

Clarification Request

References: BTL Test Plan 5.0, BTL Specified Tests, 135.1-2007

Date of BTL-WG Response: __21-Jul-2011__

Background / Proposed Solution:

What I see as problem is that in this case, the standard makes the assumption that when

- there was a transition for this state *_ever before_*,
- even if it was years ago,
- even if the transition was already acked!,

as long as the event state matches any transition, then it must be the timestamp that is wrong and so INVALID_TIME_STAMP has to be sent. But this is just an assumption! It could be the event state that is wrong.

Yellow highlighting *italics* and ~~strike through~~ in the test represented below, shows the differences between the 135.1-2007 version and the BTL Specified Tests-5.0.final version.

9.1.2.7 Unsuccessful Alarm Acknowledgment of Unconfirmed Event Notifications Because the 'Event State Acknowledged' is Invalid

Reason for Change: This test was updated to account for revision 5 specifications. There is no new SSPC proposal.

Purpose: To verify that an alarm remains unacknowledged if the 'Event State Acknowledged' is inconsistent with the other parameters that define the alarm being acknowledged.

Test Concept: An alarm is triggered that causes the IUT to notify the TD and at least one other device. The TD acknowledges the alarm using an invalid 'Event State **Acknowledged**' and verifies that the acknowledgment is not accepted by the IUT and that the IUT does not notify other devices that the alarm was acknowledged. The TD then acknowledges the alarm using the proper 'Event State **Acknowledged**' and verifies that the acknowledgment is properly noted by the IUT. The IUT notifies all other recipients that the alarm was acknowledged.

Configuration Requirements: The IUT shall be configured with at least one object that can detect alarm conditions and send **un**confirmed notifications. The Acked_Transitions property shall have the value B'111' indicating that all transitions have been acknowledged. The TD and at least one other BACnet device shall be recipients of the alarm notification.

Test Steps: The test steps defined in 9.1.2.5 shall be followed except that in the first AcknowledgeAlarm request the 'Time Stamp' shall have the same value as the 'Time Stamp' from the event, *the 'To State' in the notification shall be any offnormal transition* and the 'Event State Acknowledged' shall have an **offnormal** value that is different from the 'To State' in the event notification *and shall not be OFFNORMAL*.

Notes to Tester: A passing result is the same message sequence described as the passing result in 9.1.2.5 except that the Error Code in step 7 shall be *INVALID_EVENT_STATE*. For devices claiming a Protocol Revision less than 5, an Error Code of *INCONSISTENT_PARAMETERS* shall also be allowed.

I don't mind BACnet's habit of having very specific error codes, but they should not pretend to know more than they do.

IMVHO the error codes are not reliable and it's better to merge them into something more generic or to allow all devices to send *SERVICES - INCONSISTENT_PARAMETERS* again - because that is what is actually true - the parameters are mutually exclusive and the user has to decide, whether he corrects the timestamp or the event state parameter. The device can't tell.

The intended logic should look something like:

```
if transition_type(ack.eventstate) == OFFNORMAL
    if ack.eventstate <> OFFNORMAL and
object.lastoffnormalstate <> ack.eventstate
    return INVALID_EVENT_STATE
    If object.eventtimestamps[transition_type(ack.eventstate)] <>
ack.timestamp
    return INVALID_TIMESTAMP
```

Question:

Should 9.1.2.1, 9.1.2.4, 9.1.2.5 and 9.1.2.7 success criteria be relaxed to allow all devices to send *SERVICES - INCONSISTENT_PARAMETERS*, or either *SERVICES - INVALID_EVENT_STATE* or *SERVICES - INVALID_TIME_STAMP* error code, if either could be applicable, for each of the tests?

Response:

No. The 'To State' in the notification shall be an offnormal transition and the 'Event State Acknowledged' shall have a value that is different from the 'To State' in the event notification and shall not be OFFNORMAL. The only combinations currently matching that are High_Limit vs Low_Limit and Low_Limit vs High_Limit.