Clarification Request

References: BTL Specified Tests 7.2.2.X2

Background / Proposed Solution:

This test is designed to verify the IUT 1) responds with the correct error message, 2) represents the provided EPICS correctly.

The problem with this test is it takes a long time to run on devices which contain hundreds if not thousands of objects in the database.

For example, assume this test takes 2 seconds to read 1 property out of 1 object. We are testing 511 properties minus the objects actual properties. We can assume about 30 properties per object are defined and therefore we must test 481 properties per object. This requires 16 minutes per object. In a device with 400 objects, this requires 106 hours to run the test. If the IUT fails to respond correctly to a request, the failure timeout is around 4 seconds per request.

7.2.2.X2 Non-documented Property Test

Reason For Change: There is no test that verifies that all supported object properties are included in the EPICS. This test is included in the SSPC proposal CN-116.

Purpose: To verify that all properties contained in every object are documented in the EPICS.

Test Concept: For each object in the EPICS database, attempt to read each standard property that the EPICS does not document as being part of the object.

Test Steps:

```
1.  REPEAT X = (all objects in the IUT's database) DO {
        REPEAT Y = (0 through 511) DO {
            IF (the property Y is not in the EPICS for object X) THEN
                TRANSMIT ReadProperty-Request,
                    'Object Identifier' =       X,
                    'Property Identifier' =     Y
                RECEIVE BACnet-Error-PDU,
                    Error Class =       PROPERTY,
                    Error Code =        UNKNOWN_PROPERTY
        }
    }
```

Question:

I propose we modify the requirement to test any defined property not supported by the object (i.e. Property Identifier < 200 or use the list of properties defined for each object type), then also test property identifier 255, 256, and 511. This will reduce the number of properties significantly. I

also propose allow the tester to randomly select at least 2 instances of each object type defined in the IUT's database instead of requiring all objects to be tested.

Response:

In seeking an undocumented property, it is important to look in each place where one might be lurking.

Add this section:

Notes to Tester: The objects selected by the tester should include one instance of each supported object type. Where some instances of an object type differ in the set of supported properties, the allowable value ranges for a property, or the writability of a property, then one instance of each “flavor” of that object type should be selected.

and change the first part of Step 1 as here indicated:

1. REPEAT X = (~~all objects in the IUT's database~~ *a tester selected set of objects*) DO {
   REPEAT Y = (0 through 511) DO {
   IF (the property Y is not in the EPICS for object X) THEN