

## Clarification Request

**References:** BACnet 135-2012 clause 15.9.1.3.1, 135.1-2013/BTL - 9.22.2.1

**Date of BTL-WG Response:** \_\_19-Feb-2015\_\_

☒ All actions necessitated have been completed

### Background:

From 135-2012, clause 15.9.1.3.1

### Error Type

This parameter consists of two component parameters: (1) an 'Error Class' and (2) an 'Error Code'. See Clause 18. The 'Error Class' and 'Error Code' to be returned for specific situations are as follows:

<u>Situation</u>	<u>Error Class</u>	<u>Error Code</u>
Specified object does not exist.	OBJECT	UNKNOWN_OBJECT
Specified property does not exist.	PROPERTY	UNKNOWN_PROPERTY
An array index is provided but the property is not an array.	PROPERTY	PROPERTY_IS_NOT_AN_ARRAY
An array index is provided that is outside the range existing in the property.	PROPERTY	INVALID_ARRAY_INDEX
The specified property is currently not writable by the requestor.	PROPERTY	WRITE_ACCESS_DENIED
The datatype of the value provided is incorrect for the specified property.	PROPERTY	INVALID_DATATYPE
The property is Object_Name and the name is already in use in the device.	PROPERTY	DUPLICATE_NAME
The property is Object Identifier and the identifier is already in use in the device.	PROPERTY	DUPLICATE_OBJECT_ID
The value provided is outside the range of values that the property can take on.	PROPERTY	VALUE_OUT_OF_RANGE
There is not enough space to store the new value.	RESOURCES	NO_SPACE_TO_WRITE_PROPERTY
The data being written has a datatype not supported by the property.	PROPERTY	DATATYPE_NOT_SUPPORTED
The Priority parameter is not within the defined range of 1..16. This condition may be ignored if the property is not commandable.	SERVICES	PARAMETER_OUT_OF_RANGE

From BTL - 9.22.2.1 adapted from 135.1-2013

### 9.22.2 Negative WriteProperty Service Execution Tests

#### 9.22.2.1 Writing Non-Array Properties with an Array Index

Reason for Change: Modified test to remove dependency on EPICS values.

Purpose: To verify that the IUT can execute WriteProperty service requests when the property value is not an array but an array index is included in the service request.

Test Concept: The TD shall select an object in the IUT that contains a writable scalar property designated P1. An attempt will be made to write to this property using an array index. If no suitable object can be found, then this test shall be omitted.

Configuration Requirements: If the IUT supports any writable properties that are scalars, it shall be configured with at least one such property that can be used for this test.

Test Steps:

1. *READ*  $X = (\text{Object1}), P1$
- ~~1. *VERIFY*  $(\text{Object1}), P1 = (\text{the value defined for this property in the EPICS})$~~
2. TRANSMIT WriteProperty-Request,  
     'Object Identifier' = Object1,  
     'Property Identifier' = P1,  
     'Property Value' = (any *valid* value of the correct datatype for this property subject to the restrictions specified in the EPICS as defined in 4.4.2, except the value  $X$  read in step 1),  
     'Property Array Index' = (any positive integer)
3. IF (Protocol\_Revision is present and Protocol\_Revision  $\geq 4$ ) THEN  
     RECEIVE BACnet-Error PDU,  
         Error Class = PROPERTY,  
         Error Code = PROPERTY\_IS\_NOT\_AN\_ARRAY  
     ELSE  
         RECEIVE BACnet-Error PDU,  
         Error Class = SERVICES,  
         Error Code = INCONSISTENT\_PARAMETERS
4. ~~VERIFY  $(\text{Object1}), P1 = X(\text{the value defined for this property in the EPICS})$~~

#### Discussion:

The situation is actually that at Protocol\_Revision 4 and higher, by clause 15.9.1.3.1 only  
 Error Class = PROPERTY,  
 Error Code = PROPERTY\_IS\_NOT\_AN\_ARRAY  
 is correct.

Only when IUT claims Protocol\_Revision less than 4, then also permissible is:  
 Error Class = SERVICES,  
 Error Code = INCONSISTENT\_PARAMETERS

The observed 'Error Class': SERVICES 'Error Code': INCONSISTENT\_PARAMETERS should have been reported during testing at Protocol\_Revision 4 and above, so in all the device versions which have been seen across past 4 1/2 years. And the different error code pair being returned only by the Program object type, observed by varying the Object1 parameter and brought to light only this month, returning 'Error Class': PROPERTY 'Error Code': INVALID\_ARRAY\_INDEX is wrong too.

This test has been wrong regarding which Error Code pairs it would accept when IUT claims Protocol\_Revision less than 4. The test should also specify that the correct clause 15.9.1.3.1 Error Code pair, mandated at Protocol\_Revision 4, is also allowed earlier.

#### Question:

The mistake in the test enduring across 8 years, and five published versions of 135.1, actually poignantly illustrates that distinguishing error code returns is not necessary to interoperability. If

BTL-WG wants interoperability and not just pedantry, then isn't it sufficient to just test that a server indicates Result(-) when it can't do what it is asked, and not enforcing which code?

If it is a matter of BTL-WG applying the tables in the standard to any and all situations where they think that is a mandate, then consider: isn't matching the cost of reporting and fixing these kinds of distinction-without-a-difference defects, which makes them a deterrent to BTL participation for some potential applicants, a higher mandate?

**Response:**

**The BTL-WG holds the position that testing error codes allows for description of error conditions to the end user, and automated fail-over for certain error conditions, and improves the ability to debug misconfigured systems.**