CONTENTS

CLAUSE **PAGE**

FOREWORD 1
1. PURPOSE 3
2. SCOPE 3
3. DEFINITIONS 3
3.1 Terms Adopted from International Standards 3
3.2 Abbreviations and Acronyms Used in the Standard 3
3.3 Common language used in tests 3
4. ELECTRONIC PICS FILE FORMAT 4
4.1 Character Encoding 4
4.2 Structure of EPICS Files 5
4.3 Character Strings 5
4.4 Notational Rules for Parameter Values 5
4.5 Sections of the EPICS File 7
5. EPICS CONSISTENCY TESTS 12
6. CONVENTIONS FOR SPECIFYING BACnet CONFORMANCE TESTS 14
6.1 TCSL Components 14
6.2 TCSL Statements 15
6.3 Time Dependencies 20
6.4 BACnet References 22
6.5 TD Requirements 22
6.6 Test Execution Considerations 22
7. OBJECT SUPPORT TESTS 23
7.1 Read Support for Properties in the Test Database 23
7.2 Write Support for Properties in the Test Database 25
7.3 Object Functionality Tests 32
8. APPLICATION SERVICE INITIATION TESTS 384
8.1 AcknowledgeAlarm Service Initiation Tests 384
8.2 ConfirmedCOVNotification Service Initiation Tests 386
8.3 UnconfirmedCOVNotification Service Initiation Tests 402
8.4 ConfirmedEventNotification Service Initiation Tests 407
8.5 UnconfirmedEventNotification Service Initiation Tests 458
8.6 GetAlarmSummary Service Initiation Tests 483
8.7 GetEnrollmentSummary Service Initiation Tests 484
8.8 GetEventInformation Service Initiation Tests 485
8.9 LifeSafetyOperation Service Initiation Tests 487
8.10 SubscribeCOV Service Initiation Tests 488
8.11 SubscribeCOVProperty Service Initiation Tests 488
8.12 AtomicReadFile Service Initiation Tests 494
8.13 AtomicWriteFile Service Initiation Tests 494
8.14 AddListElement Service Initiation Tests 495
8.15 RemoveListElement Service Initiation Tests 495
8.16 CreateObject Service Initiation Tests 496
8.17 DeleteObject Service Initiation Tests 497
8.18 ReadProperty Service Initiation Tests 497
8.19 ReadPropertyConditional Service Initiation Tests 499
8.20 ReadPropertyMultiple Service Initiation Tests 500
8.21 ReadRange Service Initiation Tests 502
8.22 WriteProperty Service Initiation Tests 506
8.23 WritePropertyMultiple Service Initiation Tests 509
8.24 DeviceCommunicationControl Service Initiation Tests 511
8.25 ConfirmedPrivateTransfer Service Initiation Test 513
8.26 UnconfirmedPrivateTransfer Service Initiation Test 513
8.27 ReinitializeDevice Service Initiation Tests 513

Preview

8.28 ConfirmedTextMessage Service Initiation Tests 514
8.29 UnconfirmedTextMessage Service Initiation Tests 515
8.30 TimeSynchronization Service Initiation Tests 516
8.31 UTCTimeSynchronization Service Initiation Tests 516
8.32 Who-Has Service Initiation Tests 516
8.33 I-Have Service Initiation Tests 518
8.34 Who-Is Service Initiation Tests 518
8.35 I-Am Service Initiation Tests 518
8.36 VT-Open Service Initiation Tests 518
8.37 VT-Close Service Initiation Tests 519
8.38 VT-Data Service Initiation Tests 521
8.39 RequestKey Service Initiation Tests 522
8.40 Authenticate Service Initiation Tests 523
8.41 WriteGroup Service Initiation Tests 526
8.42 SubscribeCOVPropertyMultiple Service Initiation Tests 526
8.43 AuditLogQuery Initiation Tests 531
8.44 Who-Am-I Service Initiation Tests 531
8.45 You-Are Service Initiation Tests 532
9. APPLICATION SERVICE EXECUTION TESTS 534
9.1 AcknowledgeAlarm Service Execution Tests 534
9.2 ConfirmedCOVNotification Service Execution Tests 560
9.3 UnconfirmedCOVNotification Service Execution Tests 566
9.4 ConfirmedEventNotification Service Execution Tests 569
9.5 UnconfirmedEventNotification Service Execution Tests 573
9.6 GetAlarmSummary Service Execution Tests 575
9.7 GetEnrollmentSummary Service Execution Tests 576
9.8 GetEventInformation Service Execution Tests 579
9.9 LifeSafetyOperation Service Execution Test 581
9.10 SubscribeCOV Service Execution Tests 586
9.11 SubscribeCOVProperty Service Execution Tests 596
9.12 AtomicReadFile Service Execution Tests 608
9.13 AtomicWriteFile Service Execution Tests 614
9.14 AddListElement Service Execution Tests 624
9.15 RemoveListElement Service Execution Tests 627
9.16 CreateObject Service Execution Tests 628
9.17 DeleteObject Service Execution Tests 634
9.18 ReadProperty Service Execution Tests 635
9.19 ReadPropertyConditional Service Execution Tests 639
9.20 ReadPropertyMultiple Service Execution Tests 640
9.21 ReadRange Service Execution Tests 650
9.22 WriteProperty Service Execution Tests 663
9.23 WritePropertyMultiple Service Execution Tests 671
9.24 DeviceCommunicationControl Service Execution Test 687
9.25 ConfirmedPrivateTransfer Service Execution Tests 695
9.26 UnconfirmedPrivateTransfer Service Execution Tests 696
9.27 ReinitializeDevice Service Execution Tests 696
9.28 ConfirmedTextMessage Service Execution Tests 700
9.29 UnconfirmedTextMessage Service Execution Tests 701
9.30 TimeSynchronization Service Execution Tests 702
9.31 UTCTimeSynchronization Service Execution Tests 703
9.32 Who-Has Service Execution Tests 703
9.33 Who-Is Service Execution Tests 711
9.34 VT-Open Service Execution Tests 714
9.35 VT-Close Service Execution Tests 715
9.36 VT-Data Service Execution Tests 717
9.37 RequestKey Service Execution Test 717

Preview

9.38 Authenticate Service Execution Tests 719
9.39 General Testing of Service Execution 723
9.40 AuditLogQuery Service Execution Tests 723
9.41 WriteGroup Tests 725
9.42 SubscribeCOVPropertyMultiple Service Execution Tests 729
9.43 Who-Am-I Service Execution Tests 744
9.44 You-Are Service Execution Tests 746
10. NETWORK LAYER PROTOCOL TESTS 753
10.1 General Network Layer Tests 753
10.2 Router Functionality Tests 754
10.3 Half-Router Functionality Tests 783
10.4 B/IP PAD Tests 789
10.5 Initiating Network Layer Messages 791
10.6 Non-Router Functionality Tests 793
10.7 Route Binding Tests 795
10.8 Virtual Routing Functionality Tests 800
11. LOGICAL LINK LAYER PROTOCOL TESTS 821
11.1 UI Command and Response 821
11.2 XID Command and Response 821
11.3 TEST Command and Response 822
12. DATA LINK LAYER PROTOCOLS TESTS 823
12.1 MS/TP State Machine Tests 823
12.2 PTP State Machine Tests 886
12.3 BACnet/IP Functionality Tests 920
12.4 BACnet/IPv6 Functionality Tests 954
12.5 Secure Connect Functionality Tests 969
13. SPECIAL FUNCTIONALITY TESTS 1033
13.1 Segmentation 1033
13.2 Time Manager 1042
13.3 Character Sets 1046
13.4 Malformed PDUs 1047
13.5 Subordinate Proxy Tests 1048
13.6 Automatic Network Mapping 1050
13.7 Automatic Device Mapping 1051
13.8 Backup and Restore Procedure Tests 1051
13.9 Application State Machine Tests 1066
13.10 Workstation Scheduling Tests 1066
13.11 BACnet/SC Certificate Replacement Tests 1084
14. Reporting Test Results 1092
ANNEX A – EXAMPLE EPICS (INFORMATIVE) 1093
HISTORY OF REVISIONS 1113

Preview

Preview

(This foreword is not part of this standard. It is merely informative and does not contain requirements necessary for conformance to the standard. It has not been processed according to the ANSI requirements for a standard and may contain material that has not been subject to public review or a consensus process. Unresolved objectors on informative material are not offered the right to appeal at ASHRAE or ANSI.)

FOREWORD

ASHRAE Standard 135 (BACnet), a building automation and control networking protocol, is designed specifically to meet the communication needs of building automation and control systems for applications such as heating, ventilation, and air-conditioning control; lighting control; access control; elevators; and fire detection systems. The motivation behind BACnet is to provide an interoperable protocol allowing equipment from different vendors to integrate into a coherent automation and control system.

The motivation for this Standard is to provide the procedures and tools necessary to validate the interoperability of equipment claiming conformance to ASHRAE Standard 135 (BACnet). This standard defines the tools to allow a vendor to define the equipment to be tested, the language that is the grammar for the test descriptions and finally the test descriptions themselves.

As BACnet continues to improve and evolve with the changes in technology and building automation, so shall this standard.

With that in mind, the bulk of the changes, in this version of the standard, relate to the addition of the BACnet/IPv6 and BACnet/SC datalink layer test descriptions.

This standard defines the format of an electronic document called Electronic Protocol Implementation Conformance Statement (EPICS) that allows a vendor to provide a detailed accounting of the capabilities of the equipment to be tested. This electronic document is used by software testing tools to conduct and interpret the results of tests defined in this standard.

This standard also includes a scripting language called Testing and Conformance Scripting Language (TCSL) that provides the grammar to make the test descriptions in the standard clear and concise.

The final and largest sections in this standard are the test descriptions that provide the procedures to validate equipment main change This version of the standard includes tests related to alarm and event reporting.

This standard continues to evolve as BACnet evolves.

This standard defines the electronic document format (EPICS) that allows software testing tools documents the capabilities of the equipment by providing the necessary procedures to validate an implementation of BACnet.

This motivation for this Standard is to support ASHRAE Standard 135 by providing the necessary procedures to validate an implementation of BACnet.

BACnet, the ASHRAE building automation and control networking protocol, has been designed specifically to meet the communication needs of building automation and control systems for applications such as heating, ventilating, and air-conditioning control, lighting control, access control, elevators, and fire detection systems. The BACnet protocol provides mechanisms by which computerized equipment of arbitrary function may exchange information, regardless of the particular building service it performs. As a result, the BACnet protocol may be used by head-end computers, general-purpose direct digital controllers, and application specific or unitary controllers with equal effect.

This motivation for this Standard is to provide the necessary infrastructure to ensure the widespread desire of building owners and operators for "interoperability," the ability to integrate equipment from different vendors into a coherent automation and control system - and to do so competitively. To accomplish this, the Standard Project Committee (SPC) solicited and received input from dozens of interested firms and individuals; reviewed all relevant national and international data communications standards, whether de facto or the result of committee activity; and spent countless hours in debate and discussion of the pros and cons of each element of the protocol.

What has emerged from the committee deliberations is a network protocol model with these principal characteristics:

Preview

a. All network devices (except MS/TP Subordinate Nodes) are peers, but certain peers may have greater privileges and responsibilities than others.

b. Each network device is modeled as a collection of network-accessible, named entities called "objects." Each object is characterized by a set of attributes or "properties." While this Standard prescribes the most widely applicable object types and their properties, implementors are free to create additional object types if desired. Because the object model can be easily extended, it provides a way for BACnet to evolve in a backward compatible manner as the technology and building needs change.

c. Communication is accomplished by reading and writing the properties of particular objects and by the mutually acceptable execution of other protocol "services." While this Standard prescribes a comprehensive set of services, mechanisms are also provided for implementors to create additional services if desired.

d. Because of this Standard's adherence to the ISO concept of a "layered" communication architecture, the same messages may be exchanged using various network access methods and physical media. This means that BACnet networks may be configured to meet a range of speed and throughput requirements with commensurately varying cost. Multiple BACnet networks can be interconnected within the same system forming an internetwork of arbitrarily large size. This flexibility also provides a way for BACnet to embrace new networking technologies as they are developed.

BACnet was designed to gracefully improve and evolve as both computer technology and demands of building automation systems change. Upon its original publication in 1995, a Standing Standards Project Committee was formed to deliberate enhancements to the protocol under ASHRAE rules for "continuous maintenance." Much has happened since the BACnet standard was first promulgated. BACnet has been translated into Chinese, Japanese, and Korean, and embraced across the globe. BACnet devices have been designed, built and deployed on all seven continents. Suggestions for enhancements and improvements have been continually received, deliberated, and, ultimately, subjected to the same consensus process that produced the original standard. This publication is the result of those deliberations and brings together all of the corrections, refinements, and improvements that have been adopted.

Among the features that have been added to BACnet are: increased capabilities to interconnect systems across wide area networks using Internet Protocols, new objects and services to support fire detection, other life safety applications, lighting, physical access control, and elevator monitoring, capabilities to backup and restore devices, standard ways to collect trend data, new tools to make specifying BACnet systems easier, a mechanism for making interoperable extensions to the standard visible, and many others. The successful addition of these features demonstrates that the concept of a protocol deliberately crafted to permit extension of its capabilities over time as technology and needs change is viable and sound.

All communication protocols are, in the end, a collection of arbitrary solutions to the problems of information exchange and all are subject to change as time and technology advance. BACnet is no exception. Still, it is the hope of those who have contributed their time, energies, and talents to this work that BACnet will help to fulfill, in the area of building automation and control, the promise of the information age for the public good!

Preview

1. PURPOSE

To define a standard method for verifying that an implementation of the BACnet protocol provides each capability claimed in its Protocol Implementation Conformance Statement (PICS) in conformance with the BACnet standard.

2. SCOPE

This standard provides a comprehensive set of procedures for verifying the correct implementation of each capability claimed on a BACnet PICS including:

(a) support of each claimed BACnet service, either as an initiator, executor, or both,
(b) support of each claimed BACnet object-type, including both required properties and each claimed optional property,
(c) support of the BACnet network layer protocol,
(d) support of each claimed data link option, and
(e) support of all claimed special functionality.

3. DEFINITIONS

All definitions from ANSI/ASHRAE Standard 135-2020 also apply to this addendum.

3.1 Terms Adopted from International Standards

local network: the network to which a BACnet device is directly connected.

remote network: a network that is accessible from a BACnet device only by passing through one or more routers.

test database: a database of BACnet functionality and objects created by reading the contents of an EPICS.

3.2 Abbreviations and Acronyms Used in the Standard

BNF Backus-Naur Form syntax
EPICS electronic protocol implementation conformance statement
IUT implementation under test
TCSL testing and conformance scripting language
TD testing device
TPI text protocol information

3.3 Common language used in tests

'**any valid value**': Any valid value refers to any value of the correct data type and within the vendor's range specified for the property this is applied to.

'**any appropriate password**': Any password that meets the Configuration Requirements specified in the test or test section. Passwords when required by the vendor are required to be no more than 20 characters.

'**reset**': Some tests require to reset the IUT. Reset includes power cycle via switch, power cycle via loss of power, and reinitializeDevice WARMSTART. As defined by the BACnet standard, "WARMSTART shall mean to reboot the device and start over, retaining all data and programs that would normally be retained during a brief power outage."

Preview