 8.13.3 Supports Sending a DeviceCommunicationControl Service Request with a Finite Timeout

The IUT is capable of initiating DeviceCommunicationControl service requests with finite Timeout parameter values.

BTL - 8.24.3 - Time Duration, Disable, Password		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 8.13.4 Supports DM-RD-A

The IUT supports DM-RD-A.

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

### 8.13.5 Supports Sending a DeviceCommunicationControl Service Request with no Password

The IUT is capable of initiating a DeviceCommunicationControl service request with no password parameter.

BTL - 8.24.6 - Time Duration, Disable, No Password		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
BTL - 8.24.5 - Enable, No Password		
	Test Conditionality	Must be executed.
	Test Directives	

	Testing Hints	
--	---------------	--

### 8.13.6 Supports Sending a DeviceCommunicationControl Service Request with an Infinite Timeout

The IUT is capable of initiating a DeviceCommunicationControl service request with an infinite timeout.

BTL - 8.24.2 - Indefinite Duration, Disable, Password		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

## 8.14 Device Management - Device Communication Control - B

### 8.14.1 Base Requirements

There are no base requirements for this BIBB.

### 8.14.2 Supports Receiving a DeviceCommunicationControl Service Request with a Password

The IUT requires, or can be made to require, a valid password parameter in a DeviceCommunicationControl service request.

135.1-2013 - 9.24.2.1 - Invalid Password		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
135.1-2013 - 9.24.2.2 - Missing Password		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 8.14.3 Supports Receiving a DeviceCommunicationControl Service Request with no Password

The IUT does not require, or can be made to not require, a password parameter in a DeviceCommunicationControl service request.

135.1-2013 - 9.24.1.3 - Finite Time Duration		
	Test Conditionality	If the IUT does not support an internal clock this test may be skipped and test 9.24.1.1 shall be executed.
	Test Directives	The service request shall not contain a password.
	Testing Hints	
135.1-2013 - 9.24.1.1 - Indefinite Time Duration Restored by DeviceCommunicationControl		
	Test Conditionality	If the IUT does not support indefinite time duration, this test shall be skipped.
	Test Directives	The service request shall not contain a password.
	Testing Hints	

### 8.14.4 Supports Receiving a DeviceCommunicationControl Service Request with a Finite Duration

The IUT will accept, or can be made to accept, a DeviceCommunicationControl Service request with a Time Duration parameter.

135.1-2013 - 9.24.1.3 - Finite Time Duration		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
135.1-2013 - 9.24.1.4 - Finite Time Duration Restored by DeviceCommunicationControl		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 8.14.5 Supports Receiving a DeviceCommunicationControl Service Request with an Indefinite Duration

The IUT will accept, or can be made to accept, a DeviceCommunicationControl Service request with no Time Duration parameter.

135.1-2013 - 9.24.1.1 - Indefinite Time Duration Restored by DeviceCommunicationControl		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 8.14.6 Supports DM-RD-B

The IUT also supports the DM-RD-B BIBB.

135.1-2013 - 9.24.1.2 - Indefinite Time Duration Restored by ReinitializeDevice		
	Test Conditionality	If the IUT does not support indefinite Time Duration, this test may be skipped.
	Test Directives	
	Testing Hints	
BTL - 9.24.1.5 - Finite Time Duration Restored by ReinitializeDevice		
	Test Conditionality	If the IUT does not support an internal clock, this test may be skipped.
	Test Directives	
	Testing Hints	
BTL - 9.24.2.3 - Restore by ReinitializeDevice with Invalid 'Reinitialized State of Device'		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 8.14.7 Supports Receiving a DeviceCommunicationControl Service Request Specifying DISABLE\_INITIATION

The IUT will accept the value of DISABLE\_INITIATION in the enable-disable parameter of the service request.

135.1-2013 - 9.24.1.9 - Disable of Service Initiation Restored by Time Duration		
	Test Conditionality	If the IUT cannot be made to initiate any service, this test shall be skipped.
	Test Directives	
	Testing Hints	
135.1-2013 - 9.24.1.10 - Disable of Service Initiation Restored by DeviceCommunicationControl		
	Test Conditionality	If the IUT cannot be made to initiate any service, this test shall be skipped.
	Test Directives	
	Testing Hints	

## 8.15 Device Management - Reinitialize Device - A

### 8.15.1 Base Requirements

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

### 8.15.2 Supports Sending a ReinitializeDevice Service request with an Arbitrary Password

The IUT is capable of initiating a ReinitializeDevice request that contains a password.

<b>BTL - 8.27.2 - COLDSTART with a Password</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 8.27.4 - WARMSTART with a Password</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 8.15.3 Supports Sending a ReinitializeDevice Service Request with No Password

The IUT is capable of initiating a ReinitializeDevice request that does not contain a password. This functionality is optional because servers that do not require passwords are obliged to ignore any password that is provided in a request. Therefore interoperability is not hampered if a client is unable to leave out the password parameter from the service.

<b>135.1-2013 - 8.27.1 - COLDSTART with no Password</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 8.27.3 - WARMSTART with no Password</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

## 8.16 Device Management - Reinitialize Device - B

### 8.16.1 Base Requirements

Base requirements must be met by any IUT claiming conformance to this BIBB. There are no base requirements tests for this section.

### 8.16.2 Implements ReinitializeDevice WARMSTART with a Password

The IUT can deny, or can be made to deny, a ReinitializeDevice WARMSTART service request that does not contain a valid password.

135.1-2013 - 9.27.1.4 - WARMSTART with a Correct Password		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
BTL - 9.27.2.4 - WARMSTART with Missing or Invalid Password		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 8.16.3 Implements ReinitializeDevice WARMSTART with no Password

The IUT accepts, or can be made to accept, a ReinitializeDevice WARMSTART service request that does not contain a valid password.

135.1-2013 - 9.27.1.3 - WARMSTART with no Password		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 8.16.4 Implements ReinitializeDevice COLDSTART with a Password

The IUT can deny, or can be made to deny, a ReinitializeDevice COLDSTART service request that does not contain a valid password.

135.1-2013 - 9.27.1.2 - COLDSTART with a Correct Password		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
BTL - 9.27.2.3 - COLDSTART with Missing or Invalid Password		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 8.16.5 Implements ReinitializeDevice COLDSTART with no Password

The IUT accepts, or can be made to accept, a ReinitializeDevice COLDSTART service request that does not contain a valid password.

135.1-2013 - 9.27.1.1 - COLDSTART with no Password		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

## 8.17 Device Management - Backup and Restore - A

### 8.17.1 Base Requirements

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

<b>BTL - 13.8.2.1 - Initiate a Full Backup and Restore</b>		
	<b>Test Conditionality</b>	This test should be repeated in order to cover all of the TD characteristics listed in the definition of the test.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 13.8.2.2 - Can Abort Backup if Error Received from TD</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 13.8.2.3 - Can Abort Restore if Error Received from TD</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 8.17.2 Supports User Initiated Abort Backup

The IUT supports a user initiated abort of the backup procedure.

<b>135.1-2013 - 13.8.2.4 - Initiate an Abort Backup</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 8.17.3 Supports User Initiated Abort Restore

The IUT supports the user initiating an abort of the restore procedure.

<b>135.1-2013 - 13.8.2.5 - Initiate an Abort Restore</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

## 8.18 Device Management - Backup and Restore - B

### 8.18.1 Base Requirements

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

<b>BTL - 13.8.1.1 - Execution of Full Backup and Restore Procedure</b>		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
<b>BTL - 13.8.1.2 - Attempting Backup While Already Performing a Backup Procedure</b>		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
<b>BTL - 13.8.1.3 - Attempting Backup While Already Performing a Restore Procedure</b>		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
<b>BTL - 13.8.1.4 - Attempting Restore While Already Performing a Backup Procedure</b>		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
<b>BTL - 13.8.1.5 - Attempting Restore While Already Performing a Restore Procedure</b>		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
<b>BTL - 13.8.1.6 - Ending Backup and Restore Procedures via Timeout</b>		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
<b>BTL - 13.8.1.7 - Ending Backup and Restore Procedures via Abort</b>		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 8.18.2 Supports Password Protected Backup

The IUT denies, or can be made to deny, a ReinitializeDevice <STARTBACKUP > service request that does not contain a valid password.

<b>BTL - 13.8.1.8 - Attempting Backup with an Invalid Password</b>		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 8.18.3 Supports Non-Password Protected Backup

The IUT does not require, or can be made to not require, a password for a ReinitializeDevice <STARTBACKUP> service request.

<b>BTL - 13.8.1.10 - Executing and Ending a Backup Procedure when a password is not required</b>		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	



### 8.18.4 Supports Password Protected Restore

The IUT denies, or can be made to deny, a ReinitializeDevice <STARTRESTORE > service request that does not contain a valid password.

BTL - 13.8.1.9 - Attempting Restore with an Invalid Password		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 8.18.5 Supports Non-Password Protected Restore

The IUT does not require, or can be made to not require, a password for a ReinitializeDevice <STARTRESTORE> service request.

BTL - 13.8.1.11 - Executing and Ending a Restore Procedure when a Password is not Required		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 8.18.6 Changes Operational Behavior during a Backup Procedure

The IUT changes its operational behavior during a Backup Procedure.

BTL - 13.8.1.12 - System Status during a Backup Procedure		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 8.18.7 Changes Operational Behavior during a Restore Procedure

The IUT changes its operational behavior during a Restore Procedure.

BTL - 13.8.1.13 - System Status during a Restore Procedure		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

---

## 8.19 Device Management - Restart - A

---

### 8.19.1 Base Requirements

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

135.1-2013 - 9.3.1 - Device Restart Notifications		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	Repeat the test with unicast and broadcast notifications.

---

## 8.20 Device Management - Restart - B

---

### 8.20.1 Base Requirements

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

BTL - 8.3.10 - Device Restart Notifications		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	Repeat the test with unicast and broadcast recipients. Repeat the test with each of the restart methods that the device supports and which can be performed at will (warm start, cold start, power cycle, power lost, etc).

## 8.21 Device Management - Object Creation and Deletion - A

### 8.21.1 Base Requirements

Base requirements must be met by any IUT claiming conformance to this BIBB. There are no base requirements tests for this section.

### 8.21.2 Can Create Objects using Object\_Identifier with No Initial Values

The IUT can create the specified object using Object\_Identifier and no initial values

No Specific Test		
	Test Conditionality	
	Test Directives	
	Testing Hints	This functionality will be tested under the specific object section below

### 8.21.3 Can Create Objects using Object\_Type with no Initial Values

The IUT can create the specified object using Object\_Type and no initial values.

135.1-2013 - 8.16.2 - Creating Objects by Specifying the Object Type with no Initial Values		
	Test Conditionality	
	Test Directives	The tester should execute this test using one of the object types claimed by the vendor in this section.
	Testing Hints	None

### 8.21.4 Can Create Objects by Object\_Identifier with Initial Values which Includes Object\_Name

The IUT can create objects by specifying the Object\_Name and 0 or more additional object properties as initial values.

135.1-2013 - 8.16.3 - Creating Objects by Specifying the Object Identifier and Providing Initial Values		
	Test Conditionality	Must be Executed.
	Test Directives	
	Testing Hints	Object_Name has to be included in the list of initial values.

### 8.21.5 Can Create Objects by Object\_Type with Initial Values which Includes Object\_Name

The IUT can create objects by specifying the Object\_Name and 0 or more additional object properties as initial values.

135.1-2013 - 8.16.4 - Creating Objects by Specifying the Object Type and Providing Initial Values		
	Test Conditionality	Must be Executed.
	Test Directives	
	Testing Hints	Object_Name has to be included in the list of initial values.

### 8.21.6 Can Create and Delete Accumulator Objects

The IUT can create and delete Accumulator objects. The IUT shall not restrict the instance number which can be used to create the Accumulator object.

135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values		
	Test Conditionality	Must be tested on the Accumulator Object

	<b>Test Directives</b>	
	<b>Testing Hints</b>	
	<b>135.1-2013 - 8.17 - Delete Object Service</b>	
	<b>Test Conditionality</b>	Must be tested on the Accumulator Object
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 8.21.7 Can Create and Delete Analog Input Objects

The IUT can create and delete Analog Input objects. The IUT shall not restrict the instance number which can be used to create the Analog Input object.

<b>135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values</b>		
	<b>Test Conditionality</b>	Must be tested on the Analog Input Object
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 8.17 - Delete Object Service</b>		
	<b>Test Conditionality</b>	Must be tested on the Analog Input Object
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 8.21.8 Can Create and Delete Analog Output Objects

The IUT can create and delete Analog Output objects. The IUT shall not restrict the instance number which can be used to create the Analog Output object.

<b>135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values</b>		
	<b>Test Conditionality</b>	Must be tested on the Analog Output Object
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 8.17 - Delete Object Service</b>		
	<b>Test Conditionality</b>	Must be tested on the Analog Output Object
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 8.21.9 Can Create and Delete Analog Value Objects

The IUT can create and delete Analog Value objects. The IUT shall not restrict the instance number which can be used to create the Analog Value object.

<b>135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values</b>		
	<b>Test Conditionality</b>	Must be tested on the Analog Value Object
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 8.17 - Delete Object Service</b>		
	<b>Test Conditionality</b>	Must be tested on the Analog Value Object
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 8.21.10 Can Create and Delete Averaging Objects

The IUT can create and delete Averaging objects. The IUT shall not restrict the instance number which can be used to create the Averaging object.

<b>135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values</b>		
	<b>Test Conditionality</b>	Must be tested on the Averaging Object
	<b>Test Directives</b>	

	Testing Hints	
<b>135.1-2013 - 8.17 - Delete Object Service</b>		
	Test Conditionality	Must be tested on the Averaging Object
	Test Directives	
	Testing Hints	

### 8.21.11 Can Create and Delete Binary Input Objects

The IUT can create and delete Binary Input objects. The IUT shall not restrict the instance number which can be used to create the Binary Input object.

<b>135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values</b>		
	Test Conditionality	Must be tested on the Binary Input Object
	Test Directives	
	Testing Hints	
<b>135.1-2013 - 8.17 - Delete Object Service</b>		
	Test Conditionality	Must be tested on the Binary Input Object
	Test Directives	
	Testing Hints	

### 8.21.12 Can Create and Delete Binary Output Objects

The IUT can create and delete Binary Output objects. The IUT shall not restrict the instance number which can be used to create the Binary Output object.

<b>135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values</b>		
	Test Conditionality	Must be tested on the Binary Output Object
	Test Directives	
	Testing Hints	
<b>135.1-2013 - 8.17 - Delete Object Service</b>		
	Test Conditionality	Must be tested on the Binary Output Object
	Test Directives	
	Testing Hints	

### 8.21.13 Can Create and Delete Binary Value Objects

The IUT can create and delete Binary Value objects. The IUT shall not restrict the instance number which can be used to create the Binary Value object.

<b>135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values</b>		
	Test Conditionality	Must be tested on the Binary Value Object
	Test Directives	
	Testing Hints	
<b>135.1-2013 - 8.17 - Delete Object Service</b>		
	Test Conditionality	Must be tested on the Binary Value Object
	Test Directives	
	Testing Hints	

### 8.21.14 Can Create and Delete Calendar Objects

The IUT can create and delete Calendar objects. The IUT shall not restrict the instance number which can be used to create the Calendar object.

<b>135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values</b>		
	Test Conditionality	Must be tested on the Calendar Object
	Test Directives	
	Testing Hints	

135.1-2013 - 8.17 - Delete Object Service		
	Test Conditionality	Must be tested on the Calendar Object
	Test Directives	
	Testing Hints	

### 8.21.15 Can Create and Delete Command Objects

The IUT can create and delete Command objects. The IUT shall not restrict the instance number which can be used to create the Command object.

135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values		
	Test Conditionality	Must be tested on the Command Object
	Test Directives	
	Testing Hints	
135.1-2013 - 8.17 - Delete Object Service		
	Test Conditionality	Must be tested on the Command Object
	Test Directives	
	Testing Hints	

### 8.21.16 Can Create and Delete Event Enrollment Objects

The IUT can create and delete Event Enrollment objects. The IUT shall not restrict the instance number which can be used to create the Event Enrollment object.

135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values		
	Test Conditionality	Must be tested on the Event Enrollment Object
	Test Directives	
	Testing Hints	
135.1-2013 - 8.17 - Delete Object Service		
	Test Conditionality	Must be tested on the Event Enrollment Object
	Test Directives	
	Testing Hints	

### 8.21.17 Can Create and Delete File Objects

The IUT can create and delete File objects. The IUT shall not restrict the instance number which can be used to create the File object. This option shall not be selected if the only way the IUT can create File objects is upon the initiation of the device restore procedure.

135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values		
	Test Conditionality	Must be tested on the File Object
	Test Directives	
	Testing Hints	
135.1-2013 - 8.17 - Delete Object Service		
	Test Conditionality	Must be tested on the File Object
	Test Directives	
	Testing Hints	

### 8.21.18 Can Create and Delete Group Objects

The IUT can create and delete Group objects. The IUT shall not restrict the instance number which can be used to create the Group object.

135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values		
	Test Conditionality	Must be tested on the Group Object
	Test Directives	
	Testing Hints	

<b>135.1-2013 - 8.17 - Delete Object Service</b>		
	<b>Test Conditionality</b>	Must be tested on the Group Object
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 8.21.19 Can Create and Delete Life Safety Point Objects

The IUT can create and delete Life Safety Point objects. The IUT shall not restrict the instance number which can be used to create the Life Safety Point object.

<b>135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values</b>		
	<b>Test Conditionality</b>	Must be tested on the Life Safety Point Object
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 8.17 - Delete Object Service</b>		
	<b>Test Conditionality</b>	Must be tested on the Life Safety Point Object
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 8.21.20 Can Create and Delete Life Safety Zone Objects

The IUT can create and delete Life Safety Zone objects. The IUT shall not restrict the instance number which can be used to create the Life Safety Zone object.

<b>135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values</b>		
	<b>Test Conditionality</b>	Must be tested on the Life Safety Zone Object
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 8.17 - Delete Object Service</b>		
	<b>Test Conditionality</b>	Must be tested on the Life Safety Zone Object
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 8.21.21 Can Create and Delete Loop Objects

The IUT can create and delete Loop objects. The IUT shall not restrict the instance number which can be used to create the Loop object.

<b>135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values</b>		
	<b>Test Conditionality</b>	Must be tested on the Loop Object
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 8.17 - Delete Object Service</b>		
	<b>Test Conditionality</b>	Must be tested on the Loop Object
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 8.21.22 Can Create and Delete Multi State Input Objects

The IUT can create and delete Multi State Input objects. The IUT shall not restrict the instance number which can be used to create the Multi State Input object.

<b>135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values</b>		
	<b>Test Conditionality</b>	Must be tested on the Multi State Input Object
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 8.17 - Delete Object Service</b>		



	<b>Test Conditionality</b>	Must be tested on the Multi State Input Object
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 8.21.23 Can Create and Delete Multi State Output Objects

The IUT can create and delete Multi State Output objects. The IUT shall not restrict the instance number which can be used to create the Multi State Output object.

<b>135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values</b>		
	<b>Test Conditionality</b>	Must be tested on the Multi State Output Object
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 8.17 - Delete Object Service</b>		
	<b>Test Conditionality</b>	Must be tested on the Multi State Output Object
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 8.21.24 Can Create and Delete Multi State Value Objects

The IUT can create and delete Multi State Value objects. The IUT shall not restrict the instance number which can be used to create the Multi State Value object.

<b>135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values</b>		
	<b>Test Conditionality</b>	Must be tested on the Multi State Value Object
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 8.17 - Delete Object Service</b>		
	<b>Test Conditionality</b>	Must be tested on the Multi State Value Object
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 8.21.25 Can Create and Delete Notification Class Objects

The IUT can create and delete Notification Class objects. The IUT shall not restrict the instance number which can be used to create the Notification Class object.

<b>135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values</b>		
	<b>Test Conditionality</b>	Must be tested on the Notification Class Object
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 8.17 - Delete Object Service</b>		
	<b>Test Conditionality</b>	Must be tested on the Notification Class Object
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 8.21.26 Can Create and Delete Program Objects

The IUT can create and delete Program objects. The IUT shall not restrict the instance number which can be used to create the Program object.

<b>135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values</b>		
	<b>Test Conditionality</b>	Must be tested on the Program Object
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 8.17 - Delete Object Service</b>		
	<b>Test Conditionality</b>	Must be tested on the Program Object
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 8.21.27 Can Create and Delete Pulse Converter Objects

The IUT can create and delete Pulse Converter objects. The IUT shall not restrict the instance number which can be used to create the Pulse Converter object.

<b>135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values</b>		
	<b>Test Conditionality</b>	Must be tested on the Pulse Converter Object
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 8.17 - Delete Object Service</b>		
	<b>Test Conditionality</b>	Must be tested on the Pulse Converter Object
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 8.21.28 Can Create and Delete Schedule Objects

The IUT can create and delete Schedule objects. The IUT shall not restrict the instance number which can be used to create the Schedule object.

<b>135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values</b>		
	<b>Test Conditionality</b>	Must be tested on the Schedule Object
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 8.17 - Delete Object Service</b>		
	<b>Test Conditionality</b>	Must be tested on the Schedule Object
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 8.21.29 Can Create and Delete Trend Log Objects

The IUT can create and delete Trend Log objects. The IUT shall not restrict the instance number which can be used to create the Trend Log object.

<b>135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values</b>		
	<b>Test Conditionality</b>	Must be tested on the Trend Log Object
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 8.17 - Delete Object Service</b>		
	<b>Test Conditionality</b>	Must be tested on the Trend Log Object
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 8.21.30 Can Create and Delete Structured View Objects

The IUT can create and delete Structured View objects. The IUT shall not restrict the instance number which can be used to create the Structured View object.

<b>135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values</b>		
	<b>Test Conditionality</b>	Must be tested on the Structured View Object
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 8.17 - Delete Object Service</b>		
	<b>Test Conditionality</b>	Must be tested on the Structured View Object
	<b>Test Directives</b>	

	Testing Hints	
--	---------------	--

### 8.21.31 Can Create and Delete Load Control Objects

The IUT can create and delete Load Control objects. The IUT shall not restrict the instance number which can be used to create the Load Control object.

135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values		
	Test Conditionality	Must be tested on the Load Control Object
	Test Directives	
	Testing Hints	
135.1-2013 - 8.17 - Delete Object Service		
	Test Conditionality	Must be tested on the Load Control Object
	Test Directives	
	Testing Hints	

### 8.21.32 Can Create and Delete Access Door Objects

The IUT can create and delete Access Door objects. The IUT shall not restrict the instance number which can be used to create the Access Door object.

135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values		
	Test Conditionality	Must be tested on the Access Door Object
	Test Directives	
	Testing Hints	
135.1-2013 - 8.17 - Delete Object Service		
	Test Conditionality	Must be tested on the Access Door Object
	Test Directives	
	Testing Hints	

### 8.21.33 Can Create and Delete Proprietary Objects

The IUT can create and delete Proprietary objects. The IUT shall not restrict the instance number which can be used to create Proprietary objects.

135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values		
	Test Conditionality	Must be tested on Proprietary Objects
	Test Directives	
	Testing Hints	
135.1-2013 - 8.17 - Delete Object Service		
	Test Conditionality	Must be tested on a Proprietary Object
	Test Directives	
	Testing Hints	

### 8.21.34 Can Create and Delete Event Log Objects

The IUT can create and delete Event Log objects. The IUT shall not restrict the instance number which can be used to create the Event Log object.

135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values		
	Test Conditionality	Must be tested on the Event Log Object
	Test Directives	
	Testing Hints	
135.1-2013 - 8.17 - Delete Object Service		
	Test Conditionality	Must be tested on the Event Log Object
	Test Directives	
	Testing Hints	

### 8.21.35 Can Create and Delete Trend Log Multiple Objects

The IUT can create and delete Trend Log Multiple objects. The IUT shall not restrict the instance number which can be used to create the Trend Log Multiple object.

135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values		
	Test Conditionality	Must be tested on the Trend Log Multiple Object
	Test Directives	
	Testing Hints	
135.1-2013 - 8.17 - Delete Object Service		
	Test Conditionality	Must be tested on the Trend Log Multiple Object
	Test Directives	
	Testing Hints	

### 8.21.36 Can Create and Delete CharacterString Value Objects

The IUT can create and delete CharacterString Value objects. The IUT shall not restrict the instance number which can be used to create the CharacterString Value object.

135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values		
	Test Conditionality	Must be tested on the CharacterString Value Object
	Test Directives	
	Testing Hints	
135.1-2013 - 8.17 - Delete Object Service		
	Test Conditionality	Must be tested on the CharacterString Value Object
	Test Directives	
	Testing Hints	

### 8.21.37 Can Create and Delete DateTime Value Objects

The IUT can create and delete DateTime Value objects. The IUT shall not restrict the instance number which can be used to create the DateTime Value object.

135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values		
	Test Conditionality	Must be tested on the DateTime Value Object
	Test Directives	
	Testing Hints	
135.1-2013 - 8.17 - Delete Object Service		
	Test Conditionality	Must be tested on the DateTime Value Object
	Test Directives	
	Testing Hints	

### 8.21.38 Can Create and Delete Large Analog Value Objects

The IUT can create and delete Large Analog Value objects. The IUT shall not restrict the instance number which can be used to create the Large Analog Value object.

135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values		
	Test Conditionality	Must be tested on the Large Analog Value Object
	Test Directives	
	Testing Hints	
135.1-2013 - 8.17 - Delete Object Service		
	Test Conditionality	Must be tested on the Large Analog Value Object
	Test Directives	
	Testing Hints	

### 8.21.39 Can Create and Delete BitString Value Objects

The IUT can create and delete BitString Value objects. The IUT shall not restrict the instance number which can be used to create the BitString Value object.

135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values		
	Test Conditionality	Must be tested on the BitString Value Object
	Test Directives	
	Testing Hints	
135.1-2013 - 8.17 - Delete Object Service		
	Test Conditionality	Must be tested on the BitString Value Object
	Test Directives	
	Testing Hints	

### 8.21.40 Can Create and Delete OctetString Value Objects

The IUT can create and delete OctetString Value objects. The IUT shall not restrict the instance number which can be used to create the OctetString Value object.

135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values		
	Test Conditionality	Must be tested on the OctetString Value Object
	Test Directives	
	Testing Hints	
135.1-2013 - 8.17 - Delete Object Service		
	Test Conditionality	Must be tested on the OctetString Value Object
	Test Directives	
	Testing Hints	

### 8.21.41 Can Create and Delete Time Value Objects

The IUT can create and delete Time Value objects. The IUT shall not restrict the instance number which can be used to create the Time Value object.

135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values		
	Test Conditionality	Must be tested on the Time Value Object
	Test Directives	
	Testing Hints	
135.1-2013 - 8.17 - Delete Object Service		
	Test Conditionality	Must be tested on the Time Value Object
	Test Directives	
	Testing Hints	

### 8.21.42 Can Create and Delete Integer Value Objects

The IUT can create and delete Integer Value objects. The IUT shall not restrict the instance number which can be used to create the Integer Value object.

135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values		
	Test Conditionality	Must be tested on the Integer Value Object
	Test Directives	
	Testing Hints	
135.1-2013 - 8.17 - Delete Object Service		
	Test Conditionality	Must be tested on the Integer Value Object
	Test Directives	
	Testing Hints	

### 8.21.43 Can Create and Delete Positive Integer Value Objects

The IUT can create and delete Positive Integer Value objects. The IUT shall not restrict the instance number which can be used to create the Positive Integer Value object.

135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values		
	Test Conditionality	Must be tested on the Positive Integer Value Object
	Test Directives	
	Testing Hints	
135.1-2013 - 8.17 - Delete Object Service		
	Test Conditionality	Must be tested on the Positive Integer Value Object
	Test Directives	
	Testing Hints	

### 8.21.44 Can Create and Delete Date Value Objects

The IUT can create and delete Date Value objects. The IUT shall not restrict the instance number which can be used to create the Date Value object.

135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values		
	Test Conditionality	Must be tested on the Date Value Object
	Test Directives	
	Testing Hints	
135.1-2013 - 8.17 - Delete Object Service		
	Test Conditionality	Must be tested on the Date Value Object
	Test Directives	
	Testing Hints	

### 8.21.45 Can Create and Delete DateTime Pattern Value Objects

The IUT can create and delete DateTime Pattern Value objects. The IUT shall not restrict the instance number which can be used to create the DateTime Pattern Value object.

135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values		
	Test Conditionality	Must be tested on the DateTime Pattern Value Object
	Test Directives	
	Testing Hints	
135.1-2013 - 8.17 - Delete Object Service		
	Test Conditionality	Must be tested on the DateTime Pattern Value Object
	Test Directives	
	Testing Hints	

### 8.21.46 Can Create and Delete Time Pattern Value Objects

The IUT can create and delete Time Pattern Value objects. The IUT shall not restrict the instance number which can be used to create the Time Pattern Value object.

135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values		
	Test Conditionality	Must be tested on the Time Pattern Value Object
	Test Directives	
	Testing Hints	
135.1-2013 - 8.17 - Delete Object Service		
	Test Conditionality	Must be tested on the Time Pattern Value Object
	Test Directives	
	Testing Hints	

### 8.21.47 Can Create and Delete Date Pattern Value Objects

The IUT can create and delete Date Pattern Value objects. The IUT shall not restrict the instance number which can be used to create the Date Pattern Value object.

135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values		
	Test Conditionality	Must be tested on the Date Pattern Value Object
	Test Directives	
	Testing Hints	
135.1-2013 - 8.17 - Delete Object Service		
	Test Conditionality	Must be tested on the Date Pattern Value Object
	Test Directives	
	Testing Hints	

### 8.21.48 Can Create and Delete Network Security Objects

The IUT can create and delete Network Security objects. The IUT shall not restrict the instance number which can be used to create the Network Security object.

135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values		
	Test Conditionality	Must be tested on the Network Security Object
	Test Directives	
	Testing Hints	
135.1-2013 - 8.17 - Delete Object Service		
	Test Conditionality	Must be tested on the Network Security Object
	Test Directives	
	Testing Hints	

### 8.21.49 Can Create and Delete Global Group Objects

The IUT can create and delete Global Group objects. The IUT shall not restrict the instance number which can be used to create the Global Group object.

135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values		
	Test Conditionality	Must be tested on the Global Group Object
	Test Directives	
	Testing Hints	
135.1-2013 - 8.17 - Delete Object Service		
	Test Conditionality	Must be tested on the Global Group Object
	Test Directives	
	Testing Hints	

### 8.21.50 Can Create and Delete Access Point Objects

The IUT can create and delete Access Point objects. The IUT shall not restrict the instance number which can be used to create the Access Point object.

135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values		
	Test Conditionality	Must be tested on the Access Point Object
	Test Directives	
	Testing Hints	
135.1-2013 - 8.17 - Delete Object Service		
	Test Conditionality	Must be tested on the Access Point Object
	Test Directives	
	Testing Hints	

### 8.21.51 Can Create and Delete Access Zone Objects

The IUT can create and delete Access Zone objects. The IUT shall not restrict the instance number which can be used to create the Access Zone object.

135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values		
	Test Conditionality	Must be tested on the Access Zone Object
	Test Directives	
	Testing Hints	
135.1-2013 - 8.17 - Delete Object Service		
	Test Conditionality	Must be tested on the Access Zone Object
	Test Directives	
	Testing Hints	

### 8.21.52 Can Create and Delete Access User Objects

The IUT can create and delete Access User objects. The IUT shall not restrict the instance number which can be used to create the Access User object.

135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values		
	Test Conditionality	Must be tested on the Access User Object
	Test Directives	
	Testing Hints	
135.1-2013 - 8.17 - Delete Object Service		
	Test Conditionality	Must be tested on the Access User Object
	Test Directives	
	Testing Hints	

### 8.21.53 Can Create and Delete Access Rights Objects

The IUT can create and delete Access Rights objects. The IUT shall not restrict the instance number which can be used to create the Access Rights object.

135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values		
	Test Conditionality	Must be tested on the Access Rights Object
	Test Directives	
	Testing Hints	
135.1-2013 - 8.17 - Delete Object Service		
	Test Conditionality	Must be tested on the Access Rights Object
	Test Directives	
	Testing Hints	

### 8.21.54 Can Create and Delete Access Credential Objects

The IUT can create and delete Access Credential objects. The IUT shall not restrict the instance number which can be used to create the Access Credential object.

135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values		
	Test Conditionality	Must be tested on the Access Credential Object
	Test Directives	
	Testing Hints	
135.1-2013 - 8.17 - Delete Object Service		
	Test Conditionality	Must be tested on the Access Credential Object
	Test Directives	



	Testing Hints	
--	---------------	--

### 8.21.55 Can Create and Delete Credential Data Objects

The IUT can create and delete Credential Data objects. The IUT shall not restrict the instance number which can be used to create the Credential Data object.

135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values		
	Test Conditionality	Must be tested on the Credential Data Object
	Test Directives	
	Testing Hints	
135.1-2013 - 8.17 - Delete Object Service		
	Test Conditionality	Must be tested on the Credential Data Object
	Test Directives	
	Testing Hints	

### 8.21.56 Can Create and Delete Notification Forwarder Objects

The IUT can create and delete Notification Forwarder objects. The IUT shall not restrict the instance number which can be used to create the Notification Forwarder object.

135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values		
	Test Conditionality	Must be tested on the Notification Forwarder Object
	Test Directives	
	Testing Hints	
135.1-2013 - 8.17 - Delete Object Service		
	Test Conditionality	Must be tested on the Notification Forwarder Object
	Test Directives	
	Testing Hints	

### 8.21.57 Can Create and Delete Alert Enrollment Objects

The IUT can create and delete Alert Enrollment objects. The IUT shall not restrict the instance number which can be used to create the Alert Enrollment object.

135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values		
	Test Conditionality	Must be tested on the Alert Enrollment Object
	Test Directives	
	Testing Hints	
135.1-2013 - 8.17 - Delete Object Service		
	Test Conditionality	Must be tested on the Alert Enrollment Object
	Test Directives	
	Testing Hints	

### 8.21.58 Can Create and Delete Channel Objects

The IUT can create and delete Channel objects. The IUT shall not restrict the instance number which can be used to create the Channel object.

135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values		
	Test Conditionality	Must be tested on the Channel Object
	Test Directives	
	Testing Hints	

135.1-2013 - 8.17 - Delete Object Service		
	Test Conditionality	Must be tested on the Channel Object
	Test Directives	
	Testing Hints	

### 8.21.59 Can Create and Delete Lighting Output Objects

The IUT can create and delete Lighting Output objects. The IUT shall not restrict the instance number which can be used to create the Lighting Output object.

135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values		
	Test Conditionality	Must be tested on the Lighting Output Object
	Test Directives	
	Testing Hints	
135.1-2013 - 8.17 - Delete Object Service		
	Test Conditionality	Must be tested on the Lighting Output Object
	Test Directives	
	Testing Hints	

### 8.21.60 Can Create and Delete Binary Lighting Output Objects

The IUT can create and delete Binary Lighting Output objects. The IUT shall not restrict the instance number which can be used to create the Binary Lighting Output object.

135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values		
	Test Conditionality	Must be tested on the Binary Lighting Output Object
	Test Directives	
	Testing Hints	
135.1-2013 - 8.17 - Delete Object Service		
	Test Conditionality	Must be tested on the Binary Lighting Output Object
	Test Directives	
	Testing Hints	

### 8.21.61 Can Create and Delete Network Port Objects

The IUT can create and delete Network Port objects. The IUT shall not restrict the instance number which can be used to create the Network Port object.

135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values		
	Test Conditionality	Must be tested on the Network Port Object
	Test Directives	
	Testing Hints	
135.1-2013 - 8.17 - Delete Object Service		
	Test Conditionality	Must be tested on the Network Port Object
	Test Directives	
	Testing Hints	

### 8.21.62 Can Create and Delete Timer Objects

The IUT can create and delete Timer objects. The IUT shall not restrict the instance number which can be used to create the Timer object.

135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values		
---	--	--

	<b>Test Conditionality</b>	Must be tested on the Timer Object
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 8.17 - Delete Object Service</b>		
	<b>Test Conditionality</b>	Must be tested on the Timer Object
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 8.21.63 Can Create and Delete Elevator Group Objects

The IUT can create and delete Elevator Group objects. The IUT shall not restrict the instance number which can be used to create the Elevator Group object.

<b>135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values</b>		
	<b>Test Conditionality</b>	Must be tested on the Elevator Group Object
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 8.17 - Delete Object Service</b>		
	<b>Test Conditionality</b>	Must be tested on the Elevator Group Object
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 8.21.64 Can Create and Delete Lift Objects

The IUT can create and delete Lift objects. The IUT shall not restrict the instance number which can be used to create the Lift object.

<b>135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values</b>		
	<b>Test Conditionality</b>	Must be tested on the Lift Object
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 8.17 - Delete Object Service</b>		
	<b>Test Conditionality</b>	Must be tested on the Lift Object
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 8.21.65 Can Create and Delete Escalator Objects

The IUT can create and delete Escalator objects. The IUT shall not restrict the instance number which can be used to create the Escalator object.

<b>135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values</b>		
	<b>Test Conditionality</b>	Must be tested on the Escalator Object
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 8.17 - Delete Object Service</b>		
	<b>Test Conditionality</b>	Must be tested on the Escalator Object
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 8.21.66 Can Create and Delete Staging Objects

The IUT can create and delete Staging objects. The IUT shall not restrict the instance number which can be used to create the Staging object.

135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values		
	Test Conditionality	Must be tested on the Staging Object
	Test Directives	
	Testing Hints	
135.1-2013 - 8.17 - Delete Object Service		
	Test Conditionality	Must be tested on the Staging Object
	Test Directives	
	Testing Hints	

### 8.21.67 Can Create and Delete Audit Reporter Objects

The IUT can create and delete Audit Reporter objects. The IUT shall not restrict the instance number which can be used to create the Audit Reporter object.

135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values		
	Test Conditionality	Must be tested on the Audit Reporter Object
	Test Directives	
	Testing Hints	
135.1-2013 - 8.17 - Delete Object Service		
	Test Conditionality	Must be tested on the Audit Reporter Object
	Test Directives	
	Testing Hints	

### 8.21.68 Can Create and Delete Audit Log Objects

The IUT can create and delete Audit Log objects. The IUT shall not restrict the instance number which can be used to create the Audit Log object.

135.1-2013 - 8.16.1 - Creating Objects by Specifying the Object Identifier with no Initial Values		
	Test Conditionality	Must be tested on the Audit Log Object
	Test Directives	
	Testing Hints	
135.1-2013 - 8.17 - Delete Object Service		
	Test Conditionality	Must be tested on the Audit Log Object
	Test Directives	
	Testing Hints	

## 8.22 Device Management - Object Creation and Deletion - B

### 8.22.1 Base Requirements

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

Verify Checklist		
	Test Conditionality	Must be executed.
	Test Directives	Verify that for each object the IUT claims object creation and deletion for in the checklist that there is a corresponding claim in the EPICS 'Standard Types Supported' section.
	Testing Hints	

### 8.22.2 Supports Object Creation Using Object\_Type

The IUT can create the specified object using Object Type with or without specifying initial values. The IUT accepts any valid initial values for modifiable properties. A valid value is any value in the range for the property as defined by the EPICS.

135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	This functionality will be tested in the specific object sections below.
135.1-2013 - 9.16.1.3 - Creating Objects by Specifying the Object Type and Providing Initial Values		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	The IUT accepts any valid initial values for modifiable properties.
135.1-2013 - 9.16.2.3 - Attempting to Create an Object with an Object Identifier That is Not Creatable by Specifying Object Identifier		
	Test Conditionality	This test shall be executed if the device does not support creating objects by object identifier.
	Test Directives	
	Testing Hints	
BTL - 9.16.2.4 - Attempting to Create an Object with an Object Type Specifier and an Error in the Initial Values		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
BTL - 9.16.2.X1 - Attempting to Create a non-Supported Object Type (by Object Type)		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 8.22.3 Supports Object Creation Using Object\_Identifier

The IUT can create the specified object using Object\_Identifier with or without specifying any initial values. When creating with initial values any valid properties for the object may be used.

BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	This functionality will be tested in the specific object sections below.
BTL - 9.16.1.4 - Creating Objects by Specifying the Object Identifier and Providing Initial Values		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

	Test Directives	
	Testing Hints	
<b>BTL - 9.16.2.1 - Attempting to Create an Object That Does Not Have a Unique Object Identifier</b>		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
<b>135.1-2013 - 9.16.2.2 - Attempting to Create an Object with an Object Type That is Not Creatable by Specifying Object Type</b>		
	Test Conditionality	If Object Type creation is not supported this test shall be run.
	Test Directives	
	Testing Hints	
<b>BTL - 9.16.2.5 - Attempting to Create an Object with an Object Identifier and an Error in the Initial Values</b>		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
<b>BTL - 9.16.2.X2 - Attempting to Create a non-Supported Object Type (by Object Identifier)</b>		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

#### 8.22.4 Supports Object Deletion for each Object Type that it Supports Creation Of

The IUT can delete an object.

<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	This functionality will be tested in the specific object sections below.
<b>135.1-2013 - 9.17.2.2 - Attempting to Delete an Object That Does Not Exist</b>		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
<b>BTL - 9.17.2.1 - Attempting to Delete an Object That is Not Deletable</b>		
	Test Conditionality	Must be executed.
	Test Directives	Apply this test once specifying the Device object. If there are any other objects which are not deletable, then also apply the test one or more times to other object types.
	Testing Hints	

#### 8.22.5 Supports Object Creation and Deletion of the Accumulator Object

The Accumulator object can be created and deleted within the IUT. The Accumulator object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	Test Conditionality	Must be executed.
	Test Directives	Execute using the Accumulator Object.
	Testing Hints	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	Test Conditionality	Must be executed.
	Test Directives	Execute using the Accumulator Object.
	Testing Hints	

<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Accumulator Object.
	<b>Testing Hints</b>	

### 8.22.6 Supports Object Creation and Deletion of the Analog Input Object

The Analog Input object can be created and deleted within the IUT. The Analog Input object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Analog Input Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Analog Input Object.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Analog Input Object.
	<b>Testing Hints</b>	

### 8.22.7 Supports Object Creation and Deletion of the Analog Output Object

The Analog Output object can be created and deleted within the IUT. The Analog Output object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Analog Output Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Analog Output Object.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Analog Output Object.
	<b>Testing Hints</b>	

### 8.22.8 Supports Object Creation and Deletion of the Analog Value Object

The Analog Value object can be created and deleted within the IUT. The Analog Value object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Analog Value Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.

	<b>Test Directives</b>	Execute using the Analog Value Object.
	<b>Testing Hints</b>	
	<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>	
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Analog Value Object.
	<b>Testing Hints</b>	

### 8.22.9 Supports Object Creation and Deletion of the Averaging Object

The Averaging object can be created and deleted within the IUT. The Averaging object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Averaging Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Averaging Object.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Averaging Object.
	<b>Testing Hints</b>	

### 8.22.10 Supports Object Creation and Deletion of the Binary Input Object

The Binary Input object can be created and deleted within the IUT. The Binary Input object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Binary Input Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Binary Input Object.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Binary Input Object.
	<b>Testing Hints</b>	

### 8.22.11 Supports Object Creation and Deletion of the Binary Output Object

The Binary Output object can be created and deleted within the IUT. The Binary Output object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Binary Output Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	



	<b>Test Directives</b>	Execute using the Binary Output Object.
	<b>Testing Hints</b>	
	<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>	
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Binary Output Object.
	<b>Testing Hints</b>	

### 8.22.12 Supports Object Creation and Deletion of the Binary Value Object

The Binary Value object can be created and deleted within the IUT. The Binary value object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Binary Value Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Binary Value Object.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Binary Value Object.
	<b>Testing Hints</b>	

### 8.22.13 Supports Object Creation and Deletion of the Calendar Object

The Calendar object can be created and deleted within the IUT. The Calendar object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Calendar Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Calendar Object.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Calendar Object.
	<b>Testing Hints</b>	

### 8.22.14 Supports Object Creation and Deletion of the Command Object

The Command object can be created and deleted within the IUT. The Command object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Command Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Command Object.

	<b>Testing Hints</b>	
<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Command Object.
	<b>Testing Hints</b>	

### 8.22.15 Supports Object Creation and Deletion of the Event Enrollment Object

The Event Enrollment object can be created and deleted within the IUT. The Event Enrollment object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Event Enrollment Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Event Enrollment Object.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Event Enrollment Object.
	<b>Testing Hints</b>	

### 8.22.16 Supports Object Creation and Deletion of the File Object

The File object can be created and deleted within the IUT. The File object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the File Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the File Object.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the File Object.
	<b>Testing Hints</b>	

### 8.22.17 Supports Object Creation and Deletion of the Group Object

The Group object can be created and deleted within the IUT. The Group object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Group Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Group Object.
	<b>Testing Hints</b>	

<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Group Object.
	<b>Testing Hints</b>	

## 8.22.18 Supports Object Creation and Deletion of the Life Safety Point Object

The Life Safety Point object can be created and deleted within the IUT. The Life Safety Point object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Life Safety Point Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Life Safety Point Object.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Life Safety Point Object.
	<b>Testing Hints</b>	

## 8.22.19 Supports Object Creation and Deletion of the Life Safety Zone Object

The Life Safety Zone object can be created and deleted within the IUT. The Life Safety Zone object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Life Safety Zone Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Life Safety Zone Object.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Life Safety Zone Object.
	<b>Testing Hints</b>	

## 8.22.20 Supports Object Creation and Deletion of the Loop Object

The Loop object can be created and deleted within the IUT. The Loop object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Loop Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Loop Object.
	<b>Testing Hints</b>	

<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Loop Object.
	<b>Testing Hints</b>	

### 8.22.21 Supports Object Creation and Deletion of the Multi State Input Object

The Multi State Input object can be created and deleted within the IUT. The Multi State Input object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Multi State Input Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Multi State Input Object.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Multi State Input Object.
	<b>Testing Hints</b>	

### 8.22.22 Supports Object Creation and Deletion of the Multi State Output Object

The Multi State Output object can be created and deleted within the IUT. The Multi State Output object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Multi State Output Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Multi State Output Object.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Multi State Output Object.
	<b>Testing Hints</b>	

### 8.22.23 Supports Object Creation and Deletion of the Multi State Value Object

The Multi State Value object can be created and deleted within the IUT. The Multi State Value object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Multi State Value Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.

	<b>Test Directives</b>	Execute using the Multi State Value Object.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Multi State Value Object.
	<b>Testing Hints</b>	

### 8.22.24 Supports Object Creation and Deletion of the Notification Class Object

The Notification Class object can be created and deleted within the IUT. The Notification Class object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Notification Class Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Notification Class Object.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Notification Class Object.
	<b>Testing Hints</b>	

### 8.22.25 Supports Object Creation and Deletion of the Program Object

The Program object can be created and deleted within the IUT. The Program object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Program Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Program Object.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Program Object.
	<b>Testing Hints</b>	

### 8.22.26 Supports Object Creation and Deletion of the Pulse Converter Object

The Pulse Converter object can be created and deleted within the IUT. The Pulse Converter object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Pulse Converter Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

	<b>Test Directives</b>	Execute using the Pulse Converter Object.
	<b>Testing Hints</b>	
	<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>	
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Pulse Converter Object.
	<b>Testing Hints</b>	

### 8.22.27 Supports Object Creation and Deletion of the Schedule Object

The Schedule object can be created and deleted within the IUT. The Schedule object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Schedule Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Schedule Object.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Schedule Object.
	<b>Testing Hints</b>	

### 8.22.28 Supports Object Creation and Deletion of the Trend Log Object

The Trend Log object can be created and deleted within the IUT. The Trend Log object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Trend Log Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Trend Log Object.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Trend Log Object.
	<b>Testing Hints</b>	

### 8.22.29 Supports Object Creation and Deletion of the Structured View Object

The Structured View object can be created and deleted within the IUT. The Structured View object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Structured View Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Structured View Object.

	Testing Hints	
<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	Test Conditionality	Must be executed.
	Test Directives	Execute using the Structured View Object.
	Testing Hints	

### 8.22.30 Supports Object Creation and Deletion of the Load Control Object

The Load Control object can be created and deleted within the IUT. The Load Control object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	Test Conditionality	Must be executed.
	Test Directives	Execute using the Load Control Object.
	Testing Hints	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	Test Conditionality	Must be executed.
	Test Directives	Execute using the Load Control Object.
	Testing Hints	
<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	Test Conditionality	Must be executed.
	Test Directives	Execute using the Load Control Object.
	Testing Hints	

### 8.22.31 Supports Object Creation and Deletion of the Access Door Object

The Access Door object can be created and deleted within the IUT. The Access Door object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	Test Conditionality	Must be executed.
	Test Directives	Execute using the Access Door Object.
	Testing Hints	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	Test Conditionality	Must be executed.
	Test Directives	Execute using the Access Door Object.
	Testing Hints	
<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	Test Conditionality	Must be executed.
	Test Directives	Execute using the Access Door Object.
	Testing Hints	

### 8.22.32 Supports Object Creation and Deletion of Proprietary Objects

Proprietary objects can be created and deleted within the IUT. The Proprietary object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	Test Conditionality	Must be executed.
	Test Directives	Execute using the Proprietary Object.
	Testing Hints	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	Test Conditionality	Must be executed.
	Test Directives	Execute using the Proprietary Object.

	<b>Testing Hints</b>	
<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Proprietary Object.
	<b>Testing Hints</b>	

### 8.22.33 Supports Object Creation and Deletion of the Event Log Object

The Event Log object can be created and deleted within the IUT. The Event Log object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Event Log Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Event Log Object.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Event Log Object.
	<b>Testing Hints</b>	

### 8.22.34 Supports Object Creation and Deletion of the Trend Log Multiple Object

The Trend Log Multiple object can be created and deleted within the IUT. The Trend Log Multiple object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Trend Log Multiple Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Trend Log Multiple Object.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Trend Log Multiple Object.
	<b>Testing Hints</b>	

### 8.22.35 Supports Object Creation and Deletion of the CharacterString Value Object

The CharacterString Value object can be created and deleted within the IUT. The CharacterString Value object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the CharacterString Value Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.



	<b>Test Directives</b>	Execute using the CharacterString Value Object.
	<b>Testing Hints</b>	
	<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>	
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the CharacterString Value Object.
	<b>Testing Hints</b>	

### 8.22.36 Supports Object Creation and Deletion of the DateTime Value Object

The DateTime Value object can be created and deleted within the IUT. The DateTime Value object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the DateTime Value Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the DateTime Value Object.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the DateTime Value Object.
	<b>Testing Hints</b>	

### 8.22.37 Supports Object Creation and Deletion of the Large Analog Value Object

The Large Analog Value object can be created and deleted within the IUT. The Large Analog Value object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Large Analog Value Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Large Analog Value Object.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Large Analog Value Object.
	<b>Testing Hints</b>	

### 8.22.38 Supports Object Creation and Deletion of the BitString Value Object

The BitString Value object can be created and deleted within the IUT. The BitString Value object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the BitString Value Object.
	<b>Testing Hints</b>	

<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the BitString Value Object.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the BitString Value Object.
	<b>Testing Hints</b>	

### 8.22.39 Supports Object Creation and Deletion of the OctetString Value Object

The OctetString Value object can be created and deleted within the IUT. The OctetString Value object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the OctetString Value Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the OctetString Value Object.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the OctetString Value Object.
	<b>Testing Hints</b>	

### 8.22.40 Supports Object Creation and Deletion of the Time Value Object

The Time Value object can be created and deleted within the IUT. The Time Value object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Time Value Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Time Value Object.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Time Value Object.
	<b>Testing Hints</b>	

### 8.22.41 Supports Object Creation and Deletion of the Integer Value Object

The Integer Value object can be created and deleted within the IUT. The Integer Value object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.

	<b>Test Directives</b>	Execute using the Integer Value Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Integer Value Object.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Integer Value Object.
	<b>Testing Hints</b>	

## 8.22.42 Supports Object Creation and Deletion of the Positive Integer Value Object

The Positive Integer Value object can be created and deleted within the IUT. The Positive Integer Value object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Positive Integer Value Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Positive Integer Value Object.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Positive Integer Value Object.
	<b>Testing Hints</b>	

## 8.22.43 Supports Object Creation and Deletion of the Date Value Object

The Date Value object can be created and deleted within the IUT. The Date Value object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Date Value Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Date Value Object.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Date Value Object.
	<b>Testing Hints</b>	

## 8.22.44 Supports Object Creation and Deletion of the DateTime Pattern Value Object

The DateTime Pattern Value object can be created and deleted within the IUT. The DateTime Pattern Value object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the DateTime Pattern Value Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the DateTime Pattern Value Object.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the DateTime Pattern Value Object.
	<b>Testing Hints</b>	

## 8.22.45 Supports Object Creation and Deletion of the Time Pattern Value Object

The Time Pattern Value object can be created and deleted within the IUT. The Time Pattern Value object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Time Pattern Value Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Time Pattern Value Object.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Time Pattern Value Object.
	<b>Testing Hints</b>	

## 8.22.46 Supports Object Creation and Deletion of the Date Pattern Value Object

The Date Pattern Value object can be created and deleted within the IUT. The Date Pattern Value object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Date Pattern Value Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Date Pattern Value Object.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Date Pattern Value Object.
	<b>Testing Hints</b>	

## 8.22.47 Supports Object Creation and Deletion of the Network Security Object

The Network Security object can be created and deleted within the IUT. The Network Security object that is created must be the object that can be deleted using the delete service.

135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values		
	Test Conditionality	Must be executed.
	Test Directives	Execute using the Network Security Object.
	Testing Hints	
BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values		
	Test Conditionality	Must be executed.
	Test Directives	Execute using the Network Security Object.
	Testing Hints	
135.1-2013 - 9.17.1.1 - Successful Deletion of an Object		
	Test Conditionality	Must be executed.
	Test Directives	Execute using the Network Security Object.
	Testing Hints	

## 8.22.48 Supports Object Creation and Deletion of the Global Group Object

The Global Group object can be created and deleted within the IUT. The Global Group object that is created must be the object that can be deleted using the delete service.

135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values		
	Test Conditionality	Must be executed.
	Test Directives	Execute using the Global Group Object.
	Testing Hints	
BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values		
	Test Conditionality	Must be executed.
	Test Directives	Execute using the Global Group Object.
	Testing Hints	
135.1-2013 - 9.17.1.1 - Successful Deletion of an Object		
	Test Conditionality	Must be executed.
	Test Directives	Execute using the Global Group Object.
	Testing Hints	

## 8.22.49 Supports Object Creation and Deletion of the Access Point Object

The Access Point object can be created and deleted within the IUT. The Access Point object that is created must be the object that can be deleted using the delete service.

135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values		
	Test Conditionality	Must be executed.
	Test Directives	Execute using the Access Point Object.
	Testing Hints	
BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values		
	Test Conditionality	Must be executed.
	Test Directives	Execute using the Access Point Object.
	Testing Hints	
135.1-2013 - 9.17.1.1 - Successful Deletion of an Object		
	Test Conditionality	Must be executed.
	Test Directives	Execute using the Access Point Object.

	Testing Hints	
--	---------------	--

### 8.22.50 Supports Object Creation and Deletion of the Access Zone Object

The Access Zone object can be created and deleted within the IUT. The Access Zone object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Access Zone Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Access Zone Object.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Access Zone Object.
	<b>Testing Hints</b>	

### 8.22.51 Supports Object Creation and Deletion of the Access User Object

The Access User object can be created and deleted within the IUT. The Access User object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Access User Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Access User Object.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Access User Object.
	<b>Testing Hints</b>	

### 8.22.52 Supports Object Creation and Deletion of the Access Rights Object

The Access Rights object can be created and deleted within the IUT. The Access Rights object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Access Rights Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.

	<b>Test Directives</b>	Execute using the Access Rights Object.
	<b>Testing Hints</b>	
	<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>	
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Access Rights Object.
	<b>Testing Hints</b>	

### 8.22.53 Supports Object Creation and Deletion of the Access Credential Object

The Access Credential object can be created and deleted within the IUT. The Access Credential object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Access Credential Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Access Credential Object.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Access Credential Object.
	<b>Testing Hints</b>	

### 8.22.54 Supports Object Creation and Deletion of the Credential Data Object

The Credential Data object can be created and deleted within the IUT. The Credential Data object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Credential Data Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Credential Data Object.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Credential Data Object.
	<b>Testing Hints</b>	

### 8.22.55 Supports Object Creation and Deletion of the Notification Forwarder Object

The Notification Forwarder object can be created and deleted within the IUT. The Notification Forwarder object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
--	--	--

	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Notification Forwarder Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Notification Forwarder Object.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Notification Forwarder Object.
	<b>Testing Hints</b>	

## 8.22.56 Supports Object Creation and Deletion of the Alert Enrollment Object

The Alert Enrollment object can be created and deleted within the IUT. The Alert Enrollment object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Alert Enrollment Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Alert Enrollment Object.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Alert Enrollment Object.
	<b>Testing Hints</b>	

## 8.22.57 Supports Object Creation and Deletion of the Channel Object

The Channel object can be created and deleted within the IUT. The Channel object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Channel Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Channel Object.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Channel Object.
	<b>Testing Hints</b>	



## 8.22.58 Supports Object Creation and Deletion of the Lighting Output Object

The Lighting Output object can be created and deleted within the IUT. The Lighting Output object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Lighting Output Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Lighting Output Object.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Lighting Output Object.
	<b>Testing Hints</b>	

## 8.22.59 Supports Object Creation and Deletion of the Binary Lighting Output Object

The Binary Lighting Output object can be created and deleted within the IUT. The Binary Lighting Output object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Binary Lighting Output Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Binary Lighting Output Object.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Binary Lighting Output Object.
	<b>Testing Hints</b>	

## 8.22.60 Supports Object Creation and Deletion of the Network Port Object

The Network Port object can be created and deleted within the IUT. The Network Port object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Network Port Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Network Port Object.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		

	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Network Port Object.
	<b>Testing Hints</b>	

### 8.22.61 Supports Object Creation and Deletion of the Timer Object

The Timer object can be created and deleted within the IUT. The Timer object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Timer Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Timer Object.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Timer Object.
	<b>Testing Hints</b>	

### 8.22.62 Supports Object Creation and Deletion of the Elevator Group Object

The Elevator Group object can be created and deleted within the IUT. The Elevator Group object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Elevator Group Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Elevator Group Object.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Elevator Group Object.
	<b>Testing Hints</b>	

### 8.22.63 Supports Object Creation and Deletion of the Lift Object

The Lift object can be created and deleted within the IUT. The Lift object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Lift Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Lift Object.

	<b>Testing Hints</b>	
<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Lift Object.
	<b>Testing Hints</b>	

### 8.22.64 Supports Object Creation and Deletion of the Escalator Object

The Escalator object can be created and deleted within the IUT. The Escalator object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Escalator Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Escalator Object.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Escalator Object.
	<b>Testing Hints</b>	

### 8.22.65 Supports Object Creation and Deletion of the Staging Object

The Staging object can be created and deleted within the IUT. The Staging object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Staging Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Staging Object.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Staging Object.
	<b>Testing Hints</b>	

### 8.22.66 Supports Object Creation and Deletion of the Audit Reporter Object

The Audit Reporter object can be created and deleted within the IUT. The Audit Reporter object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Audit Reporter Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		

	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Audit Reporter Object.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Audit Reporter Object.
	<b>Testing Hints</b>	

## 8.22.67 Supports Object Creation and Deletion of the Audit Log Object

The Audit Log object can be created and deleted within the IUT. The Audit Log object that is created must be the object that can be deleted using the delete service.

<b>135.1-2013 - 9.16.1.1 - Creating Objects by Specifying the Object Type with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Audit Log Object.
	<b>Testing Hints</b>	
<b>BTL - 9.16.1.2 - Creating Objects by Specifying the Object Identifier with No Initial Values</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Audit Log Object.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.17.1.1 - Successful Deletion of an Object</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Execute using the Audit Log Object.
	<b>Testing Hints</b>	

## 8.23 Device Management - List Manipulation - A

### 8.23.1 Base Requirements

Base requirements must be met by any IUT claiming conformance to this BIBB. There are no base requirements tests for this section.

### 8.23.2 Supports Adding and Removing Notification Class / Recipient\_List Entries

The IUT is capable of adding entries to, and removing entries from, Recipient\_List properties in Notification Class objects.

135.1-2013 - 8.14.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Recipient_List property of a Notification Class object.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.15.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Recipient_List property of a Notification Class object.
	Test Directives	
	Testing Hints	

### 8.23.3 Supports Adding and Removing Schedule / List\_Of\_Object\_Property\_References Entries

The IUT is capable of adding entries to, and removing entries from, List\_Of\_Object\_Property\_References properties in Schedule objects.

135.1-2013 - 8.14.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the List_Of_Object_Property_References property of a Schedule object.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.15.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the List_Of_Object_Property_References property of a Schedule object.
	Test Directives	
	Testing Hints	

### 8.23.4 Supports Adding and Removing Calendar / Date\_List Entries

The IUT is capable of adding entries to, and removing entries from, Date\_List properties in Calendar objects.

135.1-2013 - 8.14.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Date_List property of a Calendar object.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.15.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Date_List property of a Calendar object.
	Test Directives	

	Testing Hints	
--	---------------	--

### 8.23.5 Supports Adding and Removing Device / Time\_Synchronization\_Recipients Entries

The IUT is capable of adding entries to, and removing entries from, Time\_Synchronization\_Recipients properties in Device objects.

135.1-2013 - 8.14.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Time_Synchronization_Recipients property of a Device object.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.15.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Time_Synchronization_Recipients property of a Device object.
	Test Directives	
	Testing Hints	

### 8.23.6 Supports Adding and Removing Device / UTC\_Time\_Synchronization\_Recipients Entries

The IUT is capable of adding entries to, and removing entries from, UTC\_Time\_Synchronization\_Recipients properties in Device objects.

135.1-2013 - 8.14.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the UTC_Time_Synchronization_Recipients property of a Device object.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.15.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the UTC_Time_Synchronization_Recipients property of a Device object.
	Test Directives	
	Testing Hints	

### 8.23.7 Supports Adding and Removing Device / Restart\_Notification\_Recipients Entries

The IUT is capable of adding entries to, and removing entries from, Restart\_Notification\_Recipients properties in Device objects.

135.1-2013 - 8.14.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Restart_Notification_Recipients property of a Device object.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.15.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Restart_Notification_Recipients property of a Device object.
	Test Directives	
	Testing Hints	

### 8.23.8 Supports Adding and Removing Device / Device\_Address\_Binding Entries

The IUT is capable of adding entries to, and removing entries from, Device\_Address\_Binding properties in Device objects.

135.1-2013 - 8.14.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Device_Address_Binding property of a Device object.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.15.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Device_Address_Binding property of a Device object.
	Test Directives	
	Testing Hints	

### 8.23.9 Supports Adding and Removing Device / Active\_COV\_Subscriptions Entries

The IUT is capable of adding entries to, and removing entries from, Active\_COV\_Subscriptions properties in Device objects.

135.1-2013 - 8.14.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Active_COV_Subscriptions property of a Device object.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.15.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Active_COV_Subscriptions property of a Device object.
	Test Directives	
	Testing Hints	

### 8.23.10 Supports Adding and Removing Device / Manual\_Slave\_Address\_Binding Entries

The IUT is capable of adding entries to, and removing entries from, Manual\_Slave\_Address\_Binding properties in Device objects.

135.1-2013 - 8.14.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Manual_Slave_Address_Binding property of a Device object.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.15.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Manual_Slave_Address_Binding property of a Device object.
	Test Directives	
	Testing Hints	

### 8.23.11 Supports Adding and Removing Device / Slave\_Address\_Binding Entries

The IUT is capable of adding entries to, and removing entries from, Slave\_Address\_Binding properties in Device objects.

135.1-2013 - 8.14.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Slave_Address_Binding property of a Device object.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.15.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Slave_Address_Binding property of a Device object.
	Test Directives	
	Testing Hints	

### 8.23.12 Supports Adding and Removing Group / List\_Of\_Group\_Members Entries

The IUT is capable of adding entries to, and removing entries from, List\_Of\_Group\_Members properties in Group objects.

135.1-2013 - 8.14.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the List_Of_Group_Members property of a Group object.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.15.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the List_Of_Group_Members property of a Group object.
	Test Directives	
	Testing Hints	

### 8.23.13 Supports Adding and Removing Life Safety Point / Life\_Safety\_Alarm\_Values Entries

The IUT is capable of adding entries to, and removing entries from, Life\_Safety\_Alarm\_Values properties in Life Safety Point objects.

135.1-2013 - 8.14.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Life_Safety_Alarm_Values property of a Life Safety Point object.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.15.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Life_Safety_Alarm_Values property of a Life Safety Point object.
	Test Directives	
	Testing Hints	



### 8.23.14 Supports Adding and Removing Life Safety Point / Alarm\_Values Entries

The IUT is capable of adding entries to, and removing entries from, Alarm\_Values properties in Life Safety Point objects.

135.1-2013 - 8.14.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Alarm_Values property of a Life Safety Point object.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.15.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Alarm_Values property of a Life Safety Point object.
	Test Directives	
	Testing Hints	

### 8.23.15 Supports Adding and Removing Life Safety Point / Fault\_Values Entries

The IUT is capable of adding entries to, and removing entries from, Fault\_Values properties in Life Safety Point objects.

135.1-2013 - 8.14.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Fault_Values property of a Life Safety Point object.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.15.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Fault_Values property of a Life Safety Point object.
	Test Directives	
	Testing Hints	

### 8.23.16 Supports Adding and Removing Life Safety Point / Member\_Of Entries

The IUT is capable of adding entries to, and removing entries from, Member\_Of properties in Life Safety Point objects.

135.1-2013 - 8.14.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Member_Of property of a Life Safety Point object.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.15.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Member_Of property of a Life Safety Point object.
	Test Directives	
	Testing Hints	

### 8.23.17 Supports Adding and Removing Life Safety Zone / Life\_Safety\_Alarm\_Values Entries

The IUT is capable of adding entries to, and removing entries from, Life\_Safety\_Alarm\_Values properties in Life Safety Zone objects.

135.1-2013 - 8.14.1 - Non-Array Properties		
	<b>Test Conditionality</b>	Must be executed with the IUT modifying the Life_Safety_Alarm_Values property of a Life Safety Zone object.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
135.1-2013 - 8.15.1 - Non-Array Properties		
	<b>Test Conditionality</b>	Must be executed with the IUT modifying the Life_Safety_Alarm_Values property of a Life Safety Zone object.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 8.23.18 Supports Adding and Removing Life Safety Zone / Alarm\_Values Entries

The IUT is capable of adding entries to, and removing entries from, Alarm\_Values properties in Life Safety Zone objects.

135.1-2013 - 8.14.1 - Non-Array Properties		
	<b>Test Conditionality</b>	Must be executed with the IUT modifying the Alarm_Values property of a Life Safety Zone object.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
135.1-2013 - 8.15.1 - Non-Array Properties		
	<b>Test Conditionality</b>	Must be executed with the IUT modifying the Alarm_Values property of a Life Safety Zone object.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 8.23.19 Supports Adding and Removing Life Safety Zone / Fault\_Values Entries

The IUT is capable of adding entries to, and removing entries from, Fault\_Values properties in Life Safety Zone objects.

135.1-2013 - 8.14.1 - Non-Array Properties		
	<b>Test Conditionality</b>	Must be executed with the IUT modifying the Fault_Values property of a Life Safety Zone object.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
135.1-2013 - 8.15.1 - Non-Array Properties		
	<b>Test Conditionality</b>	Must be executed with the IUT modifying the Fault_Values property of a Life Safety Zone object.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 8.23.20 Supports Adding and Removing Life Safety Zone / Member\_Of Entries

The IUT is capable of adding entries to, and removing entries from, Member\_Of properties in Life Safety Zone objects.

135.1-2013 - 8.14.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Member_Of property of a Life Safety Zone object.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.15.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Member_Of property of a Life Safety Zone object.
	Test Directives	
	Testing Hints	

### 8.23.21 Supports Adding and Removing Multi-State Input / Alarm\_Values Entries

The IUT is capable of adding entries to, and removing entries from, Alarm\_Values properties in Multi-State Input objects.

135.1-2013 - 8.14.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Alarm_Values property of a Multi-State Input object.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.15.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Alarm_Values property of a Multi-State Input object.
	Test Directives	
	Testing Hints	

### 8.23.22 Supports Adding and Removing Multi-State Input / Fault\_Values Entries

The IUT is capable of adding entries to, and removing entries from, Fault\_Values properties in Multi-State Input objects.

135.1-2013 - 8.14.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Fault_Values property of a Multi-State Input object.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.15.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Fault_Values property of a Multi-State Input object.
	Test Directives	
	Testing Hints	

### 8.23.23 Supports Adding and Removing Multi-State Value / Alarm\_Values Entries

The IUT is capable of adding entries to, and removing entries from, Alarm\_Values properties in Multi-State Value objects.

135.1-2013 - 8.14.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Alarm_Values property of a Multi-State Value object.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.15.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Alarm_Values property of a Multi-State Value object.
	Test Directives	
	Testing Hints	

### 8.23.24 Supports Adding and Removing Multi-State Value / Fault\_Values Entries

The IUT is capable of adding entries to, and removing entries from, Fault\_Values properties in Multi-State Value objects.

135.1-2013 - 8.14.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Fault_Values property of a Multi-State Value object.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.15.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Fault_Values property of a Multi-State Value object.
	Test Directives	
	Testing Hints	

### 8.23.25 Supports Adding and Removing Entries in Proprietary List Properties of Primitive Datatypes

The IUT is capable of adding entries to, and removing entries from, arbitrary proprietary properties that contain elements of a primitive datatype.

135.1-2013 - 8.14.1 - Non-Array Properties		
	Test Conditionality	Must be executed once with the IUT modifying a proprietary property in a standard object and once with the IUT modifying a proprietary property in a proprietary object.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.15.1 - Non-Array Properties		
	Test Conditionality	Must be executed once with the IUT modifying a proprietary property in a standard object and once with the IUT modifying a proprietary property in a proprietary object.
	Test Directives	
	Testing Hints	

### 8.23.26 Supports Adding and Removing Global Group / COVU\_Recipients Entries

The IUT is capable of adding entries to, and removing entries from, COVU\_Recipients properties in Global Group objects.

135.1-2013 - 8.14.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the COVU_Recipients property of a Global Group object.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.15.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the COVU_Recipients property of a Global Group object.
	Test Directives	
	Testing Hints	

### 8.23.27 Supports Adding and Removing Device / VT\_Classes\_Supported Entries

The IUT is capable of adding entries to, and removing entries from, VT\_Classes\_Supported properties in Device objects.

135.1-2013 - 8.14.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the VT_Classes_Supported property of a Device object.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.15.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the VT_Classes_Supported property of a Device object.
	Test Directives	
	Testing Hints	

### 8.23.28 Supports Adding and Removing Device / Active\_VT\_Sessions Entries

The IUT is capable of adding entries to, and removing entries from, Active\_VT\_Sessions properties in Device objects.

135.1-2013 - 8.14.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Active_VT_Sessions property of a Device object.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.15.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Active_VT_Sessions property of a Device object.
	Test Directives	
	Testing Hints	

### 8.23.29 Supports Adding and Removing Life Safety Zone / Zone\_Members Entries

The IUT is capable of adding entries to, and removing entries from, Zone\_Members properties in Life Safety Zone objects.

135.1-2013 - 8.14.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Zone_Members property of a Life Safety Zone object.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.15.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Zone_Members property of a Life Safety Zone object.
	Test Directives	
	Testing Hints	

### 8.23.30 Supports Adding and Removing Notification Forwarder / Recipient\_List Entries

The IUT is capable of adding entries to, and removing entries from, Recipient\_List properties in Notification Forwarder objects.

135.1-2013 - 8.14.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Recipient_List property of a Notification Forwarder object.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.15.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Recipient_List property of a Notification Forwarder object.
	Test Directives	
	Testing Hints	

### 8.23.31 Supports Adding and Removing Notification Forwarder / Subscribed\_Recipients Entries

The IUT is capable of adding entries to, and removing entries from, Subscribed\_Recipients properties in Notification Forwarder objects.

135.1-2013 - 8.14.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Subscribed_Recipients property of a Notification Forwarder object.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.15.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Subscribed_Recipients property of a Notification Forwarder object.
	Test Directives	
	Testing Hints	

### 8.23.32 Supports Adding and Removing Access Door / Masked\_Alarm\_Values Entries

The IUT is capable of adding entries to, and removing entries from, Masked\_Alarm\_Values properties in Access Door objects.

135.1-2013 - 8.14.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Masked_Alarm_Values property of a Access Door object.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.15.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Masked_Alarm_Values property of a Access Door object.
	Test Directives	
	Testing Hints	

### 8.23.33 Supports Adding and Removing Access Door / Alarm\_Values Entries

The IUT is capable of adding entries to, and removing entries from, Alarm\_Values properties in Access Door objects.

135.1-2013 - 8.14.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Alarm_Values property of a Access Door object.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.15.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Alarm_Values property of a Access Door object.
	Test Directives	
	Testing Hints	

### 8.23.34 Supports Adding and Removing Access Door / Fault\_Values Entries

The IUT is capable of adding entries to, and removing entries from, Fault\_Values properties in Access Door objects.

135.1-2013 - 8.14.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Fault_Values property of a Access Door object.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.15.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Fault_Values property of a Access Door object.
	Test Directives	
	Testing Hints	

### 8.23.35 Supports Adding and Removing Access Point / Access\_Alarm\_Events Entries

The IUT is capable of adding entries to, and removing entries from, Access\_Alarm\_Events properties in Access Point objects.

135.1-2013 - 8.14.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Access_Alarm_Events property of a Access Point object.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.15.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Access_Alarm_Events property of a Access Point object.
	Test Directives	
	Testing Hints	

### 8.23.36 Supports Adding and Removing Access Point / Access\_Transaction\_Events Entries

The IUT is capable of adding entries to, and removing entries from, Access\_Transaction\_Events properties in Access Point objects.

135.1-2013 - 8.14.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Access_Transaction_Events property of a Access Point object.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.15.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Access_Transaction_Events property of a Access Point object.
	Test Directives	
	Testing Hints	

### 8.23.37 Supports Adding and Removing Access Point / Failed\_Attempt\_Events Entries

The IUT is capable of adding entries to, and removing entries from, Failed\_Attempt\_Events properties in Access Point objects.

135.1-2013 - 8.14.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Failed_Attempt_Events property of a Access Point object.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.15.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Failed_Attempt_Events property of a Access Point object.
	Test Directives	
	Testing Hints	



### 8.23.38 Supports Adding and Removing Access Zone / Credentials\_In\_Zone Entries

The IUT is capable of adding entries to, and removing entries from, Credentials\_In\_Zone properties in Access Zone objects.

135.1-2013 - 8.14.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Credentials_In_Zone property of a Access Zone object.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.15.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Credentials_In_Zone property of a Access Zone object.
	Test Directives	
	Testing Hints	

### 8.23.39 Supports Adding and Removing Access Zone / Entry\_Points Entries

The IUT is capable of adding entries to, and removing entries from, Entry\_Points properties in Access Zone objects.

135.1-2013 - 8.14.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Entry_Points property of a Access Zone object.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.15.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Entry_Points property of a Access Zone object.
	Test Directives	
	Testing Hints	

### 8.23.40 Supports Adding and Removing Access Zone / Exit\_Points Entries

The IUT is capable of adding entries to, and removing entries from, Exit\_Points properties in Access Zone objects.

135.1-2013 - 8.14.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Exit_Points property of a Access Zone object.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.15.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Exit_Points property of a Access Zone object.
	Test Directives	
	Testing Hints	

### 8.23.41 Supports Adding and Removing Access Zone / Alarm\_Values Entries

The IUT is capable of adding entries to, and removing entries from, Alarm\_Values properties in Access Zone objects.

135.1-2013 - 8.14.1 - Non-Array Properties		
	<b>Test Conditionality</b>	Must be executed with the IUT modifying the Alarm_Values property of a Access Zone object.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
135.1-2013 - 8.15.1 - Non-Array Properties		
	<b>Test Conditionality</b>	Must be executed with the IUT modifying the Alarm_Values property of a Access Zone object.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 8.23.42 Supports Adding and Removing Access User / Members Entries

The IUT is capable of adding entries to, and removing entries from, Members properties in Access User objects.

135.1-2013 - 8.14.1 - Non-Array Properties		
	<b>Test Conditionality</b>	Must be executed with the IUT modifying the Members property of a Access User object.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
135.1-2013 - 8.15.1 - Non-Array Properties		
	<b>Test Conditionality</b>	Must be executed with the IUT modifying the Members property of a Access User object.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 8.23.43 Supports Adding and Removing Access User / Member\_Of Entries

The IUT is capable of adding entries to, and removing entries from, Member\_Of properties in Access User objects.

135.1-2013 - 8.14.1 - Non-Array Properties		
	<b>Test Conditionality</b>	Must be executed with the IUT modifying the Member_Of property of a Access User object.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
135.1-2013 - 8.15.1 - Non-Array Properties		
	<b>Test Conditionality</b>	Must be executed with the IUT modifying the Member_Of property of a Access User object.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 8.23.44 Supports Adding and Removing Access User / Credentials Entries

The IUT is capable of adding entries to, and removing entries from, Credentials properties in Access User objects.

135.1-2013 - 8.14.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Credentials property of a Access User object.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.15.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Credentials property of a Access User object.
	Test Directives	
	Testing Hints	

### 8.23.45 Supports Adding and Removing Access Credential / Authorization\_Exemptions Entries

The IUT is capable of adding entries to, and removing entries from, Authorization\_Exemptions properties in Access Credential objects.

135.1-2013 - 8.14.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Authorization_Exemptions property of a Access Credential object.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.15.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Authorization_Exemptions property of a Access Credential object.
	Test Directives	
	Testing Hints	

### 8.23.46 Supports Adding and Removing Access Credential / Reason\_For\_Disable Entries

The IUT is capable of adding entries to, and removing entries from, Reason\_For\_Disable properties in Access Credential objects.

135.1-2013 - 8.14.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Reason_For_Disable property of a Access Credential object.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.15.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Reason_For_Disable property of a Access Credential object.
	Test Directives	
	Testing Hints	

### 8.23.47 Supports Adding and Removing Network Port / BBMD\_Broadcast\_Distribution\_Table Entries

The IUT is capable of adding entries to, and removing entries from, BBMD\_Broadcast\_Distribution\_Table properties in Network Port objects.

135.1-2013 - 8.14.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the BBMD_Broadcast_Distribution_Table property of a Network Port object.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.15.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the BBMD_Broadcast_Distribution_Table property of a Network Port object.
	Test Directives	
	Testing Hints	

### 8.23.48 Supports Adding and Removing Network Port / Manual\_Slave\_Address\_Binding Entries

The IUT is capable of adding entries to, and removing entries from, Manual\_Slave\_Address\_Binding properties in Network Port objects.

135.1-2013 - 8.14.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Manual_Slave_Address_Binding property of a Network Port object.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.15.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Manual_Slave_Address_Binding property of a Network Port object.
	Test Directives	
	Testing Hints	

### 8.23.49 Supports Adding and Removing Network Port / Virtual\_MAC\_Address\_Table Entries

The IUT is capable of adding entries to, and removing entries from, Virtual\_MAC\_Address\_Table properties in Network Port objects.

135.1-2013 - 8.14.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Virtual_MAC_Address_Table property of a Network Port object.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.15.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Virtual_MAC_Address_Table property of a Network Port object.
	Test Directives	
	Testing Hints	

### 8.23.50 Supports Adding and Removing Network Port / Slave\_Address\_Binding Entries

The IUT is capable of adding entries to, and removing entries from, Slave\_Address\_Binding properties in Network Port objects.

135.1-2013 - 8.14.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Slave_Address_Binding property of a Network Port object.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.15.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Slave_Address_Binding property of a Network Port object.
	Test Directives	
	Testing Hints	

### 8.23.51 Supports Adding and Removing Timer / List\_Of\_Object\_Property\_References Entries

The IUT is capable of adding entries to, and removing entries from, List\_Of\_Object\_Property\_References properties in Timer objects.

135.1-2013 - 8.14.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the List_Of_Object_Property_References property of a Timer object.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.15.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the List_Of_Object_Property_References property of a Timer object.
	Test Directives	
	Testing Hints	

### 8.23.52 Supports Adding and Removing Timer / Alarm\_Values Entries

The IUT is capable of adding entries to, and removing entries from, Alarm\_Values properties in Timer objects.

135.1-2013 - 8.14.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Alarm_Values property of a Timer object.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.15.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Alarm_Values property of a Timer object.
	Test Directives	
	Testing Hints	

### 8.23.53 Supports Adding and Removing Device / Active\_COV\_Multiple\_Subscriptions Entries

The IUT is capable of adding entries to, and removing entries from, Active\_COV\_Multiple\_Subscriptions properties in Device objects.

135.1-2013 - 8.14.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Active_COV_Multiple_Subscriptions property of a Device object.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.15.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Active_COV_Multiple_Subscriptions property of a Device object.
	Test Directives	
	Testing Hints	

### 8.23.54 Supports Adding and Removing Lift / Fault\_Signals Entries

The IUT is capable of adding entries to, and removing entries from, Fault\_Signals properties in Lift objects.

135.1-2013 - 8.14.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Fault_Signals property of a Lift object.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.15.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Fault_Signals property of a Lift object.
	Test Directives	
	Testing Hints	

### 8.23.55 Supports Adding and Removing Escalator / Fault\_Signals Entries

The IUT is capable of adding entries to, and removing entries from, Fault\_Signals properties in Escalator objects.

135.1-2013 - 8.14.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Fault_Signals property of a Escalator object.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.15.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Fault_Signals property of a Escalator object.
	Test Directives	
	Testing Hints	

### 8.23.56 Supports Adding and Removing Elevator Group / Landing\_Calls Entries

The IUT is capable of adding entries to, and removing entries from, Landing\_Calls properties in Elevator Group objects.

135.1-2013 - 8.14.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Landing_Calls property of a Elevator Group object.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.15.1 - Non-Array Properties		
	Test Conditionality	Must be executed with the IUT modifying the Landing_Calls property of a Elevator Group object.
	Test Directives	
	Testing Hints	

## 8.24 Device Management - List Manipulation - B

### 8.24.1 Base Requirements

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

A device that supports DM-LM-B is required to support AddListElement and RemoveListElement on every writable list property. In addition, any list property modifiable via AddListElement shall be modifiable via WriteProperty.

<b>135.1-2013 - 9.14.1.3 - Adding a Redundant Element</b>		
	<b>Test Conditionality</b>	
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.14.2.1 - Adding a List Element to a Property That is Not a List</b>		
	<b>Test Conditionality</b>	If there is no writable property that is not a list, then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	Test against array and non-array properties that do not contain lists. Test against array properties but do not include an array index.
<b>BTL - 9.14.2.2 - Adding a List Element With an Invalid Datatype</b>		
	<b>Test Conditionality</b>	
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 9.14.2.3 - An AddListElement Failure Part Way Through a List</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.15.2.1 - Removing a List Element from a Property That is Not a List</b>		
	<b>Test Conditionality</b>	If there is no writable property that is not a list, then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	Test against array and non-array properties that do not contain lists. Test against array properties but do not include an array index.
<b>BTL - 9.15.2.2 - A RemoveListElement Failure Part Way Through a List</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 8.24.2 All Writable List Properties in the IUT Support List Manipulation

The IUT supports AddListElement and DeleteListElement modification of all writable list properties in the device.

<b>135.1-2013 - 9.14.1.1 - Adding a Single Element</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Repeat for an instance of each type of object that contains a writable list property.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.14.1.2 - Adding Multiple Elements</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Repeat for an instance of each type of object that contains a writable list property.



	<b>Testing Hints</b>	
<b>135.1-2013 - 9.15.1.1 - Removing a Single Element from a List</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Repeat for an instance of each type of object that contains a writable list property.
	<b>Testing Hints</b>	
<b>135.1-2013 - 9.15.1.2 - Removing Multiple Elements from a List</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Repeat for an instance of each type of object that contains a writable list property.
	<b>Testing Hints</b>	

### 8.24.3 Supports DS-WP-B

Any device that contains a writable list property and that supports the list manipulation routines must also support the Write Property service.

<b>Verify Checklist</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Verify the IUT claims support for DS-WP-B.
	<b>Testing Hints</b>	

## 8.25 Device Management - Text Message - A

### 8.25.1 Base Requirements

Base requirements must be met by any IUT claiming conformance to this BIBB. There are no base requirements tests for this section.

### 8.25.2 Supports Initiation of ConfirmedTextMessage

The IUT is capable of initiating ConfirmedTextMessage services.

This functionality is tested by one of the Message Class test sections.

### 8.25.3 Supports Initiation of UnconfirmedTextMessage

The IUT is capable of initiating UnconfirmedTextMessage services.

This functionality is tested by one of the Message Class test sections.

### 8.25.4 Initiates Text Messages with no Message Class

The IUT is capable of initiating a text message service with no Message Class.

135.1-2013 - 8.28.1 - Text Message with no Message Class		
	Test Conditionality	Must be executed if the IUT supports initiation of ConfirmedTextMessage services.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.29.1 - Text Message with no Message Class		
	Test Conditionality	Must be executed if the IUT supports initiation of UnconfirmedTextMessage services.
	Test Directives	
	Testing Hints	

### 8.25.5 Initiates Text Messages with an Unsigned Message Class

The IUT is capable of initiating a text message service with an Unsigned Message Class.

135.1-2013 - 8.28.2 - Text Message with an Unsigned Message Class		
	Test Conditionality	Must be executed if the IUT supports initiation of ConfirmedTextMessage services.
	Test Directives	
	Testing Hints	
135.1-2013 - 8.29.2 - Text Message with an Unsigned Message Class		
	Test Conditionality	Must be executed if the IUT supports initiation of UnconfirmedTextMessage services.
	Test Directives	
	Testing Hints	

### 8.25.6 Initiates Text Messages with a CharacterString Message Class

The IUT is capable of initiating a text message service with an CharacterString Message Class.

135.1-2013 - 8.28.3 - Text Message with a CharacterString Message Class		
	Test Conditionality	Must be executed if the IUT supports initiation of ConfirmedTextMessage services.
	Test Directives	

# BTL Test Plan

	<b>Testing Hints</b>	
<b>135.1-2013 - 8.29.3 - Text Message with a CharacterString Message Class</b>		
	<b>Test Conditionality</b>	Must be executed if the IUT supports initiation of UnconfirmedTextMessage services.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

## 8.26 Device Management - Text Message - B

### 8.26.1 Base Requirements

Base requirements must be met by any IUT claiming conformance to this BIBB. There are no base requirements tests for this section.

### 8.26.2 Supports Execution of ConfirmedTextMessage

The IUT is capable of execution of ConfirmedTextMessage services.

This functionality is tested by one of the Message Class test sections.

### 8.26.3 Supports Execution of UnconfirmedTextMessage

The IUT is capable of execution of UnconfirmedTextMessage services.

This functionality is tested by one of the Message Class test sections.

### 8.26.4 Executes Text Messages with no Message Class

The IUT is capable of executing a text message service with no Message Class.

135.1-2013 - 9.28.1 - Text Message with no Message Class		
	Test Conditionality	Must be executed if the IUT supports execution of ConfirmedTextMessage services.
	Test Directives	
	Testing Hints	
BTL - 9.29.1 - UnconfirmedTextMessage with no Message Class		
	Test Conditionality	Must be executed if the IUT supports execution of UnconfirmedTextMessage services.
	Test Directives	
	Testing Hints	

### 8.26.5 Executes Text Messages with an Unsigned Message Class

The IUT is capable of executing a text message service with an Unsigned Message Class.

135.1-2013 - 9.28.2 - Text Message with an Unsigned Message Class		
	Test Conditionality	Must be executed if the IUT supports execution of ConfirmedTextMessage services.
	Test Directives	
	Testing Hints	
BTL - 9.29.2 - UnconfirmedTextMessage with an Unsigned Message Class		
	Test Conditionality	Must be executed if the IUT supports execution of UnconfirmedTextMessage services.
	Test Directives	
	Testing Hints	

### 8.26.6 Executes Text Messages with a CharacterString Message Class

The IUT is capable of executing a text message service with an CharacterString Message Class.

135.1-2013 - 9.28.3 - Text Message with a CharacterString Message Class		
	Test Conditionality	Must be executed if the IUT supports execution of ConfirmedTextMessage services.
	Test Directives	

	<b>Testing Hints</b>	
<b>BTL - 9.29.3 - UnconfirmedTextMessage with a CharacterString Message Class</b>		
	<b>Test Conditionality</b>	Must be executed if the IUT supports execution of UnconfirmedTextMessage services.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

---

## **8.27 Device Management - Virtual Terminal - A**

---

### **8.27.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

---

## **8.28 Device Management - Virtual Terminal - B**

---

### **8.28.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

---

## **8.29 Device Management - Slave Proxy - View and Modify - A**

---

### **8.29.1 Base Requirements**

Contact BTL for interim tests for this BIBB.



---

## **8.30 Device Management - Slave Proxy - B**

---

### **8.30.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

---

## **8.31 Device Management - Dynamic Device Assignment - A**

---

### **8.31.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

---

## **8.32 Device Management - Dynamic Device Assignment - B**

---

### **8.32.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

---

## **9 Data Link Layer**

---

## 9.1 Data Link Layer - MS/TP - Master Node

### 9.1.1 Base Requirements

Base requirements for all MS/TP master devices.

<b>135.1-2013 - 12.1.1.9.6 - Max Info Frame Check</b>		
	<b>Test Conditionality</b>	
	<b>Test Directives</b>	
	<b>Testing Hints</b>	A check differs from a test in that there are no specific test steps but rather conditions that the tester should check for while executing other tests. If the conditions are violated at any time while executing other tests, then the IUT shall fail this check.
<b>135.1-2013 - 12.1.3.2 - Verify <math>T_{\text{postdrive}}</math></b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Every MS/TP device shall claim 9600 baud. Test that the device operates at each baud rate that is claimed. Devices claiming Protocol_Revision 12 or higher where Addendum 135-2008ab is incorporated, shall claim 38400.
	<b>Testing Hints</b>	
<b>BTL - 12.1.3.3 - Verify <math>T_{\text{frame\_gap}}</math></b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Every MS/TP device shall claim 9600 baud. Test that the device operates at each baud rate that is claimed. Devices claiming Protocol_Revision 12 or higher where Addendum 135-2008ab is incorporated, shall claim 38400.
	<b>Testing Hints</b>	
<b>135.1-2013 - 12.1.3.4 - Verify <math>T_{\text{turnaround}}</math></b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Every MS/TP device shall claim 9600 baud. Test that the device operates at each baud rate that is claimed. Devices claiming Protocol_Revision 12 or higher where Addendum 135-2008ab is incorporated, shall claim 38400.
	<b>Testing Hints</b>	
<b>135.1-2013 - 12.1.3.5 - Verify <math>T_{\text{reply\_delay}}</math></b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 12.1.3.6 - Verify <math>T_{\text{usage\_delay}}</math> After a Token w/ Serial Analyzer</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 12.1.3.7 - Verify <math>T_{\text{usage\_delay}}</math> After Poll For Master w/ Serial Analyzer</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 12.1.3.8 - Verify <math>N_{\text{poll}}</math> w/ Serial Analyzer</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 12.1.3.9 - Verify <math>T_{\text{usage\_timeout}}</math> w/ Serial Analyzer</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

<b>135.1-2013 - 12.1.3.12 - Master Node Data Frame Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 12.1.3.13 - Poll for Master w/ Serial Analyzer</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 12.1.3.16 - MS/TP Network Startup Tests (IUT power on Variation)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 12.1.3.17 - MS/TP Network Startup Tests (IUTs wire connected)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 12.1.3.18 - MS/TP Network Startup Tests (IUTs wire disconnected)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 12.1.3.19 - MS/TP Network Startup Tests (Reference device joins the MS/TP network)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 2.2.X1 - Data Not For Us Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

## 9.1.2 Supports Writable Max\_Master Property

The IUT contains the Max\_Master property and it is writable.

<b>135.1-2013 - 12.1.3.10 - Max_Master Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

## 9.1.3 Supports Read Only Max\_Master Property

The IUT contains the Max\_Master property that is read-only.

<b>135.1-2013 - 12.1.3.10 - Max_Master Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

## 9.1.4 Contains Configurable Max\_Info\_Frames Property

The IUT contains a configurable Max\_Info\_Frames property.

<b>135.1-2013 - 12.1.3.11 - Max_Info_Frames Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	

	Testing Hints	
--	---------------	--

### 9.1.5 Contains Non-Configurable Max\_Info\_Frames Property

The IUT contains a non-configurable Max\_Info\_Frames property.

135.1-2013 - 12.1.3.11 - Max_Info_Frames Test		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

### 9.1.6 Is a BACnet Router

The IUT is or can be configured to act as a BACnet router.

135.1-2013 - 12.1.3.15 - Sole Master Test		
	Test Conditionality	Only one device on the MS/TP network
	Test Directives	
	Testing Hints	

### 9.1.7 Supports Extended MS/TP Frames (over 501 octets)

The IUT can transmit and receive messages with an NPDU > 501 octets

BTL - 12.1.3.X1 - Frame Type Based on Transmitted NPDU Size		
	Test Conditionality	Must be executed
	Test Directives	Execute the test such that the transmitted NPDU sizes are near the 501 octet boundary.
	Testing Hints	
BTL - 12.1.3.X2 - Executing COBS Encoded Frames		
	Test Conditionality	Must be executed
	Test Directives	
	Testing Hints	

## 9.2 Data Link Layer - MS/TP - Slave Node

### 9.2.1 Base Requirements

Base Requirements for all MS/TP slave devices.

<b>135.1-2013 - 12.3.3.2 - Verify <math>T_{\text{postdrive}}</math></b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Every MS/TP device shall claim 9600 baud. Test that the device operates at each baud rate that is claimed. Devices claiming Protocol_Revision 12 or higher where Addendum 135-2008ab is incorporated, shall claim 38400.
	<b>Testing Hints</b>	
<b>BTL - 12.1.3.3 - Verify <math>T_{\text{frame gap}}</math></b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Every MS/TP device shall claim 9600 baud. Test that the device operates at each baud rate that is claimed. Devices claiming Protocol_Revision 12 or higher where Addendum 135-2008ab is incorporated, shall claim 38400. Test both maintenance (Token and Poll For Master) as well as data frames.
	<b>Testing Hints</b>	
<b>135.1-2013 - 12.1.3.4 - Verify <math>T_{\text{turnaround}}</math></b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Every MS/TP device shall claim 9600 baud. Test that the device operates at each baud rate that is claimed. Devices claiming Protocol_Revision 12 or higher where Addendum 135-2008ab is incorporated, shall claim 38400.
	<b>Testing Hints</b>	
<b>135.1-2013 - 12.1.3.5 - Verify <math>T_{\text{reply delay}}</math></b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 12.1.3.14 - Slave Node Data Frame Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 2.2.X1 - Data Not For Us Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	



## 9.3 BACnet/IP - Annex J - non-BBMD Functionality

### 9.3.1 Base Requirements

Base requirements must be met by any IUT that can act, or can be made to act, as a BACnet/IP device in a non-BBMD mode.

<b>135.1-2013 - 14.1.1 - Write-Broadcast-Distribution-Table</b>		
	<b>Test Conditionality</b>	If the IUT is able to change into/out-of BBMD Mode via Write-BDT, then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 14.1.2 - Read-Broadcast-Distribution-Table</b>		
	<b>Test Conditionality</b>	If the IUT is able to change into/out-of BBMD Mode via Write-BDT, then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 14.1.3 - Register-Foreign-Device</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 14.1.4 - Delete-Foreign-Device-Entry</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 14.1.5 - Read-Foreign-Device-Table</b>		
	<b>Test Conditionality</b>	If the IUT is able to change into/out-of BBMD Mode via Write-BDT, then this test shall be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 14.1.6 - Distribute-Broadcast-To-Network</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 14.1.10 - Forwarded-NPDU (Two-hop Distribution)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 14.1.8 - Original-Broadcast-NPDU</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 14.1.9 - Original-Unicast-NPDU</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 14.1.X11 - Processing Forwarded-NPDU request initiated from different port</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 9.3.2 Is Able to Register as a Foreign Device

The IUT can register as a foreign device with a BBMD.

This functionality is required by all BACnet/IP devices that can act in a non-BBMD mode. Requiring this functionality is all ensures that the IUT will be able to participate in a BACnet/IP network even if there are no other BACnet/IP devices on the same IP subnet without the addition of a BBMD onto the subnet.

<b>135.1-2013 - 14.8 - Registering as a Foreign Device</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	The IUT is configured as a Foreign Device
	<b>Testing Hints</b>	
<b>BTL - 14.1.X12 - Processing Forwarded-NPDU request initiated from different port when registered as a Foreign Device into a BBMD</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 9.3.3 Is Able to Initiate Original-Broadcast-NPDU

The IUT can issue a broadcast on its own local subnet or through a BBMD.

<b>BTL - 14.9.1 - Distribute-Broadcast-To-Network</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

## 9.4 BACnet/IP - Annex J - BBMD

### 9.4.1 Base Requirements

The IUT acts, or can be made to act, as a BBMD device.

These base requirements must be met by any IUT that claims to support the Annex J BACnet/IP BBMD functionality.

<b>BTL - 14.2.1.2 - Execute Forwarded-NPDU (Two-hop Distribution)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 14.2.2.2 - Execute Original-Broadcast-NPDU (Two-hop Distribution)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 14.2.3 - Execute Original-Unicast-NPDU</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 14.1.X11 - Processing Forwarded-NPDU request initiated from different port</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 9.4.2 Supports a BDT with at Least Four Entries

The IUT acts as, or can be made to act as, a BBMD with a BDT that supports at least four entries in its BDT.

This functionality is required of all devices that claim support for an Annex J BACnet/IP BBMD.

<b>135.1-2013 - 14.3.1 - Execute Write-Broadcast-Distribution-Table (Table Growth)</b>		
	<b>Test Conditionality</b>	If the IUT does not support the Write-Broadcast-Distribution-Table message, this test may be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 14.3.2 - Execute Write-Broadcast-Distribution-Table (Table Shrinkage)</b>		
	<b>Test Conditionality</b>	If the IUT does not support the Write-Broadcast-Distribution-Table message, this test may be skipped.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 14.3.3 - Verify Broadcast Distribution Table Created from the Configuration Saved During the Previous Session</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 14.5.2.2 - Original-Broadcast-NPDU Which Shall Be Forwarded (Two-hop Distribution)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 9.4.3 Registration by a Foreign Device is Supported

While configured as a BBMD, the IUT supports, or can be made to support, registration by Foreign Devices and thus contains an FDT.

<b>135.1-2013 - 14.6.1 - Execute Read-Foreign-Device-Table</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 14.6.3.1 - Non-zero-Duration Foreign Device Table Timer Operations</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 14.6.5 - Execute Delete-Foreign-Device-Table-Entry Which Should Be Rejected</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 14.6.6 - Execute Delete-Foreign-Device-Table-Entry</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 14.7.1.2 - Broadcast Message from Directly Connected IP Subnet (Two-hop Distribution)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 14.7.2.2 - Broadcast Message Forwarded by a Peer BBMD (Two-hop Distribution)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 14.7.3.2 - Broadcast Message From a Foreign Device (Two-hop Distribution)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 9.4.4 Supports 2-Hop mode

The IUT is capable of being configured for two-hop distribution. In this mode a BBMD sends forwards original BACnet/IP broadcasts to each peer BBMD instead of sending directed broadcasts to each IP subnet.

Two-Hop mode is required functionality and many of the base requirements tests are applied against BBMDs configured to operate in two-hop mode. Due to the coverage of this functionality in the other test plan areas, no two-hop specific tests are provided here.

### 9.4.5 Supports 1-Hop Mode

The IUT is capable of being configured for one-hop distribution. In this mode a BBMD sends directed IP broadcasts to each IP subnet participating in the BACnet/IP network instead of to its peer BBMDs.

<b>BTL - 14.2.1.1 - Execute Forwarded-NPDU (One-hop Distribution)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 14.2.2.1 - Execute Original-Broadcast-NPDU (One-hop Distribution)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

<b>135.1-2013 - 14.5.2.1 - Original-Broadcast-NPDU Which Shall Be Forwarded (One-hop Distribution)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 14.7.1.1 - Broadcast Message from Directly Connected IP Subnet (One-hop Distribution)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 14.7.2.1 - Broadcast Message Forwarded by a Peer BBMD (One-hop Distribution)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 14.7.3.1 - Broadcast Message From a Foreign Device (One-hop Distribution)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 9.4.6 BBMD Supports Network Address Translation

The IUT is capable of operating behind a router providing Network Address Translation as described in addendum 135-2008o-1.

<b>BTL - 14.7.1.2 - Broadcast Message from Directly Connected IP Subnet (Two-hop Distribution)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Internet Routers and the IUT shall be configured for NAT.
	<b>Testing Hints</b>	
<b>BTL - 14.7.2.2 - Broadcast Message Forwarded by a Peer BBMD (Two-hop Distribution)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Internet Routers and the IUT shall be configured for NAT.
	<b>Testing Hints</b>	
<b>BTL - 14.7.3.2 - Broadcast Message From a Foreign Device (Two-hop Distribution)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Internet Routers and the IUT shall be configured for NAT.
	<b>Testing Hints</b>	

### 9.4.7 Is Able to Initiate Original-Broadcast-NPDU

While configured as a BBMD, the IUT can issue broadcasts on its own local subnet and through partner BBMDs.

<b>135.1-2013 - 14.9.3 - Original-Broadcast-NPDU</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

---

## 9.5 Data Link Layer - ZigBee

---

### 9.5.1 Base Requirements

These base requirements must be met by any IUT that claims to support for the ZigBee Data Link Layer.

BTL - 9.18.1.X1 - Reading Properties Based on Data Type		
	Test Conditionality	
	Test Directives	The IUT shall be configured to use the ZigBee Datalink. This test shall be executed using the Device Object and reading the Object_Name property.
	Testing Hints	

---

## 9.6 Data Link Layer - Ethernet

---

### 9.6.1 Base Requirements

These base requirements must be met by any IUT that claims to support for the Ethernet Data Link Layer.

BTL - 9.18.1.X1 - Reading Properties Based on Data Type		
	Test Conditionality	
	Test Directives	The IUT shall be configured to use the Ethernet Datalink. This test shall be executed using the Device Object and reading the Object_Name property.
	Testing Hints	

---

## 9.7 Data Link Layer - ARCNET

---

### 9.7.1 Base Requirements

These base requirements must be met by any IUT that claims to support for the ARCNET Data Link Layer.

BTL - 9.18.1.X1 - Reading Properties Based on Data Type		
	Test Conditionality	
	Test Directives	The IUT shall be configured to use the ARCNET Datalink. This test shall be executed using the Device Object and reading the Object_Name property.
	Testing Hints	



---

## 9.8 Data Link Layer - LonTalk

---

### 9.8.1 Base Requirements

These base requirements must be met by any IUT that claims to support for the LonTalk Data Link Layer.

BTL - 9.18.1.X1 - Reading Properties Based on Data Type		
	Test Conditionality	
	Test Directives	The IUT shall be configured to use the LonTalk Datalink. This test shall be executed using the Device Object and reading the Object_Name property.
	Testing Hints	

---

## **9.9 Data Link Layer - IPv6**

---

### **9.9.1 Base Requirements**

Contact BTL for interim tests for this data link.

---

## **9.10 Data Link Layer - Secure Connect**

---

### **9.10.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

---

## **10 Network Management**

---

## 10.1 Network Management - Routing

### 10.1.1 Base Requirements

There are no base requirements tests for this section.

### 10.1.2 Routes Packets between Physical BACnet LANs

A BACnet network router connects two or more BACnet networks at the network layer.

The tests are designed for testing routing devices that are connected to two BACnet LANs. The test device is required to send, receive, and “sniff” messages on both LANs at the same time. If a device is capable of routing between multiple combinations of BACnet LANs, then the tester shall use all possible LAN connections during testing, but it is not necessary to run all tests on all combinations of two LANs that the device supports.

If a routing device can be connected to three or more physical BACnet LANs at the same time, the tester shall construct a test network with at least three physical BACnet LANs directly connected to the routing device. Ideally the routing device should be connected to at least one network of each LAN type supported by the device, and more than one network of each LAN type if supported. See the “Test Conditionality” for specific requirements for certain tests.

These requirements must be met by any IUT that can act, or can be made to act, as a BACnet router.

<b>135.1-2013 - 10.2.1 - Startup</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Run this test a sufficient number of times to verify that the correct I-Am-Router-To-Network message is broadcasted on all LAN types supported by the routing device.
	<b>Testing Hints</b>	
<b>135.1-2013 - 10.2.2.1 - Forward I-Am-Router-To-Network</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Run this test a sufficient number of times to exercise all supported LAN types.
	<b>Testing Hints</b>	
<b>135.1-2013 - 10.2.2.2.1 - Execute Who-Is-Router-To-Network: No Specified Network Number</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Run this test a sufficient number of times to exercise all supported LAN types.
	<b>Testing Hints</b>	
<b>135.1-2013 - 10.2.2.2.2 - Execute Who-Is-Router-To-Network: A Known Remote Network Number is Specified</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 10.2.2.2.3 - Execute Who-Is-Router-To-Network: A Network Number is Specified and the Router Does Not Respond</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 10.2.2.2.4 - Execute Who-Is-Router-To-Network: An Unknown and Unreachable Network Number is Specified</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

<b>135.1-2013 - 10.2.2.2.5 - Execute Who-Is-Router-To-Network: An Unknown Network is Discovered</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 10.2.2.2.6 - Forwarding a Who-Is-Router-To-Network from a Remote Network</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	Use varying lengths for SADR (1-6 byte addresses).
<b>135.1-2013 - 10.2.2.3 - Forward I-Could-Be-Router-To-Network</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 10.2.2.4.1 - Forwarding Router-Busy-to-Network Information for Specific DNETs</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 10.2.2.4.2 - Forwarding Router-Busy-To-Network Information for all DNETs</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 10.2.2.4.3 - Receiving Messages for a Busy Router</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 10.2.2.4.4 - Router-Busy-To-Network: Timeout</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 10.2.2.5.1 - Execute Router-Available-To-Network: Restoring Specific DNETs</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 10.2.2.5.2 - Execute Router-Available-To-Network: Restoring All DNETs</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 10.2.2.7.1 - Unknown Network</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 10.2.2.7.2 - Unknown Network Layer Message Type</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 10.2.2.7.3 - Unknown Network Layer Message Type For Someone Else</b>		
	<b>Test Conditionality</b>	Must be executed on all devices that support BACnet revision 4 or higher.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 10.2.3.1 - Ignore Local Message Traffic</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Repeat for each LAN type supported by the device.
	<b>Testing Hints</b>	
<b>BTL - 10.2.3.2 - Route Message from a Local Device to a Local Device</b>		

	<b>Test Conditionality</b>	Must be executed
	<b>Test Directives</b>	Repeat for each LAN type supported by the device. (The test checks communication in both directions.) One of the tests 10.2.3.2 and 10.2.3.3 must be run by sending the message using a Local Broadcast destination, and the other test must be run by sending the message directly to the IUT's MAC address. This test may be varied by sending a BACnet-ComplexACK-PDU or a BACnet-SimpleACK-PDU instead of a BACnet-Confirmed-Request-PDU; check the value of the data expecting reply bit.
	<b>Testing Hints</b>	
<b>135.1-2013 - 10.2.3.3 - Route Message from a Local Device to a Router</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Repeat twice for each LAN type supported by the device, once with the originating device connected to the LAN and once with the destination router connected to the LAN. One of the tests 10.2.3.2 and 10.2.3.3 must be run by sending the message using a Local Broadcast destination, and the other test must be run by sending the message directly to the IUT's MAC address. Use varying lengths for DADR (1-7 byte addresses). This test should be varied by sending a BACnet-ComplexACK-PDU, a BACnet-Unconfirmed-Request-PDU or a BACnet-SimpleACK-PDU instead of a BACnet-Confirmed-Request-PDU; check the value of the data expecting reply bit.
	<b>Testing Hints</b>	
<b>135.1-2013 - 10.2.3.4 - Route Message from One Router to Another Router</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Repeat twice for each LAN type supported by the device, once with the originating router connected to the LAN and once with the destination router connected to the LAN. Use varying lengths for DADR (1-7 byte addresses) and SADR (1-6 byte addresses). This test should be varied by sending a BACnet-ComplexACK-PDU, a BACnet-Unconfirmed-Request-PDU or a BACnet-SimpleACK-PDU instead of a BACnet-Confirmed-Request-PDU; check the value of the data expecting reply bit.
	<b>Testing Hints</b>	
<b>135.1-2013 - 10.2.3.5 - Route Message from a Router to a Local Device</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Repeat twice for each LAN type supported by the device, once with the destination device connected to the LAN and once with the source router connected to the LAN. Use varying lengths for SADR (1-6 byte addresses). This test may be varied by sending a BACnet-ComplexACK-PDU, a BACnet-Unconfirmed-Request-PDU or a BACnet-SimpleACK-PDU instead of a BACnet-Confirmed-Request-PDU; check the value of the data expecting reply bit.
	<b>Testing Hints</b>	
<b>135.1-2013 - 10.2.3.6.1 - Failed Attempt to Locate (Downstream) Router</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	There should be a pause between the transmission of the Who-Is-Router-To-Network message and the transmission of the Reject-Message-To-Network message by the IUT. The BACnet standard does not say how long this pause should be, but one to ten seconds is probably a reasonable range.
<b>135.1-2013 - 10.2.3.6.2 - Successful Attempt to Locate (Downstream) Router</b>		

	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 10.2.4.1 - Ignore Local Broadcast Message Traffic</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Repeat for each LAN type supported by the device.
	<b>Testing Hints</b>	
<b>135.1-2013 - 10.2.4.2 - Global Broadcast from a Local Device</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Repeat this test for each LAN type supported by the device. In addition, repeat this test by sending the message directly to the IUT using the IUT's MAC address as well as using a Local Broadcast destination.
	<b>Testing Hints</b>	
<b>135.1-2013 - 10.2.4.3 - Global Broadcast from a Remote Device</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Repeat at least twice, with the originating router connected to different types of LANs. Use varying lengths for SADR (1-6 bytes). In addition, repeat this test by sending the message directly to the IUT using the IUT's MAC address as well as using a Local Broadcast destination.
	<b>Testing Hints</b>	
<b>135.1-2013 - 10.2.4.4 - Remote Broadcast from a Local Device to a Directly-Connected Network</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Repeat twice for each LAN type supported by the device, once with the source device connected to the LAN and once with the LAN as the destination network for the remote broadcast. One of the tests 10.2.4.4 and 10.2.4.5 must be run by sending the message directly to the IUT using the IUT's MAC address, and the other test must be run by sending the message using a Local Broadcast destination.
	<b>Testing Hints</b>	
<b>135.1-2013 - 10.2.4.5 - Remote Broadcast from a Local Device to a Non-Directly-Connected Network</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Repeat twice for each LAN type supported by the device, once with the source device connected to the LAN and once with destination router connected to the LAN. One of the tests 10.2.4.4 and 10.2.4.5 must be run by sending the message directly to the IUT using the IUT's MAC address, and the other test must be run by sending the message using a Local Broadcast destination.
	<b>Testing Hints</b>	
<b>135.1-2013 - 10.2.4.6 - Remote Broadcast from a Remote Device to a Directly-Connected Network</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Repeat twice for each LAN type supported by the device, once with the source router connected to the LAN and once with the LAN as the destination network for the remote broadcast. Also, vary the length of SADR (1-6 bytes).
	<b>Testing Hints</b>	
<b>135.1-2013 - 10.2.4.7 - Remote Broadcast from a Remote Device to a Remote Network</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Repeat twice for each LAN type supported by the device, once with the source router connected to the LAN and once with destination router connected to the LAN. Also, Vary the length of SADR (1-6 bytes).



	Testing Hints	
<b>135.1-2013 - 10.2.4.8 - Remote Broadcast that Should Be Ignored</b>		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
<b>135.1-2013 - 10.2.5 - Hop Count Protection</b>		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
<b>135.1-2013 - 10.2.6 - Network Layer Priority</b>		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
<b>BTL - 10.2.X1 - Initiates Network-Number-Is on Startup</b>		
	Test Conditionality	If the IUT supports Protocol_Revision 11 or greater, this test must be executed.
	Test Directives	
	Testing Hints	
<b>BTL - 10.2.X2 - Routers Execute What-Is-Network-Number</b>		
	Test Conditionality	If the IUT supports Protocol_Revision 11 or greater, this test must be executed.
	Test Directives	
	Testing Hints	

### 10.1.3 Routes Packets Between a Physical LAN and One or More Virtual LANs

The device can route BACnet packets between a physical BACnet LAN and one or more virtual BACnet LANs that contain one or more virtual BACnet devices. See H.1 and H.2 in the BACnet standard for a description of virtual BACnet LANs and virtual BACnet devices.

<b>135.1-2013 - 10.8.1 - Startup</b>		
	Test Conditionality	Must be executed.
	Test Directives	Run this test a sufficient number of times to verify that the correct I-Am-Router-To-Network message is broadcasted on all LAN types supported by the routing device.
	Testing Hints	
<b>135.1-2013 - 10.8.2.1.1 - Execute Who-Is-Router-To-Network: No Specified Network Number</b>		
	Test Conditionality	Must be executed.
	Test Directives	Run this test a sufficient number of times to exercise all supported LAN types.
	Testing Hints	
<b>135.1-2013 - 10.8.2.1.2 - Execute Who-Is-Router-To-Network: A Known Remote Network Number is Specified</b>		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
<b>135.1-2013 - 10.8.2.1.3 - Execute Who-Is-Router-To-Network: A Network Number is Specified and the Router Does Not Respond</b>		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	

<b>135.1-2013 - 10.8.2.1.4 - Execute Who-Is-Router-To-Network: An Unknown and Unreachable Network Number is Specified</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 10.8.2.2.1 - Unknown Network</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 10.8.2.2.2 - Unknown Network Layer Message Type</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 10.8.3.1 - Route Request Message from a Local Device to a Virtual Device and Route Response Message from the Virtual Device to the Local Device</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 10.8.3.2 - Route Request Message from a Virtual Device to a Local Device</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	This test should be run repeatedly in order to exercise all ways that the IUT can be configured or stimulated to send a unicast message to a device on a local network. Depending on the capabilities of the IUT this may involve sending a message from the target device to the IUT (unicast or broadcast), writing the network address of the target device to an object property in the IUT, writing the Device ID of the target device to an object property in the IUT, writing the Device Name of the target device to an object property in the IUT, or configuring the IUT using a proprietary method.
	<b>Testing Hints</b>	During the test, the TD shall answer any requests that the IUT generates while attempting to locate the route to the target device. The IUT may need to broadcast a Who-Is or Who-Has request in order to discover the network address of the target device if the network address is unknown.
<b>135.1-2013 - 10.8.3.3 - Route Request Message from a Remote Device to a Virtual Device and Route Response Message from the Virtual Device to the Remote Device</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Repeat using varying lengths for SADR (1-6 byte addresses).
	<b>Testing Hints</b>	
<b>135.1-2013 - 10.8.3.4 - Route Request Message from a Virtual Device to a Remote Device</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	Repeat using varying lengths for DADR (1-7 byte addresses).
	<b>Testing Hints</b>	
<b>BTL - 10.8.3.5.1 - Unknown Network</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 10.8.3.5.2 - Network Reachable Through the Same Port</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 10.8.4.1 - Broadcasts that Should Be Ignored</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	

	Testing Hints	
<b>135.1-2013 - 10.8.4.2 - Route Global Broadcast from a Local Device to Virtual Devices</b>		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
<b>135.1-2013 - 10.8.4.3 - Route Global Broadcast from a Remote Device to Virtual Devices</b>		
	Test Conditionality	Must be executed.
	Test Directives	Use varying the lengths of SADR (1-6 bytes).
	Testing Hints	
<b>135.1-2013 - 10.8.4.4 - Route Remote Broadcast from a Local Device to Virtual Devices</b>		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
<b>135.1-2013 - 10.8.4.5 - Route Remote Broadcast from a Remote Device to Virtual Devices</b>		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
<b>135.1-2013 - 10.8.4.6 - Route Global Broadcast Message from a Virtual Device</b>		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
<b>BTL - 10.8.4.7 - Route Remote Broadcast Message from a Virtual Device to a Local Network</b>		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
<b>135.1-2013 - 10.8.4.8 - Route Remote Broadcast Message from a Virtual Device to a Remote Network</b>		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
<b>135.1-2013 - 10.8.6 - Network Layer Priority</b>		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
<b>135.1-2013 - 10.8.7.1 - Who-Is Specifying Different Device ID</b>		
	Test Conditionality	Must be executed if the device supports DM-DDB-B.
	Test Directives	
	Testing Hints	
<b>135.1-2013 - 10.8.7.2 - Who-Has Specifying Different Device ID</b>		
	Test Conditionality	Must be executed if the device supports DM-DOB-B.
	Test Directives	
	Testing Hints	
<b>135.1-2013 - 10.8.7.3 - Read of Object Not Contained by Virtual Device</b>		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	
<b>BTL - 10.8.7.4 - Who-Is Specifying Unknown Device IDs</b>		
	Test Conditionality	Must be executed if the device supports DM-DDB-B.
	Test Directives	
	Testing Hints	
<b>BTL - 10.8.7.5 - Who-Has Specifying Unknown Device IDs</b>		
	Test Conditionality	Must be executed if the device supports DM-DOB-B.
	Test Directives	
	Testing Hints	

<b>BTL - 10.2.X1 - Initiates Network-Number-Is on Startup</b>		
	<b>Test Conditionality</b>	If the IUT supports Protocol_Revision 11 or greater, this test must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 10.2.X2 - Routers Execute What-Is-Network-Number</b>		
	<b>Test Conditionality</b>	If the IUT supports Protocol_Revision 11 or greater, this test must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 10.8.3.6.X1 - Silently Drop Messages to a Virtual Device that is Offline</b>		
	<b>Test Conditionality</b>	Must be executed if any virtual device can become offline, for a time.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

## 10.2 Network Management - Router Configuration - B

The tests are designed for testing routing devices which connect two or more BACnet networks at the network layer.

### 10.2.1 Base Requirements

There are no base requirements tests for this section, since all testing except as in the following specific sections, is specified in Network Management - Routing.

### 10.2.2 Supports Routing

The IUT supports the requirements for BACnet Routers.

Verify Checklist		
	Test Conditionality	Must be executed.
	Test Directives	Verify that the IUT claims support for section 10.1 Network Management - Routing.
	Testing Hints	

### 10.2.3 Supports DS-WP-B

The IUT supports DS-WP-B, to configure writable properties of any Network Port objects which it contains.

Verify Checklist		
	Test Conditionality	Must be executed if Protocol_Revision is 17 or greater.
	Test Directives	Verify that the IUT claims support for DS-WP-B.
	Testing Hints	

## 10.3 Network Management - Connection Establishment - A

### 10.3.1 Base Requirements

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

<b>135.1-2013 - 10.5.3.1 - Establish-Connection-To-Network</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 10.5.3.2 - Disconnect-Connection-To-Network</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

---

## **10.4 Network Management - Connection Establishment - B**

---

### **10.4.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

---

## **10.5 Network Management - Router Configuration - A**

---

### **10.5.1 Base Requirements**

Contact BTL for interim tests for this BIBB.



---

## **10.6 Network Management - BBMD Configuration - A**

---

### **10.6.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

---

## **10.7 Network Management - BBMD Configuration - B**

---

### **10.7.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

## 10.8 Network Management - Foreign Device Registration - A

### 10.8.1 Base Requirements

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

<b>BTL - 14.8.1 - Register-Foreign-Device Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 14.8.X1 - Register-Foreign-Device Enable and Disable Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 14.8.X2 - Recurring Register-Foreign-Device Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 14.1.6 - Distribute-Broadcast-To-Network</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>135.1-2013 - 14.1.9 - Original-Unicast-NPDU</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	
<b>BTL - 14.1.10 - Forwarded-NPDU (Two-hop Distribution)</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 10.8.2 Supports Configurable BBMD Address

The IUT supports a configurable BBMD Address to which it sends Register-Foreign-Device NPDU.

<b>BTL - 14.8.X3 - BBMD Address Configuration Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 10.8.3 Supports a Mode Where it Transmits a Broadcast at Startup

The IUT transmits a Broadcast at Startup, which can be observed preceded by the sending of Register-Foreign-Device NPDU, when configured as a Foreign Device.

<b>BTL - 14.8.X4 - Transmits a Broadcast at Startup preceded by Register-Foreign-Device</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	
	<b>Testing Hints</b>	

### 10.8.4 Supports Configurable Time-to-Live

The IUT supports a configurable Time-to-Live which it uses in the Register-Foreign-Device NPDU it sends.

<b>BTL - 14.8.X5 - Time-to-Live Configuration Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	

## BTL Test Plan

	<b>Testing Hints</b>	
--	----------------------	--

---

## **10.9 Network Management-Secure Connect Hub-B**

---

### **10.9.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

---

## **10.10 Network Management-Secure Connect Direct Connect-A**

---

### **10.10.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

---

## **10.11 Network Management-Secure Connect Direct Connect-B**

---

### **10.11.1 Base Requirements**

Contact BTL for interim tests for this BIBB.

---

## **11 Gateway**

---



---

## **11.1 Gateway - Virtual Network - B**

---

### **11.1.1 Base Requirements**

Contact BTL for Interim tests for this BIBB.

## 11.2 Gateway - Embedded Objects - B

### 11.2.1 Base Requirements

Base requirements must be met by any IUT that claims GW-EO-B.

<b>BTL - 9.18.1.X8 - ReadProperty Service when Non-BACnet Device Offline</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	The test shall be conducted upon an object which is representing information arriving through a Gateway.
	<b>Testing Hints</b>	
<b>BTL - 9.20.1.X9 - ReadPropertyMultiple Service when Non-BACnet Device Offline</b>		
	<b>Test Conditionality</b>	If IUT does not support ReadPropertyMultiple service then this test shall be skipped.
	<b>Test Directives</b>	The test shall be conducted upon an object which is representing information arriving through a Gateway.
	<b>Testing Hints</b>	
<b>BTL - 9.21.1.X10 - ReadRange Service when Non-BACnet Device Offline</b>		
	<b>Test Conditionality</b>	If IUT does not support ReadRange service then this test shall be skipped. If IUT supports the ReadRange service but does not support a list property that maps onto data from a non-BACnet device, this test shall be skipped.
	<b>Test Directives</b>	The test shall be conducted upon an object which is representing information arriving through a Gateway.
	<b>Testing Hints</b>	

### 11.2.2 Supports Command Prioritization

Gateways are required to implement Priority\_Array properties correctly with all 16 entries

<b>135.1-2013 - 7.3.1.2 - Relinquish Default Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	The test shall be conducted upon an object which is representing information arriving through a Gateway. If no object can be made to meet the configuration requirements, this test shall be skipped.
	<b>Testing Hints</b>	
<b>135.1-2013 - 7.3.1.3 - Command Prioritization Test</b>		
	<b>Test Conditionality</b>	Must be executed.
	<b>Test Directives</b>	The test shall be conducted upon an object which is representing information arriving through a Gateway.
	<b>Testing Hints</b>	

---

## **12 Network Security BIBBs**

---

---

## **12.1 Network Security - Secure Device**

---

### **12.1.1 Base Requirements**

Contact BTL for Interim tests for this BIBB.

---

## **12.2 Network Security - Encrypted Device**

---

### **12.2.1 Base Requirements**

Contact BTL for Interim tests for this BIBB.

---

## **12.3 Network Security - Multi-Application Device**

---

### **12.3.1 Base Requirements**

Contact BTL for Interim tests for this BIBB.

---

## **12.4 Network Security - Device Master Key - A**

---

### **12.4.1 Base Requirements**

Contact BTL for Interim tests for this BIBB.

---

## **12.5 Network Security - Device Master Key - B**

---

### **12.5.1 Base Requirements**

Contact BTL for Interim tests for this BIBB.



---

## **12.6 Network Security - Key Server**

---

### **12.6.1 Base Requirements**

Contact BTL for Interim tests for this BIBB.

---

## **12.7 Network Security - Temporary Key Server**

---

### **12.7.1 Base Requirements**

Contact BTL for Interim tests for this BIBB.

---

## **12.8 Network Security - Secure Router**

---

### **12.8.1 Base Requirements**

Contact BTL for Interim tests for this BIBB.

---

## **12.9 Network Security - Security Proxy**

---

### **12.9.1 Base Requirements**

Contact BTL for Interim tests for this BIBB.

---

## **13 Audit Reporting BIBBs**

---

---

## **13.1 Audit Reporting - Logging - A**

---

### **13.1.1 Base Requirements**

Contact BTL for Interim tests for this BIBB.

---

## **13.2 Audit Reporting - Reporter - B**

---

### **13.2.1 Base Requirements**

Contact BTL for Interim tests for this BIBB.

---

## **13.3 Audit Reporting - Reporter - Simple - B**

---

### **13.3.1 Base Requirements**

Contact BTL for Interim tests for this BIBB.



---

## **13.4 Audit Reporting - Forwarder - B**

---

### **13.4.1 Base Requirements**

Contact BTL for Interim tests for this BIBB.

---

## **13.5 Audit Reporting - View - A**

---

### **13.5.1 Base Requirements**

Contact BTL for Interim tests for this BIBB.

---

## **13.6 Audit Reporting - Advanced View and Modify - A**

---

### **13.6.1 Base Requirements**

Contact BTL for Interim tests for this BIBB.

# BTL Test Plan

Version	Date	Author	Change
0.11	5-Aug-2004	Carl Neilson	<ul style="list-style-type: none"> <li>• Updates based on Nashville meeting comments on Round 3 updates.</li> </ul>
0.12	24-Aug-2004	Carl Neilson	<ul style="list-style-type: none"> <li>• Removed 9.24.4.X1, 9.24.4.X2. Now reference 9.24.1.5 and 9.24.1.6 in 135.1.</li> <li>• Changed to reference BTL - 10.2.4.4 instead of 135.1 - 10.2.4.4</li> <li>• Changed to reference BTL - 10.2.4.6 instead of 135.1 - 10.2.4.6</li> <li>• Added conditional language to 10.2.6 for test 14.X2.1</li> <li>• Corrected 4.9.1 reference to BTL - 7.3.1.X.</li> </ul>
0.13	26-Oct-2004	Roland Laird	<ul style="list-style-type: none"> <li>• Updated BACnet/IP sections based on changed tests</li> </ul>
0.14	27-Oct-2004	Carl Neilson	<ul style="list-style-type: none"> <li>• Changed referenced to 7.3.1.11, 9.1.1.1, 9.1.1.4, 9.1.2.1, 9.1.2.5 to be BTL Specified tests</li> </ul>
0.15	29-Oct-2004	Carl Neilson	<ul style="list-style-type: none"> <li>• Removed highlighted edits marked for deletion.</li> </ul>
0.16	18-Nov-2004	Carl Neilson	<ul style="list-style-type: none"> <li>• Changed referenced from BTL - 9.24.1.X1 to 135.1 - 9.24.1.4</li> </ul>
0.17	19-Nov-2004	Carl Neilson	<ul style="list-style-type: none"> <li>• Added DCC protocol revision 4 entries</li> </ul>
0.18	1-Dec-2004	Carl Neilson	<ul style="list-style-type: none"> <li>• Minor typo</li> </ul>
0.19	20-Dec-2004	Carl Neilson	<ul style="list-style-type: none"> <li>• Updated section 10.3.2 Routing from Tp-Routing-08</li> <li>• Added BTL-9.20.2.1 to RPM-B Base Requirements</li> <li>• Added BTL-7.3.2.9.8, 7.3.2.9.9, 7.3.2.15.5, 7.3.2.18.6, 7.3.2.19.5, 7.3.2.22.9, 7.3.2.13.1, 7.3.2.13.2 to WP-B - Contains Resizable Arrays.</li> <li>• Added 9.23.1.X7, 9.23.1.X8 to WPM-B Base Requirements</li> <li>• Added Alarming protocol revision 4 entries</li> </ul>
0.20	13-Jan-2005	Carl Neilson	<ul style="list-style-type: none"> <li>• Added clarifying text to the test hint for 9.2.7.6 about how filters are applied.</li> </ul>
0.21	18-Feb-2005	Carl Neilson	<ul style="list-style-type: none"> <li>• Separated DDB-A options for unicast vs broadcast use of WhoIs without range parameters</li> <li>• Made the scheduling of the INTEGER datatype optional in SCHED-E-B.</li> </ul>
0.22	15-Jun-2005	Carl Neilson, Roland Laird	<ul style="list-style-type: none"> <li>• Fixed typo in 3.3</li> <li>• Added into 9.20.1.7,-9.20.19, allowance for skipping the test when APDU and segmentation restrictions make the test impossible.</li> <li>• Added 4.8.10 Can Cancel Subscriptions.</li> <li>• Added 4.8.12 Can Request Infinite Subscriptions.</li> <li>• Clarified why some datatypes are not explicitly listed in DS-RP-B and DS-RPM-B</li> <li>• Added testing hint to 135.1 - 9.7.2.6</li> <li>• Added two-hop section to match the checklist</li> <li>• Added revision 4 scheduling entries</li> </ul>
0.23	16-Jun-2005	Carl Neilson	<ul style="list-style-type: none"> <li>• Added test plan entries for modifiable lists</li> </ul>
0.24	20-Jul-2005	Carl Neilson	<ul style="list-style-type: none"> <li>• Added WhoHas tests 9.32.1.X1, 9.32.1.X2</li> </ul>
0.25	05-Oct-2005	Carl Neilson	<ul style="list-style-type: none"> <li>• Added MS/TP restart tests 2.2.14..2.2.17</li> <li>• Added RPM/RP fallback tests 8.20.5.1, 8.20.5.2</li> </ul>

# BTL Test Plan

0.26	24-Oct-2005	Carl Neilson	<ul style="list-style-type: none"> <li>Added Notification Class object &amp; moved Recipient_List entries from AE-N-I-B &amp; T-ATR-B</li> <li>Removed BUFFER_READY specific sections from T-ATR-B and added the tests into the intrinsic and algorithmic sections of T-ATR-B</li> <li>Added test 7.3.1.X3 Array Sizing Test to WP-B and WPM-B</li> <li>Added 13.X2.1 APDU Retry and Timeout in basic functionality</li> </ul>
0.27	27-Oct-2005	Carl Neilson	<ul style="list-style-type: none"> <li>Removed leftover Recipient_List entry from AE-N-I-B</li> <li>A few minor typos fixed</li> </ul>
4.0.0	13-Sep-2006	Carl Neilson	<ul style="list-style-type: none"> <li>Changed revision numbering</li> </ul>
4.0.1	04-Apr-2007	Carl Neilson	<ul style="list-style-type: none"> <li>Round 4 changes (excluding SCHED)</li> </ul>
4.0.2	08-Jun-2007	Lori Tribble	<ul style="list-style-type: none"> <li>Applied changes required for CRR-0008, CRR-0015, CRR-0017, CRR-0020, CRR-0021</li> <li>Reconciled some of the differences between the checklist and this document.</li> </ul>
4.0.3	30-Jul-07	Lori Tribble	<ul style="list-style-type: none"> <li>Changed test references from BTL to 135.1a.</li> </ul>
4.0.4	25-Aug-07	Lori Tribble	<ul style="list-style-type: none"> <li>Highlighted changes</li> </ul>
4.0.5	25-Oct-07	Lori Tribble	<ul style="list-style-type: none"> <li>Removed Highlights</li> <li>Changed references from 135.1 to 135.1-2003 or 135.1-2003a where appropriate.</li> <li>Removed (includes NM-RC-B) from Routing title.</li> <li>Updated the BTL Mark.</li> <li>Updated test conditionality for 9.18.1.X1.</li> </ul>
4.0.6	2-Nov-2007	Lori Tribble	<ul style="list-style-type: none"> <li>Removed 'Supports writable Number_Of_States Property' from Multi-State Objects because tests included in WP-B.</li> <li>Added tests from WP-B 'resizable arrays' to WPM-B section 'resizable arrays'.</li> </ul>
4.0.7	11-Jan-2008	Lori Tribble	<ul style="list-style-type: none"> <li>Added DM-TS-B and DM-UTC-B to SCH-I-B</li> <li>Added Virtual Routing</li> <li>Added DM-LM-A and DM-LM-B</li> <li>Removed test 14.X2.3 from the BBMD test section 10.2.6 Is Able to Register as a Foreign Device per BTL-CRR-0041</li> <li>Changed 5.2.3 Supports Notification Class Object to reference Verify Checklist instead of a test that is already run under the Notification Class Object set of tests.</li> <li>Updated Test Conditionality for section 10.2.2 tests due to optional functionality requirements.</li> <li>Added place holder and partial B&amp;R-A section.</li> </ul>
4.0.8	8-Feb-2008	Lori Tribble	<ul style="list-style-type: none"> <li>Updated DM-BR-A</li> <li>Added test BTL-9.1.1.X3 in AE-ACK-B for new revision 5 requirements.</li> <li>Removed 8.3.6 and 8.3.7 from DOB-A - BTL-CRR-0042</li> <li></li> </ul>
4.0.9	22-Feb-2008	Lori Tribble	<ul style="list-style-type: none"> <li>Updated List Manipulation-B test references to include 2003. Changed configuration to As per 135.1-2003.</li> </ul>

# BTL Test Plan

			<ul style="list-style-type: none"> <li>• Fixed references to DeviceCommunicationControl</li> <li>• Added place holders for SCH-AVM-A, SCH-VM-A, SCH-WS-A.</li> <li>• Fixed CHANGE_OF_STATE test reference from 135.1 to BTL.</li> </ul>
4.0.10	13-Mar-2008	Lori Tribble	<ul style="list-style-type: none"> <li>• Fixed test references for 5.1.13 and 5.1.14.</li> <li>• Updated page header format</li> </ul>
4.0.11	16-Apr-2008	Lori Tribble	<ul style="list-style-type: none"> <li>• Updated test references from 135.1 to BTL</li> <li>• Updated B&amp;R-A tests from 13.X2 to 13.X5.</li> </ul>
4.0.12	12-May-2008	Lori Tribble	<ul style="list-style-type: none"> <li>• Corrected references for tests in DM-ADM-A and DM-ANM-A.</li> </ul>
4.0.13	21-May-2008	Lori Tribble	<ul style="list-style-type: none"> <li>• Changed test references from 135.1 to BTL per BTL-CRR-0017</li> <li>• Modified AE-N-I-B requirements per BTL-CRR-0054.</li> <li>• Added EE requirement line item to AE-N-E-B.</li> <li>• Added Calendar object requirements.</li> <li>• Added Event Enrollment object requirements.</li> <li>• Added Schedule Object requirements.</li> <li>• Added Trend Log Object requirements.</li> <li>• Added SCH-WS-I-B and SCH-R-B</li> <li>• Modified test references from 135.1 to BTL per WS-038-4.</li> <li>• Updated table of contents.</li> </ul>
4.0.14	28-Jun-2008	Lori Tribble	<ul style="list-style-type: none"> <li>• Accepted all previous modifications</li> <li>• Added the BTL requirements sections and moved appropriate tests. Corresponds with Checklist 4.0.14.</li> <li>• Modified AE-AS-A requirements to not reference AE BIBBs that are being deleted by Addendum L.</li> <li>• Added comments where newly approved Addendum M affects tests and test plan.</li> <li>• Changed -2003 references to -2007. Updated test numbers where appropriate but left old numbers for reference.</li> </ul>
4.0.15	9-Sep-2008	Lori Tribble	<ul style="list-style-type: none"> <li>• Applied BTL-CRR-0061 Effective_Period</li> <li>• Applied changed to DM-ADM-A</li> <li>• Applied BTL-CRR-0059 DCC_TimeDuration</li> <li>• Changed references for 7.3.2.8.1, 7.3.2.8.2, 7.3.2.8.3 from 135.1 to BTL.</li> <li>• Changed references for 7.3.2.21.3.1 and 7.3.2.21.3.2 from 135.1 to BTL.</li> <li>• Changed reference for 7.3.2.23.2 from 135.1 to BTL.</li> <li>• Changed all BTL references which were changed by 135.1-2009.</li> <li>• Changed references for 7.3.2.23.3.1 - 9 from 135.1 to BTL.</li> <li>• Changed references for 7.3.2.23.4 - 8 from 135.1 to BTL.</li> <li>• Added Test Conditionality to tests in SCH-I-B and SCH-E-B.</li> </ul>

# BTL Test Plan

			<ul style="list-style-type: none"> <li>• Add changes required by current version of SCH-A document v9. More changes will be required to complete this section.</li> </ul>
4.0.16	17-Sep-2008	Lori Tribble	<ul style="list-style-type: none"> <li>• Accepted all changes made previously.</li> <li>• Corrected some format issues.</li> </ul>
4.0.17	7-Oct-2008	Lori Tribble	<ul style="list-style-type: none"> <li>• Corrected test reference for COV from 24 hours to 8 hours.</li> </ul>
4.0.18	17-Oct-2008	Lori Tribble	<ul style="list-style-type: none"> <li>• Fixed header and footer on first page.</li> <li>• Removed 9.23.1.7 WPM maximum properties and 9.20.1.12 RPM max properties</li> <li>• Updated SCH-WS-A, SCH-AVM-A, and SCH-VM-A per latest approved documents.</li> <li>• Added test 7.2.2.1 Read-Only Property Test and 7.2.2.X2 Non-documented Property Test to EPICS section of test plan.</li> <li>• Removed the 'input-tracking' and 'output-tracking' tests from AI, AO, BI, BO base requirements.</li> </ul>
5.0.1	17-Oct-2008	Lori Tribble	<ul style="list-style-type: none"> <li>• Accepted all previous changes</li> <li>• Modified test called out for WPM, Contains Writable List Properties.</li> </ul>
5.0.2	24-Feb-2009	Lori Tribble	<ul style="list-style-type: none"> <li>• Rearranged Analog Output section to match order in Checklist.</li> <li>• Rearranged RPM-A section to match order in Checklist.</li> <li>• Removed DS-COVU-A and DS-COVU-B sections as they are not in the Checklist.</li> <li>• Changed title of BTL-8.22.X4 to capitalize 'whole'.</li> <li>• Changed test requirements for Will Accept Infinite COV Subscriptions (4.10.17)</li> <li>• Changed reference from 135.1-2009-7.2.2 to BTL-7.2.2.</li> <li>• Removed Segmentation sections since tests are not available yet.</li> <li>• Added resizable support under object sections and removed from WP-B and WPM-B sections.</li> <li>• Added note to OCD-A Support File Objects to clarify when it should be selected.</li> <li>• Corrected Test Conditionality for Multi-State objects Support State_Text section.</li> <li>• Corrected typo in AE-ACK-B Testing Hints for test BTL-9.1.2.1.</li> <li>• Added unknown object tests to RP-B and RPM-B sections.</li> <li>• Added Ethernet section to Test Plan.</li> </ul>
5.0.3	20-March-2009	Frank Schubert	<ul style="list-style-type: none"> <li>• Formatted test references</li> <li>• Corrected Verify Checklist statements (removed "The tester shall")</li> <li>• Changed BTL - 8.5.X2 - Extended Algorithm Tests to BTL - 8.5.X3</li> <li>• Changed revision =4 to revision &gt;=4</li> <li>• Changed "revision x later" to "revision x higher"</li> <li>• Replaced multi-whitespaces to single whitespaces</li> </ul>

# BTL Test Plan

			<ul style="list-style-type: none"> <li>• Changed Hints or Conditionality = “none” to blank fields</li> <li>• Changed “Must always be executed” to “Must be executed”</li> <li>• Added field “Test Directives”</li> <li>• Changed some tests marked in red color to black color</li> <li>• Moved “Testing Hints” of Verify Checklist tests to “Test Directives”</li> <li>• Changed BACnet Basic Functionality to match IUT checklist</li> <li>• Changed “SCH-“ to “SCHED-“</li> <li>• Changed “T-VM-“ to “T-VMT-“</li> <li>• Renamed chapter 9 Data-Link-Layer</li> <li>• Moved BACnet/IP to chapter 9</li> <li>• Moved Ethernet to Chapter 9</li> <li>• Moved Routing to Chapter 10</li> <li>• Changed all “Results” to “Notes &amp; Results”</li> <li>• Formatted all tables to have width of 6”</li> <li>• Changed tests specified for DM-LM-B Supports writable proprietary list properties of primitive datatypes.</li> </ul>
5.0.4	27-Mar-2009	Lori Tribble	<ul style="list-style-type: none"> <li>• Added new sections in SCHED-I-B and SCHED-E-B that reflect changes to Addendum L.</li> <li>• Added ‘Manual’ to all ‘Verify Checklist’ entries.</li> <li>• Renumbered 9.23.1.X7 to 9.23.2.6 as defined in 135.1-2009.</li> <li>• Renumbered 9.23.1.X8 to 9.23.2.7 as defined in 135.1-2009.</li> <li>• Changed reference from 135.1-2009-7.3.2.24.8 to BTL-7.3.2.24.8 due to BTL-CRR-0070 changes.</li> <li>• Added Database_Revision tests to Device Object section.</li> <li>• Rearranged Object Creation and Deletion -A section to remove requirement of create by Object_Type.</li> </ul>
5.0.5	6-Apr-2009	Lori Tribble	<ul style="list-style-type: none"> <li>• Accepted previous changes</li> <li>• Changed reference for Stop_When_Full test. It has been fixed in the BTL Specified Tests document. (BTL-7.3.2.24.6.1)</li> </ul>
5.0.6	9-Apr-2009	Lori Tribble	<ul style="list-style-type: none"> <li>• Changed reference to ‘Reading a Single, Unsupported Property from a Single Object’ from 135.1-2009 to BTL. This test was corrected by BTL-CRR-0039.</li> <li>• Modified references for Create and Delete tests per document proposed in BTL-WG and approved 4/9/2009.</li> </ul>
5.0.7	8-Jun-2009	Lori Tribble	<ul style="list-style-type: none"> <li>• Removed Effective_Period tests from SCHED-I-B base requirements. They are now part of a separate section in the SCHED-I-B group. Same was done for SCHED-WS-I-B.</li> <li>• Updated Binary Input and Output object requirements per BTL-CRR-KV01.</li> </ul>



# BTL Test Plan

			<ul style="list-style-type: none"> <li>Updated requirements to DM-DCC-B per BTL-CRR-KV02.</li> <li>Modified references to test 7.3.1.1 to reference the new BTL version. Changed the Configuration to indicate the object required during the test execution.</li> </ul>
5.0.8	22-Jun-2008	Lori Tribble	<ul style="list-style-type: none"> <li>Removed duplicate test (BTL-9.23.1.X2) in section 8.4.6.</li> <li>Removed note in configuration for tests 7.3.1.1 per discussion in BTL-WG meeting on 6/18/2009</li> <li>Corrected Polarity tests description paragraph.</li> <li>Removed the 'No Specific Test' under Base Requirements for section 8.14.</li> <li>Corrected test name for test number BTL - 9.21.1.4</li> </ul>
5.0.9	7-Jul-2009	Lori Tribble	<ul style="list-style-type: none"> <li>Accepted all changes per BTL-WG meeting on June 18, 2009.</li> <li>Corrected formatting errors.</li> <li>Removed AE-ANS-A and AE-ANS-B sections</li> <li>Deleted Macros from document.</li> </ul>
5.0.final	7-Jul-2009	Lori Tribble	<ul style="list-style-type: none"> <li>Renamed to final</li> </ul>
6.0.6	5-Jan-2011	Duffy O'Craven	<ul style="list-style-type: none"> <li>Integrated changes for 135-2004d-5_v3.doc</li> <li>Integrated changes for BTL-CRR-KV06-ReadingDataTypes.pdf into DS-RP-B/RPM-B</li> <li>Added Contains Octet String to DS-WP-B/WPM-B</li> <li>Integrated Trend Log Multiple, Event Log, and Structured View changes</li> <li>Added Processes UNSIGNED_RANGE notifications to AE-N-A</li> <li>Integrated deprecation of static router binding</li> <li>Renamed DM-OCD-A sections from Supports ... to Can Create and Delete ...</li> <li>Added ZigBee Data Link Layer</li> <li>Added Supports Clock-aligned Logging</li> </ul>
6.0.7	19-Jan-2011	Duffy O'Craven	<ul style="list-style-type: none"> <li>Initial capitalized Create and Delete, and Clock-align Logging section names</li> <li>Renamed Initates to Issues in BACnet/IP sections</li> <li>Revised Conditionality of 7.3.2.24.6.1/7.3.2.24.6.2 Stop_When_Full tests based upon BTL-CRR-0120 Stop_When_Full.doc</li> </ul>
6.0.8	19-Jan-2011	Duffy O'Craven	<ul style="list-style-type: none"> <li>Revised AE-N-I-B and AE-N-E-B based upon 135-2004m-5 r4 Add requirements to Alarm and Event BIBBs.doc</li> <li>Added ReinitializeDevice with invalid 'Reinitialized State of Device', based upon 135-2004m-8 r2 Clarify DeviceCommunicationControl and ReinitializeDevice interactions.doc</li> </ul>
6.0.9	19-Jan-2011	Duffy O'Craven	<ul style="list-style-type: none"> <li>Revised all test references to 135.1 - 2009 if the test is present there.</li> <li>Corrected test references to 135.1 - 2009 as appropriate, based upon errors brought to light in BTL-CRR-0110_7.3.2.23.9_reference.doc, BTL-CRR-0125_9.1.2.6.doc, BTL-CRR-</li> </ul>

# BTL Test Plan

			<p>0129_7.3.2.23.9_conditionality.doc, and BTL-CRR-0147_9.2.1.2_reference.doc</p> <ul style="list-style-type: none"> <li>Added conditionality on tests 9.8.3 and 9.8.6 according to BTL-CRR-0133_9.8.3_conditionality.doc</li> </ul>
6.0.10	26-Jan-2011	Duffy O'Craven	<ul style="list-style-type: none"> <li>Added tests 9.1.1.X4 and 9.1.1.X5 to ACK-B, based upon BTL Specified Tests-Add135-2004m-4-ReAckAlarms-3.doc</li> <li>Added test 9.21.1.X5 Reading Items with Negative Count and MOREITEMS</li> <li>Derived tests from 135.1-2009 in DCC-A and RD-A, adding proper password treatment based upon BTL-CRR-0078 - DeviceCommunicationControl_Password.doc</li> <li>Added conditionality to 9.24.1.X3 and 9.24.1.X4 - Disable of Service Initiation Restored by Time Duration and by DeviceCommunicationControl, based upon BTL-CRR-KV04-DisableServiceInitiation.doc</li> <li>Revised tests 9.24.2.1, 9.24.2.2, 9.27.1.1 and 9.27.1.3, and added tests 9.24.2.X3, 9.27.2.X3 and 9.27.2.X4 in DCC-B and RD-B, based upon 135-2004m-8 r2 Clarify DeviceCommunicationControl and ReinitializeDevice interactions.doc</li> <li>Eliminated section: Supports AE-AVN-A, in AE-AVM-A since Addendum 135-2008l specifies no such requirement.</li> </ul>
9.0.11	09-Feb-2011	Duffy O'Craven	<ul style="list-style-type: none"> <li>Combined Structured View optional sections into one</li> <li>DM-OCD-A combined first two lineitems</li> <li>BACnet/IP unicast is not optional so is moved into Base Requirements</li> <li>BACnet/IP broadcast section renamed to Is able to initiate Original-Broadcast-NPDU</li> </ul>
9.0.12	10-Feb-2011	Duffy O'Craven	<ul style="list-style-type: none"> <li>Corrected test references in COV for Life Safety Objects to BTL - 9.2.1.X2 according to BTL-CRR-0111_9.2.1.X2_reference.doc</li> <li>Moved AE algorithms' requirement that the test be repeated from Test Conditionality to Test Directives</li> <li>If Event Enrollment objects are supported, ensure AE-N-I-B is tested on Event Enrollment objects.</li> </ul>
9.0.13	23-Feb-2011	Duffy O'Craven	<ul style="list-style-type: none"> <li>Corrected Test Conditionality in "Supports writable and resizable Subordinate_List and contains a Subordinate_Annotations property" to Must be executed.</li> <li>Added missing section "Is able to Present and Modify Weekly_Schedule of INTEGER (Signed) Type"</li> <li>Reordered "Supports Writable Polarity Property" in Binary Input objects to match Checklist order</li> <li>Reordered "Initiates Who-Has Service Request with Object Name Identifier Parameter with Range</li> </ul>

# BTL Test Plan

			<p>Parameters” and Initiates Who-Has Service Request with Object Identifier Name Parameter and No Range Parameters” to match Checklist order</p> <ul style="list-style-type: none"> <li>• Reordered “Supports Non-Password Protected Backup” to match Checklist order</li> <li>• Test Conditionality added for “Priority_For_Writing is writable” in consequence of BTL-CRR-0091_Must_Priority_For_Writing_be_Writable.doc</li> </ul>
9.0.14	09-Mar-2011	Duffy O’Craven	<ul style="list-style-type: none"> <li>• Incorporated “Enables/Disables BBMD Mode via Write-BDT” changes</li> <li>• In List Manipulation-B moved Negative Tests to “Base Requirements” and Positive Tests to “All writable list properties in the IUT support list manipulation”</li> <li>• Reordered “Is able to Schedule Octet String Values” in Schedule-External-B to match Checklist order</li> <li>• In consequence of BTL-CRR-0179_9600_baud.doc, testing now ensures that every MS/TP device claims 9600 baud, and that the device operates at each baud rate that is claimed. Devices claiming Protocol_Revision 12 or higher where Addendum 135-2008ab is incorporated, shall claim 38400.</li> <li>• In consequence of BTL-CRR-0180_P_C_C.doc, testing now ensures that every deleted property is detected, if present.</li> </ul>
9.0.15	06-Apr-2011	Duffy O’Craven	<ul style="list-style-type: none"> <li>• Corrected test references for test 9.10.2.1 The Monitored Object Does Not Support COV Notification from 135.1 - 2007 to BTL Specified tests as specified in BTL-CRR-0104_correcting_9.10.2.1.doc</li> <li>• Revised Conditionality of 9.10.1.X2 Ensuring Subscription Lifetimes Are Not Effected By Time Changes test based upon BTL-CRR-0183_9.10.1.X2.doc</li> <li>• In consequence of re-consideration of BTL-CRR-0091_Must_Priority_For_Writing_be_Writable.doc reverted the Test Conditionality added for “Priority_For_Writing is writable”.</li> <li>• Removed duplicate section 6.7.5 “Can Be Made to Contain a Schedule That Schedules NULL Values” and incorrect test BTL - 7.3.2.23.X1.1 in Schedule - Readonly - B. It appears later as 6.7.23, correctly with test BTL - 7.3.2.23.X1.2</li> <li>• The preamble for WPM-B “Contains Writable NULL properties” is clarified, because this section and tests are explicitly for non-commandable properties, in consequence of BTL-CRR-0156_4.5.6_4.5.7.doc</li> <li>• In consequence of BTL-CRR-0182_9.10.1.2, added Directive that tests 9.10.1.1 and 9.10.1.2 shall be</li> </ul>

# BTL Test Plan

			executed against objects which will accept the subscription.
9.0.16	26-Apr-2011	Duffy O'Craven	<ul style="list-style-type: none"> <li>• Clarified by renaming the sections in AE-AS-A, that the filters apply only to GetEnrollmentSummary requests</li> <li>• Removed section "Supports AE-AVN-A" in AE-AVM-A</li> <li>• Renamed "Can write NULL property values" in DS-WP-A, and the prior section, in consequence of BTL-CRR-0156_4.5.6_4.5.7.doc</li> <li>• Renamed "Can write NULL property values" and the prior section in DS-WPM-A, in consequence of BTL-CRR-0156_4.5.6_4.5.7.doc</li> <li>• Added "Supports SCHED-I-B" and "Supports DS-WP-A" to Scheduling - External - B, because they are required</li> <li>• Renamed "Is able to present and modify Weekly_Schedule of INTEGER (signed) type" to match the Checklist and other section's names which includes the parenthetical</li> <li>• Removed "Supports DS-RP-B" and "Supports DS-WP-B" from Scheduling - Weekly Schedule - Internal - B, because those are implicitly required elsewhere</li> <li>• Removed "Supports DS-RP-B" from Scheduling - Readonly- B, because that is explicitly required elsewhere</li> <li>• Reordered "Is able to schedule Octet String values" to match Checklist order</li> <li>• Corrected test references for test 9.33.2.3 General Inquiry, Directed to a Remote Device from 135.1 - 2007 to BTL Specified tests as specified in BTL-CRR-0101_9.33.2.3.doc</li> <li>• Corrected test 9.32.2.1 Object ID Version, Global Broadcast from a Remote Network from the version in 135.1 - 2009 and changed to reference it in BTL Specified Tests, as specified in BTL-CRR-KV05-ObjectID VersionRemotebroadcast.doc.</li> <li>• Corrected test 9.10.1.7 Finite Lifetime Subscriptions from the version in 135.1 - 2009 and changed to reference it in BTL Specified Tests, as specified in BTL-CRR-0184_9.10.1.7.doc and BTL-CRR-0194_ACK_in_9.10.1.1_and_9.10.1.7.doc and 0200_9.10.1.7.doc</li> </ul>
9.0.17	24-May-2011	Duffy O'Craven	<ul style="list-style-type: none"> <li>• Renamed section name to "Supports Correct Interaction of Weekly and Exception Schedules" in SCHED-I-B, in consequence of BTL-CRR-0208_No-Weekly_Schedule.</li> <li>• Reverted alterations of Test Conditionality and Test Directives and Test Hints in "Can write NULL property values" and the prior section in DS-WP-A and DS-WPM-A, in consequence of simplified BTL-CRR-0156_4.5.6_4.5.7.doc</li> </ul>

# BTL Test Plan

9.0.18	30-May-2011	Duffy O'Craven	<ul style="list-style-type: none"> <li>Removed Testing Hints from tests 10.X.3 in consequence of BTL-CRR-0106_10.X.3_TestingHints.doc</li> <li>Added sections Supports TimeSynchronization_Recipients and Supports UTC_TimeSynchronization_Recipients in consequence of BTL-CRR-0123_UTC_or_Time_Recipients_present.doc</li> <li>Reverted section name to "Supports concurrent Weekly and Exception Schedules" in SCHED-I-B, as the response to BTL-CRR-0208_No-Weekly_Schedule was: no change.</li> <li>Corrected test references for tests 9.30.1.1, 9.30.1.2, 9.31.1.1 and 9.31.1.2 from 135.1 - 2007 to BTL Specified tests as specified in BTL-CRR-0113_9.31.1.1_diverge_dissimilar_tests.doc, and added a Test Directive that the parameter used in test 9.31.1.1 - UTCTimeSynchronization Local Broadcast that shall cause the result to cross midnight.</li> <li>Renamed the sections involving datatype NULL in DS-WP-A and DS-WPM-A, to mention commandable and non-commandable properties, in consequence of BTL-CRR-0156_4.5.6_4.5.7.doc</li> </ul>
9.0.19	09-Jun-2011	Duffy O'Craven	<ul style="list-style-type: none"> <li>Renamed the sections involving datatype NULL in DS-WP-B and DS-WPM-B, to mention they are for non-commandable properties, extending upon BTL-CRR-0156_4.5.6_4.5.7.doc</li> </ul>
9.0.20	12-Jun-2011	Duffy O'Craven	<ul style="list-style-type: none"> <li>Added Test Directives and Hint to 10.X.5 to ensure that the packet actually reaches the IUT, and that the test uses an address which resembles the actual address of IUT, in consequence of BTL-CRR-0138_10.X.5_same_DADR.doc and BTL-CRR-0139_10.X.5_ensure_reaches_IUT.doc</li> <li>Changed references on tests 7.3.2.21.3.X and 8.22.X1, as the versions in 135.1-2009g-6 and 135.1-2009i-8 incorporated them.</li> <li>Changed references on tests 7.1 and 13.1.12.1 to BTL Specified Tests versions, in consequence of BTL-CRR-0177_server_in_Abort-PDU.doc</li> <li>Changed reference on test 7.3.2.24.12 to BTL Specified Tests version, in consequence of BTL-CRR-0165_7.3.2.24.12.doc</li> <li>Changed reference on test 9.21.1.4 as 135.1-2009g-16 replaced it, in consequence of BTL-CRR-0201_9.21.1.4.doc</li> <li>Changed reference on test 14.7.3.1 as 135.1-2009e replaced it, in consequence of BTL-CRR-0209_14.7.3.1.doc</li> </ul>
9.0.21	14-Jun-2011	Duffy O'Craven	<ul style="list-style-type: none"> <li>Aligned Test Plan section names with Checklist lineitem names</li> </ul>
9.0.22	28-Sep-2011	Duffy O'Craven	<ul style="list-style-type: none"> <li>Marked tests 7.3.2.24.X8 and 7.3.2.24.X9 as <i>This test is not yet defined and shall be skipped.</i>, since those tests are incomplete and were never ratified.</li> </ul>

# BTL Test Plan

			<ul style="list-style-type: none"> <li>• Incorporated tests 13.2.1 through 13.2.7, the Time Master tests.</li> </ul>
9.0.23	30-Sep-2011	Duffy O'Craven	<ul style="list-style-type: none"> <li>• Changed reference for 9.1.10.X2 to 9.10.X in 135.1-2009d-1</li> <li>• Changed reference for 7.3.1.X1 to 7.3.2.10.1 in 135.1-2009d-2</li> <li>• Changed referenced for tests in sections 14.1, 14.2, 14.3, 14.5, 14.6, 14.7, and 14.X to be 135.1-2009e-1</li> <li>• Changed referenced for 7.3.1.11, 9.1.1.1, 9.1.1.4, 9.1.2.1, and 9.1.2.5 to be 135.1-2009f-1</li> <li>• Changed referenced for 7.3.2.10.X1, 7.3.2.10.X2, and 7.3.2.10.X3 to be 135.1-2009f-2</li> <li>• Changed reference for 8.16.3, 8.16.4, 9.16.1.1, 9.16.1.3, 9.16.1.4, 9.16.2.2, 9.16.2.3, 9.16.2.4, and 9.16.2.5 to be 135.1-2009f-3</li> <li>• Changed reference for 9.17.1.1 to be 135.1-2009f-4</li> <li>• Changed reference from BTL - 9.24.1.X2 to 135.1-2009g-8 - 9.24.1.X1 and for BTL - 9.24.1.X3 to 135.1-2009g-8 - 9.24.1.X2</li> <li>• Changed references for 7.3.1.1 to be 135.1-2013</li> <li>• Changed reference from BTL - 10.X.5 to 135.1-2009g-10 - 10.X.1 and for BTL - 10.X.6 to 135.1-2009g-10 - 10.X.2 and for BTL - 10.X.7 to 135.1-2009g-10 - 10.X.3</li> <li>• Changed reference from BTL - 10.X.1 to 135.1-2009g-10 - 10.Y.1 and for BTL - 10.X.2 to 135.1-2009g-10 - 10.Y.2 and for BTL - 10.X.3 to 135.1-2009g-10 - 10.Y.3</li> <li>• Changed reference for test 7.3.2.21.3.4 as the version in 135.1-2009g-11 takes precedence but that only portrayed the intended revision with a context-diff, so the entirety of the revised test is rendered here.</li> <li>• Changed references for test 8.1 as the version in 135.1-2009g-13 takes precedence.</li> <li>• Changed references from BTL - 8.18.X3 to be 135.1-2009g-14 - 8.18.3</li> <li>• Changed references from BTL - 9.4.X1 to 135.1-2009g-15 9.4.5 and for BTL - 9.4.X2 to 135.1-2009g-15 9.4.6 and for BTL - 9.5.X1 to 135.1-2009g-15 9.5.1 and for BTL - 9.5.X2 to 135.1-2009g-15 9.5.2</li> </ul>
9.0.24	10-Oct-2011	Duffy O'Craven	<ul style="list-style-type: none"> <li>• Fixed cut&amp;paste typo in section 4.5.11 for Multi-state Value where Multi-state Output was mentioned instead.</li> <li>• Changed references on test 9.14.2.2 to BTL Specified Tests version, in consequence of BTL-CRR-0117_9.14.2.2_First_Failed_Element.doc</li> <li>• Changed references for test 7.3.2.24.9 as the version in 135.1-2009g-16 takes precedence.</li> <li>• Changed references for tests 7.3.2.23.3.1, 7.3.2.23.X2.3.1, 7.3.2.23.X2.3.2, 7.3.2.23.X2.3.3, 7.3.2.23.X2.3.4, 7.3.2.23.X2.3.5, and</li> </ul>

# BTL Test Plan

			<p>7.3.2.23.X2.3.6 as the versions in 135.1-2009g-17 take precedence. Note that the test numbers used in 135.1-2009g-17 each specify X rather than the X2 used in Test Plan-5.0.final and BTL Specified Test-5.0.final.</p> <ul style="list-style-type: none"> <li>• Changed references for test 13.X3 to be 135.1-2009g-19 13.X1 and for test 13.X4 to be 135.1-2009g-19 13.X2 as those take precedence.</li> <li>• Changed references for test 8.3.X1 to be 135.1-2009g-20 8.3.X and for test 9.3.X8 to be 135.1-2009g-20 9.3.1 as those take precedence.</li> <li>• Added the conditionality from 135.1-2009h-1 to tests 9.8.3 and 9.8.6</li> <li>• Changed references for 8.18.1 and 8.18.2 from BTL to 135.1-2009i-4</li> <li>• Changed references for tests 7.3.2.23.X1.1, 7.3.2.23.X1.2, 7.3.2.23.X1.3, and 7.3.2.23.X1.4 as the versions in 135.1-2009g-21 take precedence. Note that the test numbers used in 135.1-2009g-21 each specify Y rather than the X1 used in Test Plan-5.0.final and BTL Specified Test-5.0.final.</li> <li>• Changed references for tests 8.8.1 and 8.8.2 from BTL to 135.1-2009i-5</li> <li>• Changed references for test 8.20.Y1.X1 to 135.1-2009i-6 - 8.20.Y1.1 and for test 8.20.Y1.X2 to 135.1-2009i-6 - 8.20.Y1.2</li> <li>• Changed references for tests 7.23.9.3, 8.23.1, 8.23.2, 8.23.3, and 8.23.4 from 135.1-2009 to 135.1-2009i-7 in consequence of an added Notes to tester:</li> <li>• Changed references for 8.22.1 and 8.22.2 from BTL to 135.1-2009i-7 as that addendum ratified the Notes to Tester: addition that had caused these revised tests to supercede the 135.1 - 2003 - 8.22.1 and 135.1 - 2003 - 8.22.2 versions.</li> <li>• Changed reference for test 8.22.X1 to be 135.1-2009i-8 - 8.22.X1</li> </ul>
9.0.25	24-Nov-2011	Duffy O'Craven	<ul style="list-style-type: none"> <li>• Prefixed referenced tests 9.30.1.1, 9.30.1.2, 9.31.1.1, 9.31.1.2 with TimeSynchronization and UTCTimeSynchronizartion, matching their new names in BTL Specified Tests.</li> <li>• Removed the duplicate reference to test 13.2.1 (the one with wrong conditionality) that had been introduced by a paste error, in the section on Automatic Time Synchronization - A.</li> <li>• Changed reference for test BTL - 7.3.1.X2 to 135.1-2009i-15 - 7.3.2.11.X</li> <li>• Changed references for tests 7.3.2.23.X2.3.1 through 7.3.23.X2.3.6 to 135.1-2009g-17 - 7.3.2.23.X.3.1 through 135.1-2009g-17 - 7.3.2.23.X.3.6 as the versions in 135.1-2009g-17 take precedence. Note that the test numbers used in 135.1-2009g-17 each specify X rather than the X2 used in Test Plan-5.0.final and BTL Specified Test-5.0.final.</li> </ul>

# BTL Test Plan

			<ul style="list-style-type: none"> <li>• Changed reference for test BTL - 8.34.X1 to 135.1-2009i-12.</li> <li>• Changed references for tests 9.1.1.X4 and 9.1.1.X5 from BTL to 135.1-2009i-17.</li> <li>• Changed reference for 9.1.2.6 from 135.1-2009 to BTL.</li> <li>• Changed reference for 9.7.1.1 from 135.1-2009 to BTL.</li> <li>• Changed reference for 9.16.1.4 back to BTL for it contains a more accurate restriction of "...any unique object identifier of a type that is creatable and an instance number that is creatable".</li> <li>• Changed reference for tests 9.21.1.1, 9.21.1.2, 9.21.1.3, 9.21.1.4.X1, 9.21.1.6.X1, 9.21.1.6.X2, 9.21.1.X1, 9.21.1.X2, and 9.21.2.X4 because the versions in 135.1-2009i-14 take precedence. Note that BTL - 9.21.1.X3 is preserved for it contains a more accurate list: "Qualifying tests are: 9.21.1.1, 9.21.1.2, 9.21.1.3, 9.21.1.4, 9.21.1.4.X1, 9.21.1.X1 or 9.21.1.X2."</li> <li>• Changed reference for test 9.23.2.6 as the version in 135.1-2009i-10 takes precedence.</li> <li>• Changed reference for test 9.20.2.1 as the version in 135.1-2009i-11 takes precedence.</li> <li>• Corrected references for tests 9.24.2.1 and 9.24.2.2 from BTL to 135.1-2009</li> </ul>
9.0.26	28-Nov-2011	Duffy O'Craven	<ul style="list-style-type: none"> <li>• Added a list of the additional addenda incorporated into this test plan to achieve Protocol_Revision 9.</li> <li>• Added a list of the additional addenda incorporated into this test plan to adopt pan-Protocol BIBB amendments.</li> <li>• Corrected the references to test 7.1 from 135.1-2009 to BTL in RPM as was already done in RP.</li> <li>• Replaced all endashes with hyphens, and made the spacing around them consistent.</li> <li>• Corrected the references to tests 135.1-2009g-10 - 10.X.1, 10.X.2, and 10.X.3 from the higher numbers 10.X.5, 10.X.6, and 10.X.7 which were used in BTL Specified Tests-5.0.fnal</li> <li>• Removed trailing period from name of test 7.3.2.23.X1.3 - Externally Written Datatypes Test, non-NULL values.</li> <li>• Changed Test Conditionality to "<i>This test is not yet defined and shall be skipped.</i>" For tests 7.3.2.24.X8 and 7.3.2.24.X9 in T - VMMT - I - B as well as in T - VMT - I - B.</li> <li>• Corrected the name (removing the underscores and replacing the second one with a space) for the 13.2.1 - TimeSynchronization Recipients Test</li> <li>• Corrected references for test 9.32.2.1 from BTL to 135.1-2009, since the wholesale changes suggested in BTL-CR-0112 are not yet completed.</li> </ul>



# BTL Test Plan

9.0.27	28-Nov-2011	Duffy O'Craven	<ul style="list-style-type: none"> <li>• Fixed test 8.34.X1 reference (Unicast is not part of test name) and a typo in the number as 135,1 should be 135.1</li> <li>• Changed references to all four tests of 8.21.X from BTL to 135.1-2009i-19.</li> <li>• Changed references for tests BTL - 9.1.1.X4 - Successful Alarm Re-Acknowledgment of Confirmed Event Notifications and BTL - 9.1.1.X5 Unconfirmed Event Notifications to 135.1-2009i-17 - 9.1.1.X1 and 9.1.1.X2 respectively.</li> <li>• Changed references for test 9.16.1.1 from BTL to 135.1-2009f-3</li> <li>• Changed the occurrences of BTL Specified Tests in Test Configuration for tests 9.16.1.2 and 9.17.1.1 to BTL Specified Tests.</li> <li>• Changed references for test 9.17.1.1 from BTL to 135.1-2009f-4</li> <li>• Note that BTL - 9.21.1.X3 is preserved for it contains a more accurate list: "Qualifying tests are: 9.21.1.1, 9.21.1.2, 9.21.1.3, 9.21.1.4, 9.21.1.4.X1, 9.21.1.X1 or 9.21.1.X2."</li> <li>• Change Test Configuration for test 9.21.1.X4 to <b>ASHRAE 135.1-2009</b>. 8.22.19 Supports Object Creation and Deletion of the Life Safety Zone Object</li> <li>• Changed reference to 9.22.2.4 from BTL to 135.1-2009i-9.</li> <li>• Changed references from BTL - 9.23.1.X1 to 135.1-2009i-9 - 9.23.1.X</li> <li>• Change reference for test 9.24.2.2 from 135.1-2009 to 135.1-2009i-16.</li> <li>• Changed reference for test 9.24.2.X3 from BTL to 135.1-2009i-16.</li> <li>• Change references to 9.27.1.1, 9.27.1.3, 9.27.2.X3, 9.27.2.X4 to 135.1-2009i-16.</li> <li>• Change reference to 9.33.2.3 from 135.1-2009 to 135.1-2009i-18.</li> <li>• Changed reference of test BTL - 9.2.1.X8 - Change of Value Notification from Proprietary Objects to BTL - 9.3.X9</li> <li>• Did not change the reference from 135.1-2009 to non-existent test BTL - 9.32.2.1, since the wholesale changes suggested in BTL-CR-0112 are not yet completed.</li> </ul>
9.0.28	29-Nov-2011	Duffy O'Craven	<ul style="list-style-type: none"> <li>• Changed reference missing a number for the test to BTL - 9.17.1.1 in section "Supports Object Creation and Deletion of the Life Safety Zone Object".</li> <li>• Changed reference for the test 7.3.2.23.3.9 in section "Scheduling - Internal - B" from 135.1-2009 to BTL, as it already is in "Scheduling - Readonly - B".</li> <li>• Added leading space character after hyphen before references to 9.21.1.1 -Reading All Items in the List</li> </ul>

# BTL Test Plan

			<ul style="list-style-type: none"> <li>Capitalized Test consistently in 7.3.2.24.1 - Enable Test</li> </ul>
9.0.29	29-Nov-2011	Duffy O'Craven	<ul style="list-style-type: none"> <li>Changed another pair of references in section "Supports Initiation of GetEventInformation Service" for tests 8.8.1 and 8.8.2 from BTL to 135.1-2009i-5</li> <li>Replaced less-than-symbol in the 13.2.1 - TimeSynchronization Recipients Test, Protocol_Revision &lt; 7 with a left-brace, so that it cut&amp;pastes more conventionally.</li> <li>Reverted italics to normal in "Supports Time_Synchronization_Recipients" and "Supports UTC Time Synchronization Recipients"</li> </ul>
9.0.30	30-Nov-2011	Duffy O'Craven	<ul style="list-style-type: none"> <li>Changed references for tests 9.33.1.1, 9.33.1.2, 9.33.1.4, 9.33.1.5, 9.33.1.6, 9.33.2.1, 9.33.2.2, and 9.33.2.3 from 135.1-2009 to 135.1-2009i-18 which was based upon "BTL Specified Tests-Add135-2004q-1-UnicastIAm-2.doc"</li> <li>Fixed the section boundaries between 5.16.1 and 5.16.2 so that they don't display two-across in Web Layout view.</li> </ul>
9.0.final	01-Dec-2011	Duffy O'Craven	<ul style="list-style-type: none"> <li>Updated from 9.0.30 to 9.0.final, accepting all change tracking</li> </ul>
12.0.1	24-Jul-2012	Lori Tribble	<ul style="list-style-type: none"> <li>Applied Errata 9.0 7/19/2012</li> <li>Applied Addendum 9.0-a</li> <li>Applied Addendum 9.0-b</li> <li>Applied Addendum 9.0-c</li> <li>Applied Errata 12.0 7/23/2012</li> </ul>
12.0.2	02-Aug-2012	Lori Tribble	<ul style="list-style-type: none"> <li>Applied Errata-BTL Test Package 9.0 plus addenda 8/02/2012 (includes above Errata which was not published)</li> </ul>
12.0.final	02-Aug-2012	Lori Tribble	<ul style="list-style-type: none"> <li>Accepted changes and changed name to final</li> </ul>
12.1.1	27-Sept-2013	Lori Tribble	<ul style="list-style-type: none"> <li>Applied Addendum 12.0b</li> </ul>
12.1.2	27-Sept-2013	Lori Tribble	<ul style="list-style-type: none"> <li>Applied Addendum 12.0c</li> </ul>
12.1.3	30-Sept-2013	Lori Tribble	<ul style="list-style-type: none"> <li>Applied Addendum 12.0d</li> </ul>
12.1.4	30-Sept-2013	Lori Tribble	<ul style="list-style-type: none"> <li>Applied Addendum 12.0e</li> </ul>
12.1.5	1-Oct-2013	Lori Tribble	<ul style="list-style-type: none"> <li>Applied Addendum 12.0f</li> </ul>
12.1.6	1-Oct-2013	Lori Tribble	<ul style="list-style-type: none"> <li>Applied Addendum 12.0g</li> </ul>
12.1.7	1-Oct-2013	Lori Tribble	<ul style="list-style-type: none"> <li>Applied Errata</li> </ul>
14.0.a	1-Nov-2014	Lori Tribble	<ul style="list-style-type: none"> <li>Applied Addendum 12.1a</li> </ul>
14.0.b	1-Nov-2014	Lori Tribble	<ul style="list-style-type: none"> <li>Applied Addendum 12.1b</li> </ul>
14.0.c	1-Nov-2014	Lori Tribble	<ul style="list-style-type: none"> <li>Applied Addendum 12.1c</li> </ul>
14.0.d	1-Nov-2014	Lori Tribble	<ul style="list-style-type: none"> <li>Applied Addendum 12.1d</li> </ul>
14.0.e	1-Nov-2014	Lori Tribble	<ul style="list-style-type: none"> <li>Applied Addendum 12.1e</li> </ul>
14.0.plus errata	3-Nov-2014	Lori Tribble	<ul style="list-style-type: none"> <li>Cleaned up remaining BTL vs 135.1 references.</li> </ul>
14.0.final	19-Nov-2014	Duffy O'Craven	<ul style="list-style-type: none"> <li>Accepted changes, deleted comments, and changed name to final</li> </ul>
15.0.05	24-Aug-2017	Lori Tribble	<ul style="list-style-type: none"> <li>Applied Addenda 14.0b-j plus errata</li> </ul>
15.0.07	16-Sep-2017	Lori Tribble	<ul style="list-style-type: none"> <li>Applied additional errata.</li> </ul>
15.0.08	25-Sep-2017	Lori Tribble	<ul style="list-style-type: none"> <li>Removed of 'supports Reliability Evaluation'</li> </ul>
15.0.09	25-Sep-2017	Lori Tribble	<ul style="list-style-type: none"> <li>Changed test references in Supports User Initiated Abort Backup to the correct test. Changed test</li> </ul>

# BTL Test Plan

			reference in Supports User Initiated Abort Restore to the correct test.
15.0.10	28-Sep-2017	Lori Tribble	• Applied two errata.
15.0.final	28-Sep-2017	Lori Tribble	• Accepted all changes.
15.1.1	29-Mar-2018	Lori Tribble	• Applied addenda a, b, c and d and errata
15.1.2	29-Mar-2018	Lori Tribble	• More errata changes
15.1.3	5-Apr-2018	Lori Tribble	• Additional changes to complete Addenda b changes.
15.1.4	6-Apr-2018	Lori Tribble	• Accepted all changes.
15.1.5	17-Apr-2018	Lori Tribble	• Applied additional Errata
15.1.6	1-May-2018	Lori Tribble	• Applied additional Errata
15.1.final	1-June-2018	Lori Tribble	• Renamed to final
15.2.1		Lori Tribble	• Applied errata and e
15.2.2		Lori Tribble	• Applied addenda f
15.2.3	13-Oct-2018	Lori Tribble	• Applied addenda g and added missing sections to match the checklist added sections.
15.2.4	11-Nov-2018	Lori Tribble	• Applied errata
15.2.final	11-Nov-2018	Lori Tribble	• Accepted all changes and changed revision number
15.3.3	16-Nov-2018	Lori Tribble	• Reformatted the tables to remove 'Method', 'Configuration', and 'Notes & Results'.
16.0.1	19-Aug-2019	Lori Tribble	• Updated to docx. Rebuilt TOC. Changed version. Errata changes: Titles to match checklist items, removed italic, changed long dashes to short dashes.
16.0.2	19-Aug-2019	Lori Tribble	• Applied Addenda h.
16.0.3	26-Aug-2019	Lori Tribble	• Applied Addenda i
16.0.4	27-Aug-2019	Lori Tribble	• Applied Addenda j
16.0.5	28-Aug-2019	Lori Tribble	• Applied Addenda k
16.0.6	29-Aug-2019	Lori Tribble	• Applied Addenda l
16.0.7	29-Aug-2019	Lori Tribble	• Applied Addenda m
16.0.8	29-Aug-2019	Lori Tribble	• Applied Addenda n
16.0.9	30-Aug-2019	Lori Tribble	• Applied Addenda o
16.0.10	30-Aug-2019	Lori Tribble	• Applied Addenda p
16.0.11	30-Aug-2019	Lori Tribble	• Applied Addenda r
16.0.12	30-Aug-2019	Lori Tribble	• Applied Addenda s
16.0.13	30-Aug-2019	Lori Tribble	• Applied Errata
16.0.14	25-Sep-2019	Lori Tribble	• Accepted all changes. Formatting fixes.
16.0.final	25-Sep-2019	Lori Tribble	• Renamed to Final
16.0.final.v2	5-Nov-2019	Lori Tribble	• Added Ipv6 data link place holder.
16.1	10-Dec-2019	Lori Tribble	• Applied Errata, Added PR21 and PR22 items, renamed to 16.1.
16.1.Final	10-Jan-2020	Emily Hayes	• Renamed to Final