INTERPRETATION IC 135-2004-16 OF
ANSI/ASHRAE STANDARD 135-2004 BACnet® -
A Data Communication Protocol for Building
Automation and Control Networks

Approval Date: June 24, 2006

Request from: David Hudson (dhudson@deltacontrols.com) Delta Controls, 17850 56th Avenue, Surrey, BC V3S-1C7.

Reference: This request for interpretation refers to the requirements presented in ANSI/ASHRAE Standard 135-2004, Sections 12.1.8, 12.2.8, 12.3.8, 12.4.7, 12.6.8, 12.7.8, 12.8.7, 12.17.7, 12.18.8, 12.20.7, and 12.23.8 related to Event_State and Relibility properties.

Background: There appear to be two different relationships between Event_State and Reliability when intrinsic reporting is not supported in an object, depending on the object type.

For Analog Input, Analog Output, Analog Value, Brinary Input, Binary Output, Binary Value, and Loop object types, Event_State (example, Section 12.2.8) is defined defined as:
The Event_State property, of type BACnetEventState, is included in order to provide a way to determine if this object has an active event state associated with it. If the object supports intrinsic reporting, then the Event_State property shall indicate the event state of the object. If the object does not support intrinsic reporting, then the value of this property shall be NORMAL. If the Reliability property is present and does not have a value of NO_FAULT_DETECTED, then the value of the Event_State property shall be FAULT. Changes in the Event_State property to the value FAULT are considered to be "fault" events.

The sentence "If the Reliability property is present and does not have a value of NO_FAULT_DETECTED, then the value of the Event_State property shall be FAULT" contradicts the sentence "If the object does not support intrinsic reporting, then the value of this property shall be NORMAL."

The Accumulator, Mulistate Input, Multistate Value and Pulse Convertor object types, all indicate that if the object does not support intrinsic reporting, and if the Reliability property is present and does not have a value of NO_FAULT_DETECTED, then the value of Event State shall be FAULT.

Interpretation: Our interpretation is there are two different relationships between Event_State and Reliability when intrinsic reporting is not supported in an object. Namely:
For Analog Input, Analog Output, Analog Value, Binary Input, Binary Output, Binary Value and Loop object types: if an object does not support intrinsic reporting then the value of Event_State will always be NORMAL regardless of the value of Reliability if Reliability is present.
And, for Accumulator, Multistate Input, Multistate Value, and Pulse Converter object types: if an object does not support intrinsic reporting, and if the Reliability property is present and does not have a value of NO_FAULT_DETECTED, then the of Event_State will be FAULT.

Question: Is this interpretation correct?

Answer: No.

Comments: The committee agrees that the language across objects is different and that the standard should be revised to use the same language. Until the language in the standard has been unified, for those object types where the behavior of Event_State is inconsistently mandated, either interpretation for Event_State behavior shall be acceptable.