

Clarification Request

References: BTL Test 7.3.2.23.9 - Exception_Schedule Size Change Test
BTL Test 9.22.1.X1
BTL Test 7.3.1.X3
BTL Test 9.22.1.X4

Question:

The Exception_Schedule property is writable and the number of array entries in Exception_Schedule is not modifiable. Any write at index 0 will return a WRITE-ACCESS-DENIED.

Should I change my answer for a VALUE-OUT-OF-RANGE when someone tries to write in index 0?

For the Weekly_Schedule (also a BACnetArray), who is writable and the number of array entries is not modifiable. Any write at index 0 will return a INVALID-ARRAY-INDEX. I don't really understand why our driver has a different behavior on that.

Should I apply that change on both, Exception_Schedule and Weekly_Schedule?

Response:

Returning WRITE-ACCESS-DENIED or VALUE-OUT-OF-RANGE when someone tries to write in index 0 of the Exception_Schedule property, when the number of array entries in Exception_Schedule is not modifiable, are both OK. Contrarily, WRITE-ACCESS-DENIED is probably slightly more correct than a VALUE-OUT-OF-RANGE when someone tries to write in index 0 of the Weekly_Schedule property, since index 0 of the Weekly_Schedule property is explicitly read-only in the standard (by implication since the length of 7 is mandated and immutable.) But the BTL-WG doesn't deem the choice of another error-response when someone tries to write in index 0 of the Weekly_Schedule property to be an interoperability issue that merits inclusion in the test plan, since there should be no one depending upon the result of trying to write something which is explicitly read-only in the standard. It shouldn't work.

Ancillary to the discussion of the Clarification's original question however, the device behavior to store a property value with five BACnetSpecialEvents when the device receives a WriteProperty_Request bearing a propertyValue with two BACnetSpecialEvents, is not sanctioned. Though 135-2008x-3 specifies: "The ordering of list elements when a list is written or modified is not required to be preserved upon subsequent reading of the same list, even if the set of elements that make up the list has not changed." Note that the standard allows reordering a list, but it does not let you create new entries nor have fewer than were written.

In this case of the Exception_Schedule, the behavior is identical to what would be produced by BTL Test 7.3.2.23.9 - Exception_Schedule Size Change Test, and that in fact because of the empty list of BACnetTimeValue, the effect is semantically identical to that of the original WriteProperty propertyValue provided. In the general case, you can't say that will be semantically identical. In a case such as State_Text[0], it is not.

If a device behavior is implemented to store a property value with five BACnetSpecialEvents at all times, then when the device receives a WriteProperty_Request bearing a propertyValue with any other number of BACnetSpecialEvents, it shall return a VALUE-OUT-OF-RANGE.

The BTL-WG recognizes: there is no test in Test Plan 5.0.final which would fail for this behavior, except if the test 9.22.1.3 Writing a Non-Commandable Property with a Priority, is contrived to do so, by enforcing Step 4. They acknowledge that judicial activism that affects a device in testing, is too heavy handed. Perhaps there is a line of products for which the A-side behavior is generation of a WriteProperty_Request bearing a propertyValue with fewer than five BACnetSpecialEvents.

A BACnetError-PDU may not be interoperably acceptable to the A-side, with its current implementation. So it would be unfair to fail the device currently in testing for this issue against Test Plan 5.0.final

The BTL-WG will, however, formally advise that all future BTL-WG Test Plans will not have this loophole. It will fail if this behavior is exhibited by a device that submits for testing against a future BTL-WG Test Plan.

The vendor will be asked if there is no impediment to their implementing the specified behavior, then please do so: store the original WriteProperty propertyValue provided, or return a BACnetError-PDU. Do not create new entries nor have fewer than were written.