BACnet TESTING LABORATORIES

TESTING GUIDE

Revised October 26, 2005

(Equivalent to version 0.22)

BACnet is a registered trademark of ASHRAE. ASHRAE does not endorse, approve or test products for compliance with ASHRAE standards. Compliance of listed products to the requirements of ASHRAE Standard 135 is the responsibility of the BACnet Manufacturers Association (BMA). BTL is a registered trademark of the BMA.

Table Of Contents

1 Introduction 1
1.1 Test Planner 1
1.2 Tester 1
1.3 The Test Process 1
2 Test Planning 3
2.1 Step 1 – Complete Functionality Checklist 3
2.1.1 Purpose 3
2.1.2 Requirements 3
2.1.3 Completing the Functionality Checklist 4
2.2 Step 2 – Plan the IUT Configuration 5
2.2.1 Purpose 5
2.2.2 Requirements 6
2.2.3 Designing The IUT Configuration 6
2.3 Step 3 – Select the Tests 7
2.3.1 Purpose 7
2.3.2 Requirements 7
2.3.3 Selecting Tests 7
2.4 Step 4 – Check Test Conditionality 8
2.4.1 Purpose 8
2.4.2 Requirements 8
2.4.3 Evaluating Conditionality 8
2.5 Step 5 – Select Test Parameters 9
2.5.1 Purpose 9
2.5.2 Requirements 9
2.5.3 Selecting Test Parameters 10
2.6 Step 6 – Create the EPICS 13
2.6.1 Purpose 13
2.6.2 Requirements 13
2.6.3 Generating an EPICS 13
3 Test Execution 14
3.1 Step 7 – Apply the Tests 14
3.1.1 Purpose 14
3.1.2 Requirements 14
3.1.3 Executing Tests 14
4 Test Parameter Selection Hints 18
5 BTL Lab Testing 19
5.1 Processing a Testing Application 19
5.2 Pre-Testing Support 19
5.3 The First Testing Pass 19
5.4 Regression Testing 20
5.5 Product Retesting Requirements 20
5.6 Severe Problem Resolution Procedure 21

1 Introduction

The ***BTL Testing Guide*** provides a detailed walk through of the application of the BTL test suite on an IUT. This guide is intended to help in the preparation of a BACnet device for official BTL testing and/or to improve the in-house testing of BACnet devices.

Before proceeding with this guide's testing procedures, the ***BTL Getting Started Guide*** should be read. To get the most out of the ***BTL Testing Guide*** a general understanding of the documents in the BTL Documentation Package should be acquired.

The ***BTL Testing Guide*** is written for the Test Planner and Tester, which are defined below.

1.1 Test Planner

The Test Planner is the individual that will identify the testable IUT functionality, prepare the IUT's Configuration, determine the tests to apply, and select appropriate parameters for those tests. The Test Planner uses the BTL test package to create a custom test package for the IUT that will be used by the Tester. The Test Planner requires a thorough understanding of the functionality of the IUT and how to make the IUT exhibit all claimed functionality. All information the Test Planner needs in order to accomplish the task is contained in following sections of this guide or within the documents referenced by this guide.

1.2 Tester

The Tester is the individual that applies the test plan developed by the Test Planner to the IUT. The Tester will use VTS and other test tools to execute the tests. The Tester requires a thorough understanding of the configuration of the IUT and the test tools being used. The Tester will need to be able to understand the concepts used by tests. All information the Tester needs in order to accomplish this task is contained within the last step, or within the documents referenced by the last step, of this guide.

1.3 The Test Process

The testing process is divided into two phases: test planning and test execution. The test planning phase produces the IUT's custom test plan. The test execution phase applies the IUT's custom test plan to the IUT.

Figure 1: The test planning and execution process.

2 Test Planning

The test planning phase produces the IUT's custom test plan.

2.1 Step 1 – Complete Functionality Checklist

2.1.1 Purpose

The goal is to complete the ***BTL Functionality Checklist*** for the IUT. The ***BTL Functionality Checklist*** contains a table of BACnet functionality that may or may not apply to the IUT. The goal of this step to understand the table and select the functionality within the table that is implemented by the IUT. The completed table identifies the IUT's capabilities that will, in turn, be mapped onto tests.

The ***BTL Functionality Checklist*** document is filled out in this step by marking the items implemented by the IUT with an X.

Figure 2: Complete Functionality Checklist step.

2.1.2 Requirements

The ***BTL Functionality Checklist*** document will be filled out during this step and is the main working document used during this step. The ***BTL Test Plan*** will be used as a reference document during this step. It contains enhanced descriptions of the items listed in the Checklist and will be used to better understand the items when necessary. The ***IUT's PICS*** may be useful as a reminder as to what functionality is implemented by the IUT.

Figure 3: Relationships between the *BTL Functionality Checklist* and the *BTL Test Plan* documents.

2.1.3 Completing the Functionality Checklist

For information on the columns and codes in the ***BTL Functionality Checklist***, refer to the document description at the start of the ***BTL Functionality Checklist*** document.

For each section in the ***BTL Functionality Checklist***, determine whether the IUT implements the related functionality. Most sections in the ***BTL Functionality Checklist*** correspond either to a BIBB or an Object Type. If the IUT implements the BIBB or can contain objects of the specified type, place an X in the 'Supports' column for the Base Requirements row.

For each row in the section marked as supported, determine whether the IUT implements the functionality described by the row. If the description in the 'Option' column does not provide enough information to understand the functionality, refer to the corresponding section in the ***BTL Test Plan*** document for more information. If more information is needed to determine whether the IUT supports the functionality review the test definition in ***ASHRAE 135.1*** or ***BTL Specified Tests***. If the IUT implements the functionality, place an X in the 'Supports' column.

Figure 4: Part of a completed *BTL Functionality Checklist* document.

This simple device does not support any address discovery.

This device does not support any segmentation options

This device is implemented to a newer version of BACnet and supports the database_revixsion property

This device supports Analog Input and Analog Output objects. In both objects, the device supports writable Out_Of_Service properties.

Support	Listing	Option
Basic BACnet Functionality (applies to all BACnet devices)		
X	R	Base Requirements
	C[1]	Uses static router address configuration
	C[1]	Uses Who-Is and I-Am services for router addres
	C[1]	Uses Who-Is-Router-to-Network (any network) f
	C[1]	Uses Who-Is-Router-to-Network (specific networ
	C[1]	Uses MAC broadcasts for router discovery
[1] At least one of these options must be supported if the service requests other than COV notifications.		
Segmentation - Transmits Segmented Messages		
	R	Base Requirements
	C[1]	Transmits segmented requests
	C[2]	Transmits segmented responses
[1] Required if the device is capable of sending a reques [2] Required if the device is capable of sending a respon		
Segmentation - Receives Segmented Messages		
	R	Base Requirements
	C[1]	Processes received segmented responses
[1] Required if the device is capable of sending confirme response APDU of more than 50 octets in length.		
Device Object		
X	R	Base Requirements
X	C[1]	Supports database revision properties
[1] This option must be supported if the device implemen		
Analog Input Object		
X	R	Base Requirements
X	S	Supports writable Out_Of_Service properties
Analog Output Object		
X	R	Base Requirements
X	R	Supports command prioritization
X	S	Supports writable Out_Of_Service properties

2.2 Step 2 – Plan the IUT Configuration

2.2.1 Purpose

Determine how the IUT should be configured to allow the tests to be executed.

In order for the BTL tests to be applied, the IUT must be configured to exhibit all of the claimed functionality. For example, the IUT will be required to contain at least one instance of each supported object type. The goal of this step is to design an IUT Configuration that is capable of exhibiting all of the IUT's claimed functionality. In some cases, multiple IUT Configurations will be required.

It is possible that the IUT Configuration(s) that is developed in this step may need to be modified when the tests parameters are selected in Step 5 – Select Test Parameters.

Figure 5: Plan the IUT Configuration step.

2.2.2 Requirements

To fill out the design of the IUT Configuration the Test Planner should have the completed ***BTL Functionality Checklist*** and the ***IUT's PICS*** on hand.

The ***BTL Functionality Checklist*** and the ***IUT's PICS*** will be used as references to ensure that the IUT Configuration exercises all of the IUT's functionality.

2.2.3 Designing The IUT Configuration

For each functional item claimed in the completed ***BTL Functional Checklist***, determine the IUT setup required to exercise the functionality. The initial IUT Configuration that is designed in this step may be modified in later steps in this process as the test planning continues.

Note that there is no actual documented output from this step in the process. The IUT's Configuration should be documented in whatever form the Test Planner is comfortable with. In the last step of the Test Planning phase, the IUT's Configuration will be used to generate the ***IUT's EPICS***. EPICS are not generated at this stage as they are generally difficult to generate and maintain and as such are not created until the final IUT Configuration is determined.

If there are two or more functional items that cannot coexist in the IUT at the same time, either multiple IUT Configurations must be provided, or multiple IUTs must be provided. For example, if the firmware supports both Analog and Binary output objects, but no model of the IUT supports both simultaneously, then multiple IUTs or IUT Configurations must be provided.

During this stage, the Test Planner will:

1. Identify all of the object types supported by the IUT. Ensure that at least one instance of each object type is contained in the IUT's Configuration. If the IUT supports multiple forms of an object type (each supporting a different set of properties), then the IUT's Configuration should contain a sample of each form of the object type.

2. Provide property values that allow all conditionally writable properties to be writable. In some IUTs certain properties will only be writable under specific conditions which are usually related to the values of other properties. The Test Planner should choose property values such that conditionally writable properties are writable, and such that the general functionality of the IUT is not hindered.

3. Review all of the functional items supported by the IUT. If, for a specified functional item, the IUT must contain a type of object, or an object configured in a specific manner, then the IUT's Configuration should be modified to contain such an object. For example, if the IUT supports alarming using Event Enrollment objects, an Event Enrollment object should be supplied for each Event-Type that the IUT supports.

4. Identify all of the physical media that the IUT supports. Ensure that either the IUT is equipped with a network connection of each type, or that an instance of the product is available with a network connection of each type.

2.3 Step 3 – Select the Tests

2.3.1 Purpose

Test selection consists of identifying tests that will be applied to the IUT. During this step, the test planner will be creating an initial test plan for the IUT based on the ***BTL Test Plan*** document. This initial test plan will consist of all tests apply to the IUT.

The parameters for the tests will not be chosen in this step; test parameters are identified in *Step 5 – Select Test Parameters*.

Figure 6: Selects the Tests step.

2.3.2 Requirements

The ***BTL Test Plan*** is the template document that will be filled out during this step and is the main working document for this step. The completed ***BTL Functionality Checklist*** is used as a reference document to determine which entries in the ***BTL Test Plan*** need to be included in the ***IUT's Test Plan***.

2.3.3 Selecting Tests

Create a copy of the ***BTL Test Plan*** document for the IUT. This copy will be referred to as the ***IUT's Test Plan*** to distinguish it from the original ***BTL Test Plan*** document. The ***IUT's Test Plan*** will be modified during the remainder of the Test Planning phase to become the custom test plan for the IUT.

For each section in the ***IUT's Test Plan***, if the section is checked in the completed ***BTL Functionality Checklist***, mark the section in the ***IUT's Test Plan*** document. The test plan section can be marked by placing an X in the first column of the 'Notes & Results' row, or the whole section can be highlighted. With highlighting, the selected sections will be easier to find when scanning in the document.

Figure 7: An example of marking a selected test.

Selected tests can be marked by placing an X in first column of the 'Notes & Results' row.

	135.1 - 7.3.1.3 - Command Prioritization Test	
	Test Method	Manual
	Configuration	As per 135.1.
	Test Conditionality	Must be executed.
	Testing Hints	None
X	**Notes & Results**	

The resulting document is the set of tests that apply to the IUT and will be referred to as the ***IUT's Test Plan***.

2.4 Step 4 – Check Test Conditionality

2.4.1 Purpose

To identify tests that cannot be applied to the IUT due to limitations in the test design, or in the IUT's configurability.

Some of the tests in the ***BTL Test Plan*** document have conditions under which the test should be skipped. These conditions are either described in the test definition, or within the test plan entry in the 'Test Conditionality' row. This step will identify the conditional tests and determine which will not need to be applied to the IUT.

Figure 8: Check Test Conditionality step.

2.4.2 Requirements

To evaluate the conditionality of tests the ***IUT's Test Plan***, ***BTL Specified Tests***, and ***ASHRAE 135.1*** will be required. The ***IUT's Test Plan*** will be modified during this step and is the main working document for this step. The ***BTL Specified Tests*** and ***ASHRAE 135.1*** are reference documents used during this step to detail the test steps and to further clarify test conditionality.

Figure 9: 'Test Conditionality' specifies the conditions under a which a test may ve skipped.

2.4.3 Evaluating Conditionality

In each test entry in the ***BTL Test Plan*** document, there is a 'Test Conditionality' field. This field indicates whether the test must always be executed against a device that supports the related functionality, or whether there are conditions under which it may be skipped. The conditions are either described in the 'Test Conditionality' field, or the field will refer to the test definition. The test definition will either be in ***ASHRAE 135.1*** or in the ***BTL Specified Tests*** document.

For each test in the ***IUT's Test Plan***, if there are conditions under which the test shall be skipped, review the conditions to determine if the conditions exist for the IUT. If the conditions exist, mark the test entry as

'Skipped' in the 'Notes & Results' section with a description of why the test was skipped. Otherwise the test will be included in the set of test that must be applied to the IUT.

If desired, the test parameter selection which is described by the next step, can be performed while test conditionality is evaluated.

2.5 Step 5 – Select Test Parameters

2.5.1 Purpose

Determine the parameters for each test in the ***IUT's Test Plan*** that will be applied to the IUT.

Most of the tests that will be applied to the IUT will be customized for the IUT. This customization is achieved through the appropriate selection of test parameters. For script based tests, copies of the scripts will be made for the IUT. These copied scripts will make up the ***IUT's Script Library***.

Figure 10: Select Test Parameters step.

2.5.2 Requirements

To select the parameters for the tests the following documents are needed: the ***IUT's Test Plan***, ***ASHRAE 135.1***, ***BTL Specified Tests,*** the IUT's Configuration, ***BTL Special Test Instructions***, and the ***BTL Scripts Library***.

Before proceeding with this step, review the section in the ***BTL Getting Started Guide*** that discusses test parameter selection.

The ***IUT's Test Plan*** will be used for recording the parameters selected for the tests and will be the main working document for this step. In addition, for any tests that are to be applied using scripts, the script will be copied from the ***BTL Script Library*** into the ***IUT's Script Library*** and configured with the selected test parameters.

The test definitions contained in the ***BTL Specified Tests***, and ***ASRHAE 135.1*** documents will be used during this step to help in identifying and selecting test parameters.

The IUT's Configuration will be reviewed and modified during this test. It is from the IUT's Configuration that most parameters will be selected. In some cases, the IUT's Configuration will be insufficient as designed and may need to be modified to accommodate all selected tests.

While selecting the test parameters, the Test Planner will consider how to cause the IUT to exhibit the tested behaviour. If the Tester will have to take any special actions during the test, these actions should be noted in the ***IUT Special Test Instructions*** document.

2.5.3 Selecting Test Parameters

Almost all BTL tests will require that some parameters be selected. These parameters will generally be objects, properties, new property values, etc and/or properties contained in the IUT. The parameters for a given test are defined in the test script for automated testing, or in the test definition for manual testing.

In a script, the parameters are documented in the parameter section at the top of the script.

Figure 11: An example script parameter section.

```
--------------  S C R I P T   V E R S I O N   I N F O  ---------------
;
; Version 0.1     2003-02-02  Super Dood        - Created
;
---------------------- S C R I P T   H E A D E R ----------------------
;
; Script:   OF-9.32.1.1
; Test:     135.1 - 9.32.1.1  Command Prioritization
;
----------------- S C R I P T   P A R A M E T E R S -------------------
SETUP

TESTED_OBJECT = Analog-Output, 1    ; Set this to the object that will
                                    ;   be tested

V_LOW = 10                          ; These 3 values correspond to the
V_MED = 20                          ;  3 values referenced in the test
V_HIGH = 30                         ;  definition. They will be written
                                    ;  to the present-value property of
                                    ;  the tested object.
                                    ;
P_LOW = 5                           ; These 3 values should be set to
P_MED = 3                           ;   priorities between 1 and 5.
P_HIGH = 1                          ;

;
---------------------- S C R I P T   B O D Y ------------------------
;
...
```

Script parameters are specified and documented in the 'Script Parameters' section of the script.

For manual testing, the parameters consist of all of the items referred to in the test definition for which a specific value is not identified.

Figure 12: An example of a test parameter required for manual testing.

9.32.1.1 Object ID Version with No Device Range

Purpose: To verify that the IUT can correctly respond to a local broadcast Who-Has service request that utilizes the object identifier form and does not restrict device ranges.

Test Steps:

This item is not specified by the test definition. As such it is a parameter that the Test Planner must determine.

1. TRANSMIT
   DA = LOCAL BROADCAST,
   SA = TD,
   Who-Has-Request,
   'Object Identifier' = (any object identifier specified in the EPICS)
2. WAIT **Internal Processing Fail Time**
3. RECEIVE
   DA = LOCAL BROADCAST | GLOBAL BROADCAST,
   SA = IUT,
   I-Have-Request,
   'Device Identifier' = (the IUT's Device object),
   'Object Identifier' = (the object identifier specified in step 1),
   'Object Name' = (the object name specified in the EPICS for this object)

Apply the following steps to each selected test in the ***IUT's Test Plan*** (excluding those that that will be skipped as identified in the previous section).

1. Identify any configuration changes that must be made to the IUT in order to make the IUT exhibit the functionality being tested. If the changes are not of the type that would be represented in the IUT Configuration, document the steps that the Tester must take in a new entry in the ***IUT Special Test Instructions*** document.

Figure 13: An example entry in the *IUT Special Test Instructions* form.

Identify the test that the extra infoirmation is for. This should be entered as found in the ***IUT's Test Plan*** document.

Test:	135.1 - 8.18.1 Reading Non-Array Properties
To cause the Binford 3000 Super Controller to initiate a ReadProperty-request, use the Binford Configuration Tool to create a Binford Poll object and assign a remote object reference to the object reference field. Reset the Binford 3000 and wait 1.5 minutes for the ReadProperty-Request.	
Test:	

Enter instructions for the test in this field.

2. Review the test definition to understand the test and its purpose.
3. If the test is automated, and running the automated test will impose some hindrance upon the Tester, the Test Planner can choose to have the test run manually. In this case, the Test Planner should make a note in the 'Notes & Results' field indicating that the test should be run manually.
4. Review the Testing Hints field in the ***IUT's Test Plan*** for any information on selecting parameters for the test.
5. Refer either to the script for automated tests, or to the test definition for manual tests to determine the parameters that are required.
6. Select appropriate parameters and record them in the Results section. If the IUT Configuration does not contain values that would be appropriate, modify the IUT Configuration as required. If the IUT Configuration cannot be modified to support the test, another IUT Configuration may be required.

Figure 14: An example record of test parameters in the 'Notes & Results' section.

135.1 - 7.3.1.3 - Command Prioritization Test		
	Test Method	Manual
	Configuration	As per 135.1.
	Test Conditionality	Must be executed.
	Testing Hints	None
X	**Notes & Results**	Tested Object = Analog Output, 1 Vlow = 10 Vmed = 20 Vhigh = 30 Plow = 5 Pmed = 3 Phigh = 1

Enter the test parameter values in the 'Notes & Results' field.

7. If the test is automated, save a copy of the script into the ***IUT's Script Library***, and setup the parameter values in the parameter section of the script. Record the name of the copied script file in the 'Notes & Results' section.

Figure 15: An example of a completed script parameter section.

```
-------------  S C R I P T   V E R S I O N   I N F O   --------------
;
; Version 0.1     2003-02-02  Super Dood        - Created
;
--------------------- S C R I P T   H E A D E R ---------------------
;
; Script:   OF-9.32.1.1
; Test:     135.1 - 9.32.1.1  Command Prioritization
;
---------------- S C R I P T   P A R A M E T E R S ---------------
SETUP

TESTED_OBJECT = Analog-Output, 1      ; Set this to the object that w
                                      ;   be tested

V_LOW = 10                            ; These 3 values correspond to the
V_MED = 20                            ;  3 values referenced in the test
V_HIGH = 30                           ;  definition. They will be written
                                      ;  to the present-value property of
                                      ;  the tested object.
                                      ;
P_LOW = 5                             ; These 3 values should be set to
P_MED = 3                             ;   priorities between 1 and 5.
P_HIGH = 1                            ;

;
--------------------- S C R I P T   B O D Y ------------------------
;
...
```

Change the values in the script's 'Script Parameter' section to match the values selected for the test.

Figure 16: An example of a script name recorded in the 'Notes & Results' section.

X	**Notes & Results**	Tested Object = Analog Output, 1 Vlow = 10 Vmed = 20 Vhigh = 30 Plow = 5 Pmed = 3 Phigh = 1 Script = Binford3000-OF-9.32.1.1.vts

Enter the script name in the 'Notes & Results' field.

8. If the testing requires multiple IUT Configurations, record the IUT Configuration to apply the test with in the 'Notes & Results' section.

   If the IUT setup needs to be changed to make it exhibit the desired functionality, and those changes will not be supplied in an IUT Configuration provided to the BTL, record the steps that the Tes ter must take in the ***IUT Special Test Instructions*** form.

For more information on choosing test parameters, refer to section **Error! Reference source not found.**, ***Error! Reference source not found.***.

2.6 Step 6 – Create the EPICS

2.6.1 Purpose

During this step, the Test Planner will generate an ***EPICS*** document for each of the IUT Configurations. The ***IUT's EPICS*** will be used during the testing of the IUT in the next step *Step 7 – Apply the Tests*.

Figure 17: Create the EPICS step.

2.6.2 Requirements

The Tester will require all of the completed IUT Configurations to generate the ***IUT's EPICS***. The ***IUT's PICS*** contain much of the same information as the ***IUT's EPICS*** and may be useful as a reference when creating the ***IUT's EPICS***. The syntax and structure of an EPICS is defined in ***ASHRAE 135.1***. ***ASHRAE 135.1*** and the ***BTL Sample EPICS*** can be used reference documents during this step.

2.6.3 Generating an EPICS

The B.I.G-E.U. has created a tool to aid in the generation of EPICS documents. This tool is available from the BTL website. In addition, a sample EPICS has been included in the documentation package that can be modified to match the IUT. Refer to ***ASHRAE 135.1*** for a detailed description of the EPICS syntax.

The Test Planner should create an EPICS for each of the IUT Configurations that have been generated. The EPICS must match the IUT Configuration exactly. If it does not match then some of the tests will fail.

When the EPICS is complete, load it into VTS to ensure that it is well formed.

3 Test Execution

The test execution phase applies the IUT's custom test plan to the IUT.

3.1 Step 7 – Apply the Tests

3.1.1 Purpose

In this step the Tester will apply the tests identified in the ***IUT's Test Plan*** to the IUT, recording the results in the ***IUT's Test Plan***.

Figure 18: Apply the Tests step.

3.1.2 Requirements

The Tester should ensure that the Test Planner provided the Tester with the ***IUT's Test Plan***, the ***IUT's Script Library***, all of the IUT Configurations and each associated ***IUT EPICS***, and the tools to change the IUT's configuration.

Before proceeding with this step, the ***IUT's Test Plan*** will have had the tests to apply selected, and configured. The Tester will use the ***IUT's Script Library*** to apply script-based tests. For manual tests, the Tester will follow the steps laid out in the ***BTL Manual Test Instructions*** document.

The Tester should expect that some of the test definitions will be referred to during the testing and as such should have ***ASHRAE 135.1*** and ***BTL Specified Tests*** available as reference documents.

The main working document during this step is the ***IUT's Test Plan*** which not only identifies the tests to be applied but is also used as a repository for the test results.

Before proceeding with this step, the Tester should have a reasonable understanding of VTS and the other test tools that will be required. For more information on how to use VTS, refer to the ***VTS Usage Guide***.

3.1.3 Executing Tests

The ***IUT's Test Plan*** will indicate all of the tests that apply to the IUT. Each entry in the test plan that applies to the IUT will be marked with an *X* next to the 'Notes & Results' field. If a test that applies to the IUT should not be executed, the Test Planner will have noted this in the 'Notes & Results' field.

Figure 19: An example of an automated test that the Test Planner decided should be run manually.

Even though this test applies to the IUT, the Test Planner has determined that applying this test will be cumbersome. The Test Planner has instructed the Tester to skip this test and execute 9.1.19.X2 in its place (as allowed by the *Test Conditionality* field).

135.1 - 7.2.2 Write Support Test Procedure		
	Test Method	Manual Test 3.3
	Configuration	As per 135.1.
	Test Conditionality	If the tester concludes that execution of 7.2.2 then the tester may execute 9.1.19.X2 on at le datatype in place of 7.2.2. To satisfy this test item, test 7.2.2 need only b
	Testing Hints	None.
X	**Notes & Results**	Due to the number of writable properties in thi this shall be skipped and 9.1.19.X2 shall be ex See the test plan entry for 9.1.19.X2 for detail

Before executing any of the tests, the Tester should determine if the Test Planner developed multiple IUT Configurations. If there are multiple IUT Configurations, then the Tester may decide to determine the order in which to apply the tests based on the IUT Configuration used by each test.

1. Read the test plan entry. Evaluate and understand all of the fields.
2. Review the test definition in ***ASHRAE 135.1*** or ***BTL Specified Tests***.

Figure 20: Identifying the location of a test plan entry's test definition.

This field identifies the test. The test definition can either be found in ***135.1*** or the ***BTL Specified Tests*** document. This field will indicate which document the definition is located in.

135.1 - 7.2.2 Write Support Test Procedure		
	Test Method	Manual Test 3.3
	Configuration	As per 135.1.
	Test Conditionality	If the tester concludes that execution of 7.2.2 then the tester may execute 9.1.19.X2 on at le datatype in place of 7.2.2. To satisfy this test item, test 7.2.2 need only b
	Testing Hints	None.
X	**Notes & Results**	Due to the number of writable properties in thi this shall be skipped and 9.1.19.X2 shall be ex See the test plan entry for 9.1.19.X2 for detail

3. Ensure that the IUT and network setup match the configuration as defined by the 'Configuration' field of the test plan entry.

Figure 21: Evaluating test configuration.

The IUT and network configuration are indicated in this field. Most test plan entries will refer back to the test definition for general configuration information.

The Test Planner will have documented the test parameters in the 'Notes &Results' field.

135.1 - 7.3.1.12 - Notify Type Test		
	Test Method	Manual
	Configuration	As per 135.1.
	Test Conditionality	If the IUT cannot be configured to meet the 135.1 configuration then this test shall be skipped.
	Testing Hints	None
X	**Notes & Results**	Test Object = Event Enrollment, 1

4. Determine whether the test is to be executed manually or via script. This is specified in the 'Test Method' field in the test plan entry. Note that the Test Planner may have opted to have a test applied manually in which case the Test Planner will have made a note in the 'Notes & Results' field indicating that the test should be executed manually. If the test is to be applied manually skip to step 8.

Figure 22: Determining the whether the test will be run manually, or via a script.

The 'Test Method' field will indicate whether the test will be run manually or via script. This example shows that the tester should apply a specific manual test (instead of the *General Manual Test Method*).

The 'Test Method' field will indicate whether the test will be run manually or via script. If it is to be run via a script, this field will indicate the name of the script to execute. The script can be found in the ***IUT's Script Library*** where the Test Planner placed it.

Sometimes the Test Planner will decide that even though a standard script is available, that the test should be run manually. The Test Planner will note this in the ' Notes & Results' field.

135.1 - 7.1 Read Support for Properties in the Test Database	
Test Method	Manual Test 3.2
Configuration	As per 135.1.
Test Conditionality	Must be executed. To satisfy this test item, test 7.1 need
Testing Hints	None
Notes & Results	

135.1 - 9.20.1.1 Reading a Single Property from a Single Object	
Test Method	DS-9.20.1.1.vts
Configuration	As per 135.1.
Test Conditionality	Must be executed.
Testing Hints	None
Notes & Results	Test Object = Analog Output, 1 Property = Max_Pres_Value Script = Binford3000-DS-9.20.1.1.vts

135.1 - 9.20.1.2 Reading Multiple properties from a Single Object	
Test Method	DS-9.20.1.2.vts
Configuration	As per 135.1.
Test Conditionality	Must be executed.
Testing Hints	None
Notes & Results	Test Object = Calendar, 5 Property1 = Description Property2 = Object_Identifier Property3 = Object_Name Run this test manually because the scri

Script Based Testing:

5. Start VTS, and load the ***IUT's EPICS***. For instructions on how to use VTS, refer to the ***VTS Guide***.
6. Determine the script to run. The test planner will have documented this in the 'Notes & Results' field. The script should be located in the ***IUT's Script Library*** where the Test Planner saved it. Load the script into VTS and execute it. For instructions on how to run scripts using VTS, refer to the ***VTS Guide***.
7. Skip to step 10.

Manual Testing:

8. Determine which manual test to execute. If the 'Test Method' field does not indicate a specific manual test then the *General Manual Test Method* should be applied. The ***BTL Manual Test Instructions*** document describes all manual tests, including the *General Manual Test Method*.
9. Follows the steps outlined in the ***BTL Manual Test Instructions*** document.

Both Testing Methods:

10. Record the time and date that the test was executed.
11. Record the result of the test in the 'Notes & Results' field of the test plan entry. Indicate a Pass or Fail by placing a P or F under the X in front of the 'Notes & Results' field of the test plan entry.
12. If the IUT failed the test, indicate the step in the test definition that failed, and how the IUT failed the test step.

13. Regardless of whether the IUT passed or failed the test, record any interesting information such as, "the IUT took longer than usual to process the request to respond during step 6", or "the IUT generated a number of unrelated ReadProperty-Requests after step 3".

4 Test Parameter Selection Hints

This section discusses how to select test parameters so as to maximize the probability of finding defects in the IUT. The purpose of pre-testing a BACnet product is to increase the probability that the IUT will pass the test when applied by the BTL. Good parameter selection during pre-testing is important to realize this goal. Keep in mind that the BTL's tester will be trying to find the bugs in your device, so he is likely to choose parameter values that will stretch the limits of your device.

1. Some tests will provide more benefit if they are executed multiple times with varying parameters. The pre-test package is developed such that each test is executed once, but the Test Planner can always choose to have a test applied two or more times. If this is the case, the Test Planner should provide multiple sets of parameters in the 'Notes & Results' field of the test plan entry.
2. When selecting parameter values, try not to select the normal values that would be encountered. Odds are that those values have already been tested and work. Try values at each end of the valid value range, such as the minimum and maximum values.
3. Negative tests are those that ensure that the IUT responds correctly to invalid requests, such as invalid parameter values. History has shown that these tests commonly fail. Repeating these tests with a variety of unexpected parameters may reveal many bugs. Use your imagination!
4. Tests that stretch the limit of internal buffers are also known to result in numerous bugs. If the test allows for a selection of data values that will significantly increase the size of the request, or the response, the Test Planner should attempt to choose values that do so. An example would be to choose character string values that are very long when writing to character string properties.

5 BTL Lab Testing

This section contains an overview of the process followed by the BTL lab in testing products.

5.1 Processing a Testing Application

The BTL reviews the testing application before scheduling a product for lab testing. The main purpose of this review is to understand the product's BACnet capabilities. In some cases, testing of a product may have to be delayed if the product contains a substantial amount of BACnet functionality that the BTL has not yet been authorized to test or if the product does not fit into one of the product classes that the BTL is authorized to test.

5.2 Pre-Testing Support

The BTL's test engineer will provide limited free e-mail and telephone support to a test engineer who is pre-testing a product in preparation for submission of the product to the BTL lab. This support includes providing guidance on how to install and use the BTL's recommended testing tools, assisting with EPICS creation, and answering questions about the BTL's test plan and test execution. The BTL also offers paid pre-testing and training services.

5.3 The First Testing Pass

After the manufacturer submits a product for testing, the BTL lab reviews the submitted documents (PICS, EPICS, functionality checklist, etc.) and prepares a customized test plan using the ***BTL Test Plan***. The customized test plan should provide good coverage of the product's BACnet capabilities, but in order to improve test coverage the BTL's test engineer may also develop and execute various "free-form" tests of BACnet functionality that are not documented in the ***BTL Test Plan***.

The following rules regulate free form testing by the BTL lab:

1. The amount of time spent on free form testing shall not exceed 25% of the total testing time.
2. If running a free-form test leads to the detection of a BACnet compliance problem, and if the problem can also be observed by applying a single test contained in the ***BTL Test Plan***, then the product will fail (i.e. the problem must be fixed before the product may receive a BTL listing).
3. If running a free-form test leads to the detection of a BACnet compliance problem that is not considered severe, and if the problem cannot be observed by applying a single test contained in the ***BTL Test Plan***, then the problem will not cause the device to fail, which means that the manufacturer may obtain a BTL listing for the product before the problem is fixed. The BTL will document these cases, along with a suggested new test for each problem, and present a request for the new test to the BTL-WG. If the BTL-WG sanctions the new test, the BTL will inform the manufacturer that the product must be resubmitted within one year of the BTL-WG sanctioning the new test in order for the BTL to verify that the problem has been fixed. If the BTL-WG does not sanction the new test (or a similar test that would have uncovered the problem) then the manufacturer is not required to fix the problem.
4. If a BACnet compliance problem is found and is considered severe, the procedure outlined in 5.6 shall be followed.
5. The manufacturer does not pay for free form testing.

For each problem that the test lab finds in a product, the BTL will provide the following information to the manufacturer so that the manufacturer can reproduce the problem:

- a short description of the problem,
- a reference to a test in the BTL Test Plan, or in the case of free-form testing, the sequence of test steps used when the problem was discovered,
- the test parameter values,
- the test step where the problem was detected,

- the severity of the problem from the BTL's standpoint, and
- a protocol analyzer log, if available, with an indication of the packets relevant to the problem.

Following the receipt of a problem report, the manufacturer may contact the BTL for more information or to ask for a re-evaluation of a problem.

5.4 Regression Testing

The BTL lab finds problems in almost every product that is submitted for testing. Here is a description of the BTL's regression testing process, starting with the first testing pass:

0. The BTL lab runs a suite of tests on Manufacturer's product. (This is the first testing pass.)
1. The BTL lab reports all problems found in the product to Manufacturer.
2. Manufacturer reproduces all reported problems.
3. Manufacturer may request a re-evaluation of one or more problems by the BTL lab.
4. If there are no problems that are required to be fixed, go to step 9.
5. Manufacturer fixes the problems and runs his own regression tests.
6. Once Manufacturer is confident that the product will meet all of the BTL requirements, Manufacturer sends a product or firmware update to the BTL lab.
7. The BTL lab selects specific regression tests and test parameters, and then the lab runs those tests.
8. If one or more regression tests failed during step 7, go to step 1.
9. The BTL lab reports to Manufacturer that the product passed all required tests. The product now qualifies for a BTL listing.

Regression tests are selected by the BTL lab (in step 7) in order to check whether previously identified problems have been fixed. In most cases, the BTL lab will select tests that previously failed, although closely related tests that previously passed and new free-form tests may also be included.

For each regression test, the BTL lab chooses a reasonable number of sets of test parameter values. The values will typically include a mix of new parameter values and values that were previously used.

5.5 Product Retesting Requirements

A BTL listing will be awarded to a product if the product meets the BTL's current requirements. However, partial or complete retesting of a BTL-listed product is required under certain circumstances. Generally, the BTL will evaluate the need for retesting once per year around the anniversary of the awarding of a BTL listing to a product. The manufacturer of a BTL-listed product is responsible for providing the BTL with contact information for a person who is authorized to respond to the BTL's correspondence about the company's BTL-listed products.

Here are some specific circumstances that will lead to the requirement for retesting of a BTL-listed product:

- BACnet functionality is added to the product.
- Significant changes are made to the product's BACnet protocol stack implementation.
- The product contains BACnet functionality that was untested by the BTL, but the BTL has subsequently acquired the capability to test that functionality.
- The BTL-WG sanctions a new test that the product failed during free form testing (as described previously).
- The BTL is provided with documented evidence of a significant interoperability problem in the product.

The periodic retesting evaluation process begins with the BTL sending a letter to a manufacturer requesting information that the BTL will use to determine whether retesting is required. The manufacturer must respond to this letter within one month; failure to respond to the letter and to a follow-up call from the BTL means that complete retesting may be required. The BTL evaluates the information provided by the

manufacturer, and if the BTL determines that retesting is required then the BTL will send a letter to the manufacturer requesting that the product be submitted to the BTL lab within 6 months for retesting.

The scope of retesting is determined by the BTL based on information provided by the manufacturer. Complete retesting of a product will usually not be required unless there have been many significant changes to the BACnet protocol stack. The BTL will provide the manufacturer with information detailing the scope of testing to be performed.

5.6 Severe Problem Resolution Procedure

In the event of a disagreement between the manufacturer and the BTL over the resolution of a severe problem in the product discovered during testing, but which is not covered by existing tests, the following procedure shall apply. A "severe" problem is defined here as a problem that is likely to prevent the device from interoperating with other devices in an otherwise properly functioning system.

1. The BTL lab manager reviews the problem. If he deems it to be a severe problem he reports it to the manufacturer as such. If he does not deem it to be a severe problem, this procedure is terminated.

2. If the manufacturer agrees to fix the problem, this procedure is terminated.

3. The BTL lab documents the problem and its reasoning why this is a severe problem that must be fixed and presents it to the manufacturer and the BTL lab manager.

4. The manufacturer either presents a written rebuttal to the BTL lab manager that will be appended to the severe bug report, or declares that he is waiving the right. There is no time limit set for this step.

5. The BTL lab manager convenes a review committee with five representatives from BMA members, affiliates and members of other organizations with which the BMA has reciprocal testing agreements, not including the manufacturer, and presents the BTL lab's documentation and reasoning, along with the manufacturer's rebuttal, if any, to the review committee.

6. The review committee evaluates the documentation and rebuttal. The review committee is permitted to ask questions of the BTL lab and the manufacturer in the course of its evaluation.

7. If the review committee does not agree that the problem is severe and must be fixed, this procedure is terminated and the BTL lab shall not refuse to award the BTL Mark because of this problem.

8. If the review committee agrees that this is a severe problem that must be fixed, it shall produce a test for this problem within one month of being convened. If it fails to do so, this procedure is terminated and the BTL lab shall not refuse to award the BTL Mark because of this problem.

9. The BTL lab performs the test upon the device. If the device passes the test, the BTL shall not refuse to award the Mark because of this problem. If the device fails the test, it must be fixed in order to receive the BTL Mark, or the manufacturer can elect to appeal to the BMA per the procedures in the document "BTL Appeals Proc."

5.7 Bug Tracking and Reporting

Bug reports are an important means of communication between the BTL lab and the product manufacturer.

In the past, the bug reporting mechanism was based on a Word document that was passed back and forth between the test lab and the manufacturer, with both parties editing the document. This mechanism was cumbersome and inconsistent.

The BTL lab is currently using a bug-tracking software package called TestTrack Pro. This package allows the lab to track issues and communicate test results to testing customers in a consistent way. Manufacturers use client software to access the bug database maintained by the lab.

The BTL working group has asked the test lab to provide the following information about bugs:

1. Identification of the standard (BTL-approved) test that was run, test parameters that were used and the test step that failed

2. In the case of a free-form test:
   - A description of the test
   - Reference to the clause(s) in the BACnet test standard, BTL requirements, or other documents that are being violated
   - The set of parameters used for the specific test
   - Potential impact on interoperability

3. Standard bug severity classification, such as the following:
   - Fix Required: failure of a standard test or (in some cases) a free-form test
   - Fix Suggested: BACnet standard violation (could require a 1-year retest) or Device implementation guidelines violation
   - FYI: unusual/undesirable product behaviour observed by the lab

The BTL working group has also discussed a standard way of identifying the state of the bug through its life cycle, but discussion of this is ongoing. The following states have been proposed:

- Reported: The BTL has reported the issue to the applicant.
- Fixed: The Applicant claims that the problem has been resolved.
- Re-reported: The BTL re-tested the problem after it was reportedly resolved and it still exists, or recurs after having been previously accepted.
- Contested: The Applicant contests the bug report, and the BTL is (or will be) in the process of considering the dispute.
- Passed: The Applicant fixed the bug, then the BTL tested and accepted the fix.
- Retracted: The Applicant successfully argued that the reported bug is not a problem
- Acknowledged: The Applicant acknowledged a bug that is not required to be fixed.

5.8 Test Reporting and Statistics

The BTL is restricted by confidentiality rules in how it can communicate about test results. However, it is useful to be able to talk about test results in general terms so that the BTL working group can improve its test procedures and the device implementation guidelines. There is also a need for the quality of the BTL lab's performance to be assessed.

To that end, the BTL working group requires that the officially sanctioned test lab(s) do the following:

The test lab should track all reported bugs. Statistics tracked by the BTL should include:

- The minimum/maximum/average number of bugs found per product.
- The number of bugs found by a specific test (both officially sanctioned tests and free-form tests).
- The number of bugs found counted by product-test iteration.
- The number of bugs found counted by severity classification.
- The number of bugs reported but not fixed as a percentage of all bugs reported.

In order to support the tracking and reporting of bugs to the BTL working group and to the test lab's customers, the test lab will use issue-tracking software. The BTL working group will need to obtain various summaries, so the test lab should track the device type, testing round, and test package used.

The BTL shall record every successfully contested finding. Statistics on successfully contested findings shall be generated with successfully contested findings grouped into the following classifications (types):

1. Errors in application of the test, reading of the test results, reading of the devices capabilities, or reading of the application documentation.

2. Errors in sanctioned tests, overly/under-restrictive test plan entries, overly/under-restrictive checklist entries.

3. Errors in interpretation of the standards (ASHRAE 135 and 135.1).

4. Re-evaluated resolution requirements (the applicant successfully argues that the fix is not required even though there is agreement that it is a bug).

Details of every successfully contested finding of type 2 or type 3 shall be made available to the BTL-WG for review and remedy. The BTL shall publish a list of such contestations and include the list in the test package.

Version	Date	Author	Change
0.17	12-Oct-2004	Bill Swan	• Severe Problem Resolution Procedure
0.18	19-Oct-2004	Bill Swan	• Updated Severe Problem Resolution Procedure
0.19	25-Jan-2005	Carl Neilson	• Fixed typos
0.20	02-Feb-2005	Carl Neilson	• Formatted new text.
0.21	11-Jul-2005	Jim Butler	• Added section on test reporting and statistics
0.22	26-Oct-2005	Jim Butler	• Added section on bug tracking and reporting