



Testing and Conformance Clarification-Request No.: 20071105-14

(Request-Nr, assigned by moderator: yyyyymmxx where xx=sequence# within the month)

Request from: Frank.Schubert@mbs-software.de

Stage:

- Request
- Listed by moderator
- Analysis by (TGTC or individual):.....
- Resolved

Reference: [referenced document(s) with number and revision]
BTL 7.3.2.23.9

Background:

1. Name should be changed to contain 'X' somewhere in the name (according to the naming conventions).
2. Wrong calculation of total_record_count

Question / proposed solution:

```
# IF ( Total_Record_Count != Record_Count + (value X returned in step 4)) THEN
```

Must be changed to read:

```
# IF ( Total_Record_Count != Record_Count + (value X returned in step 4) - 1) THEN
```

Response:

[By BIG-EU TGTC or by BTL-WG]

[BTL-WG] The BTL-WG agrees and the test shall be corrected.



7.3.2.23.9 Total_Record_Count Test

~~Reason for Change: It is not clear whether or not log records should be added to a Trend Log when Record_Count is set to 0 and the Trend Log is disabled. As such the test steps were re-ordered to ignore this issue. This test is not in any SSPC proposal.~~

Reason For Change: The Total_Record_Count did not take into account the buffer-purged record that is added when the Record_Count is set to 0. This test is not in any SSPC proposal.

Dependencies: ReadProperty Service Execution Tests, 9.15; WriteProperty Service Execution Tests, 9.18.

BACnet Reference Clause: 12.23.16.

Purpose: To verify that the Total_Record_Count property increments for each record added to the Log_Buffer, even after Buffer_Size records have been added. (Note: it is not reasonable to test for the requirement of BACnet 12.23.16 that the value wrap from $2^{32}-1$ to 0; even if a record was collected every 100th of a second it could take more than 497 days to complete the test.)

Test Concept: The Trend Log is configured to acquire data by whatever means. Record_Count is set to zero and Total_Record_Count is read. Collection of data proceeds until Record_Count changes, collection is halted and Total_Record_Count is checked that it has incremented by Record_Count. If, for whatever reason, the IUT cannot be configured such that the TD is able to halt collection before Buffer_Size records are collected this test shall not be performed.

Configuration Requirements: Start_Time, if present, shall be configured with a date and time preceding the beginning of the test. Stop_Time, if present shall be configured with the latest possible date and time, in order that it occur after the end of the test. Log_Enable shall be set to FALSE.

Test Steps:

1. WRITE Record_Count = 0
2. WAIT **Internal Processing Fail Time**
- ~~3. WRITE Log_Enable = TRUE~~
4. TRANSMIT ReadProperty-Request,
 - 'Object Identifier' = (the object being tested),
 - 'Property Identifier' = Total_Record_Count
- ~~5. RECEIVE ReadProperty-ACK,~~
 - 'Object Identifier' = (the object being tested),
 - 'Property Identifier' = Total_Record_Count,
 - 'Property Value' = (any valid value, X)
- ~~35. WRITE Log_Enable = TRUE~~
6. WHILE (Record_Count = 2) DO { }
7. WRITE Log_Enable = FALSE
8. WAIT **Internal Processing Fail Time**
9. IF (Total_Record_Count != Record_Count-1+(value X returned in step 4)) THEN
ERROR "Total_Record_Count has incorrect value."

7.2.23.10 Notification_Threshold Test

Reason For Change: The calculation for the Total_Record_Count and other record count properties were incorrect. The Event_Values description was also incorrect. This test is not in any SSPC proposal.

Dependencies: ReadProperty Service Execution Tests, 9.18; WriteProperty Service Execution Tests, 9.22.

BACnet Reference Clause: 12.23.17.

Purpose: To verify that the Notification_Threshold property reflects the number of records collected since a previous notification, or since logging started, that causes a Buffer_Ready notification to be sent.



Test Concept: The Trend Log is configured to acquire data by whatever means. Record_Count is set to zero. Collection of data proceeds until a notification is seen, collection is halted and the value of Record_Count is checked. Collection resumes until the second notification, when collection is again halted and Record_Count verified. If, for whatever reason, the IUT cannot be configured such that the TD is able to halt collection before another record is collected after issuing the notification this test shall not be performed.

Configuration Requirements: Start_Time, if present, shall be configured with a date and time preceding the beginning of the test. Stop_Time, if present shall be configured with the latest possible date and time, in order that it occurs after the end of the test. Log_Enable shall be set to FALSE.

Test Steps:

1. WRITE Record_Count = 0
2. WAIT **Internal Processing Fail Time**
3. TRANSMIT ReadProperty-Request,
 - 'Object Identifier' = (the object being tested),
 - 'Property Identifier' = Total_Record_Count
4. RECEIVE ReadProperty-ACK,
 - 'Object Identifier' = (the object being tested),
 - 'Property Identifier' = Total_Record_Count
 - 'Property Value' = (any valid value, X)
5. WRITE Log_Enable = TRUE
6. MAKE (Trend Log object collect number of records *so as to cause a notification to go out*)
7. RECEIVE ConfirmedEventNotification-Request,
 - 'Process Identifier' = (any valid process ID),
 - 'Initiating Device Identifier' = IUT,
 - 'Event Object Identifier' = (the Trend Log object being tested),
 - 'Time Stamp' = (any appropriate BACnetTimeStamp value),
 - 'Notification Class' = (the configured notification class),
 - 'Priority' = (the value configured to correspond to a TO-NORMAL transition),
 - 'Event Type' = BUFFER_READY,
 - 'Notify Type' = EVENT | ALARM,
 - 'AckRequired' = TRUE | FALSE,
 - 'From State' = NORMAL,
 - 'To State' = NORMAL,
 - 'Event Values' = (BACnetObjectIdentifier of the IUT's Device object),
(BACnetObjectIdentifier of the Trend Log object),
(~~any BACnetDateTime~~)(any valid value < X),
(~~current local BACnetDateTime~~)(any value Y_1 where $Y_1 \leq$
 $X + Notification_Threshold$)
8. TRANSMIT BACnet-SimpleACK-PDU
9. VERIFY ($Total_Record_Count \geq Y_1$)
9. WRITE Log_Enable = FALSE
10. IF ($Total_Record_Count$ (value read in step 4) \neq Notification_Threshold) THEN
~~ERROR "Notification_Threshold value is incorrect."~~
11. WRITE Log_Enable = TRUE
- 12.10. MAKE (Trend Log object collect ~~number of~~ records *until* $Total_Record_Count == Y_1 +$ ~~specified by~~ Notification_Threshold)
- 13.11. RECEIVE ConfirmedEventNotification-Request,
 - 'Process Identifier' = (any valid process ID),
 - 'Initiating Device Identifier' = IUT,
 - 'Event Object Identifier' = (the Trend Log object being tested),
 - 'Time Stamp' = (any appropriate BACnetTimeStamp value),
 - 'Notification Class' = (the configured notification class),

'Priority' = (the value configured to correspond to a TO-NORMAL transition),
 'Event Type' = BUFFER_READY,
 'Notify Type' = EVENT | ALARM,
 'AckRequired' = TRUE | FALSE,
 'From State' = NORMAL,
 'To State' = NORMAL,
 'Event Values' = (BACnetObjectIdentifier of the IUT's Device object),
 (BACnetObjectIdentifier of the Trend Log object),
~~(BACnetDateTime sent in step 7)(Y₁),~~
~~(current local BACnetDateTime)(Y₁ + Notification_Threshold)~~

~~14~~12. TRANSMIT BACnet-SimpleACK-PDU

13. VERIFY (Total_Record_Count >= Y₁ + Notification_Threshold)

~~15~~14. WRITE Log_Enable = FALSE

~~16~~. IF (Total_Record_Count — (value X returned in step 4) != 2 * Notification_Threshold) THEN
 — ERROR “Notification_Threshold value is incorrect.”

7.3.2.23.X2 Time_Change Test

Reason for Change: No test exists for this functionality. This test is not in any SSPC proposal.

Dependencies: ReadRange Service Execution Tests, 9.21; (TimeSynchronization Service Execution Tests, 9.30 or UTCTimeSynchronization Service Execution Tests, 9.31)

This test may be skipped if the device does not support the Local_Time property in the device object or there is no way to change the time in the device.

BACnet Reference Clause: 12.23.14.

Purpose: To verify proper logging of time-change events in the log buffer

Test Concept: Change the clock in the device and verify that a record is logged indicating the number of seconds that the clock changed by or zero if unknown.

Test Configuration: The Trend Log is configured to acquire data by whatever means available. The Log_Buffer should be cleared, such that the Record_Count is 0. Configure the logging such that the entire test may be run without the trend buffer overflowing.

Test Steps:

1. WRITE Log_Enable = FALSE
2. WRITE Record_Count = 0
3. VERIFY (Log_Buffer contains ~~no entries~~ 1 entry, and it is the buffer-purged event)
4. TRANSMIT ReadProperty-Request,
 'Object Identifier' = (device that contains Trend Log)
 'Property Identifier' = Local_Time
5. RECEIVE ReadProperty-Ack,
 'Object Identifier' = (device that contains Trend Log)
 'Property Identifier' = Local_Time
 'Property Value' = (**currentTime**)
6. WRITE Log_Enable = TRUE
7. MAKE the time change on the device by a reasonable amount (**deltaTime**) (change by one hour or more)
8. WRITE Log_Enable = FALSE
9. VERIFY (Record_Count => 4)
10. VERIFY (Log_Buffer contains a status entry of time-change)
11. IF time-change amount is not zero, THEN



VERIFY (time-change value \approx **deltaTime**)

12. VERIFY (TimeStamp on the time-change entry \approx (**currentTime** + **deltaTime**)