

Clarification Request

References: BTL Specified Tests-5.0.final , BTL Specified Tests-9.0.final, test 13.X1.1, also in 135.1-2009m-4_PPR2 13.X.1.1

Date of BTL-WG Response: May 9, 2012

Background:

BTL specified Tests 9.0.final, test 13.X1.1

```

17. REPEAT X = (BACnetObjectIdentifiers obtained from step 7) DO {
    IF (Object_List obtained in step 10 does not contain X)
        TRANSMIT CreateObject-Request
            'Object Identifier' = X
        RECEIVE CreateObject-ACK
            'Object Identifier' = X

        TRANSMIT ReadProperty-Request
            'Object Identifier' = X
            'Property Identifier' = File_Size
        RECEIVE ReadProperty-ACK
            'Object Identifier' = X
            'Property Identifier' = File_Size
            'Property Value' = (any value)
        IF (File_Size is not equal to the size of the backed up file)
            TRANSMIT WriteProperty-Request
                'Object Identifier' = X
                'Property Identifier' = File_Size
                'Property Value' = 0
            RECEIVE BACnet-Simple-ACK-PDU
        IF (Y = RECORD_ACCESS)
            TRANSMIT AtomicWriteFile-Request
                'File Identifier' = X
                'File Start Record' = 0
                'Record Data' = (file content for first record obtained in step 6)
            RECEIVE AtomicWriteFile-ACK
                'File Start Record' = 0
        FOR Z = (number of records obtained in step 6 - 1) {
            TRANSMIT AtomicWriteFile-Request
                'File Identifier' = X
                'File Start Record' = -1
                'Record Count' = 1
                'Record Data' = (file contents for this record obtained in step 6)
            RECEIVE AtomicWriteFile-ACK

```

```

        'File Start Record' = Z - 1 + 1
    }
ELSE
    WHILE (There is still octets to be written for this file) DO {
        TRANSMIT AtomicWriteFile-Request
        'File Identifier' = X
        'File Start Position' = (The next unwritten octet, starting at 0)
        'Record Data' = (file contents obtained in step 6, the number of octets
                        being equal to the minimum of the number of octets
                        determined prior to the start of the test)
        RECEIVE AtomicWriteFile-ACK
        'File Start Position' = (The next unwritten octet, starting at 0)
    }
19. TRANSMIT ReinitializeDevice-Request,

```

In step 17 in the AtomicWriteFile Ack for the Record Access Case always a -1 is expected. This seems to be wrong. The position of the written record should be expected.

Question:

Are the above changes correct?

Response:

No. The test is incorrect and needs to be changed as below. Note: adding the terms "1 to" in the FOR step makes the test clearer.

```

FOR Z = (1 to number of records obtained in step 6 - 1) {
    TRANSMIT AtomicWriteFile-Request
    'File Identifier' = X
    'File Start Record' = -1
    'Record Count' = 1
    'Record Data' = (file contents for this record obtained in step 6)
    RECEIVE AtomicWriteFile-ACK
    'File Start Record' = Z + 1
}

```