SLOVENSKI STANDARD
SIST EN ISO 16484-5:2008/A1:2009

01-oktober-2009

Avtomatizacija stavb in sistemi za regulacijo - 5. del: Protokol izmenjave podatkov - Dodatek (ISO 16484-5:2008/Amd 1:2009)

Building automation and control systems - Part 5: Data communication protocol - Amendment 1 (ISO 16484-5:2008/Amd 1:2009)

Systeme der Gebäudeautomation - Teil 5: Datenkommunikationsprotokoll - Änderung 1 (ISO 16484-5:2008/Amd 1:2009)

Systèmes d'automatisation et de gestion technique du bâtiment - Partie 5: Protocole de communication de données - Amendement 1 (ISO 16484-5:2008/Amd 1:2009)

Ta slovenski standard je istoveten z: **EN ISO 16484-5:2008/A1:2009**

ICS:

35.240.99	Uporabniške rešitve IT na drugih področjih	IT applications in other fields
97.120	Avtomatske krmilne naprave za dom	Automatic controls for household use

SIST EN ISO 16484-5:2008/A1:2009 **en,fr,de**

EUROPEAN STANDARD

NORME EUROPÉENNE

EUROPÄISCHE NORM

EN ISO 16484-5:2008/A1

May 2009

ICS 35.240.99; 91.040.01

English Version

Building automation and control systems - Part 5: Data communication protocol - Amendment 1 (ISO 16484-5:2008/Amd 1:2009)

Systèmes d'automatisation et de gestion technique du bâtiment - Partie 5: Protocole de communication de données - Amendement 1 (ISO 16484-5:2008/Amd 1:2009)

Systeme der Gebäudeautomation - Teil 5: Datenkommunikationsprotokoll - Änderung 1 (ISO 16484-5:2008/Amd 1:2009)

This amendment A1 modifies the European Standard EN ISO 16484-5:2008; it was approved by CEN on 20 April 2009.

CEN members are bound to comply with the CEN/CENELEC Internal Regulations which stipulate the conditions for inclusion of this amendment into the relevant national standard without any alteration. Up-to-date lists and bibliographical references concerning such national standards may be obtained on application to the CEN Management Centre or to any CEN member.

This amendment exists in three official versions (English, French, German). A version in any other language made by translation under the responsibility of a CEN member into its own language and notified to the CEN Management Centre has the same status as the official versions.

CEN members are the national standards bodies of Austria, Belgium, Bulgaria, Cyprus, Czech Republic, Denmark, Estonia, Finland, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Lithuania, Luxembourg, Malta, Netherlands, Norway, Poland, Portugal, Romania, Slovakia, Slovenia, Spain, Sweden, Switzerland and United Kingdom.

EUROPEAN COMMITTEE FOR STANDARDIZATION
COMITÉ EUROPÉEN DE NORMALISATION
EUROPÄISCHES KOMITEE FÜR NORMUNG

Management Centre: Avenue Marnix 17, B-1000 Brussels

Contents

Page

Foreword

This document (EN ISO 16484-5:2008/A1:2009) has been prepared by Technical Committee ISO/TC 205 "Building environment design" in collaboration with Technical Committee CEN/TC 247 “Building Automation, Controls and Building Management” the secretariat of which is held by SNV.

This European Standard shall be given the status of a national standard, either by publication of an identical text or by endorsement, at the latest by November 2009, and conflicting national standards shall be withdrawn at the latest by November 2009.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. CEN [and/or CENELEC] shall not be held responsible for identifying any or all such patent rights.

According to the CEN/CENELEC Internal Regulations, the national standards organizations of the following countries are bound to implement this European Standard: Austria, Belgium, Bulgaria, Cyprus, Czech Republic, Denmark, Estonia, Finland, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Lithuania, Luxembourg, Malta, Netherlands, Norway, Poland, Portugal, Romania, Slovakia, Slovenia, Spain, Sweden, Switzerland and the United Kingdom.

Endorsement notice

The text of ISO 16484-5:2008/Amd 1:2009 has been approved by CEN as a EN ISO 16484-5:2008/A1:2009 without any modification.

INTERNATIONAL STANDARD

ISO 16484-5

Second edition
2007-03-15

AMENDMENT 1
2009-05-15

Building automation and control systems —

Part 5: Data communication protocol

AMENDMENT 1

Systèmes d'automatisation et de gestion technique du bâtiment —

Partie 5: Protocole de communication de données

AMENDEMENT 1

Reference number
ISO 16484-5:2007/Amd.1:2009(E)

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Published in Switzerland

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of technical committees is to prepare International Standards. Draft International Standards adopted by the technical committees are circulated to the member bodies for voting. Publication as an International Standard requires approval by at least 75 % of the member bodies casting a vote.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO shall not be held responsible for identifying any or all such patent rights.

Amendment 1 to ISO 16484-5:2007 was prepared by Technical Committee ISO/TC 205, *Building environment design*.

Introduction

The purpose of this Addendum is to add a number of independent substantive changes to the BACnet standard. The changes are summarized below.

135-2004*a*-1, p. 1: Revise Life Safety Point and Life Safety Zone object types to modify their behaviour when placed out of service.

135-2004*c*-1, p. 4: Add BACnet/WS Web Services Interface.

135-2004*d*-1, p. 39: Add a new Structured View object type.

135-2004*d*-2, p. 45: Allow acknowledgement of unseen TO-OFFNORMAL event notifications.

135-2004*d*-3, p. 46: Relax the Private Transfer and Text Message BIBB requirements.

135-2004*d*-4, p. 47: Exclude LIFE_SAFETY and BUFFER_READY notifications from the Alarm Notifications BIBBs.

135-2004*d*-5, p. 49: Establish the minimum requirements for a BACnet device with an application layer.

135-2004*d*-6, p. 51: Remove the requirement for the DM-DOB-A BIBB from the B-OWS and B-BC device profiles.

135-2004*d*-7, p. 52: Relax mandated values for APDU timeouts and retries when configurable, and change default values.

135-2004*d*-8, p. 53: Fix EventCount handling error in MS/TP Master Node State Machine.

135-2004*d*-9, p. 54: Permit routers to use a local network number in Device_Address_Binding.

135-2004*d*-10, p. 55: Identify conditionally writable properties.

135-2004*d*-11, p. 56: Specify Error returns for the AcknowledgeAlarm service.

135-2004*e*-1, p. 58: Add a new Load Control object type.

135-2004*f*-1, p. 71: Add new Access Door object type.

In this Amendment, text being added to existing clauses of ISO 16484-5 is indicated through the use of *italics*, while deletions are indicated by ~~strikethrough~~. Plain type is used throughout where entirely new subclauses are added.

Building automation and control systems —

Part 5:
Data communication protocol

AMENDMENT 1

135-2004*a*-1. Revise Life Safety Point and Life Safety Zone object types for out-of-service operation.

Addendum 135-2004*a*-1

[Change **Table 12-18**, p. 194]

Table 12-18. Properties of the Life Safety Point Object Type

Property Identifier	Property Datatype	Conformance Code
...	...	...
Present_Value	BACnetLifeSafetyState	~~R~~ ^{1}*R*
Tracking_Value	BACnetLifeSafetyState	~~O~~*R* ^{1}
...	...	...

[Change **12.15.4**, p. 195]

12.15.4 Present_Value

This property, of type BACnetLifeSafetyState, reflects the state of the Life Safety Point object. The means of deriving the Present_Value shall be a local matter. Present_Value may latch non-NORMAL state values until reset. ~~The Present_Value property shall be writable when Out_Of_Service is TRUE.~~

[Change **12.15.5**, p. 195]

12.15.5 Tracking_Value

This ~~optional~~ property, of type BACnetLifeSafetyState, reflects the non-latched state of the Life Safety Point object. The means of deriving the state shall be a local matter. Unlike Present_Value, which may latch non-NORMAL state values until reset, Tracking_Value shall continuously track changes in the state. *The Tracking_Value property shall be writable when Out_Of_Service is TRUE.*

[Change **12.15.11**, p. 196]

12.15.11 Out_Of_Service

The Out_Of_Service property, of type BOOLEAN, is an indication whether (TRUE) or not (FALSE) the input(s) or process the object represents is not in service. This means that changes to the ~~Present_Value~~ *Tracking_Value* property are decoupled from the input(s) or process when the value of Out_Of_Service is TRUE. In addition, the Reliability property and the corresponding state of the FAULT flag of the Status_Flags property shall be decoupled when Out_Of_Service is TRUE. While the Out_Of_Service property is TRUE, the ~~Present_Value~~ *Tracking_Value* and Reliability properties may be changed to any value as a means of simulating specific fixed conditions or for testing purposes. Other functions that depend on the state of the ~~Present_Value~~ *Tracking_Value* or Reliability properties shall respond to changes made to these properties while Out_Of_Service is TRUE, as if those changes had occurred to the input(s) or process.

[Change **Table 12-19**, p. 200]

Table 12-19. Properties of the Life Safety Zone Object Type

Property Identifier	Property Datatype	Conformance Code
...	...	...
Present_Value	BACnetLifeSafetyState	~~R~~[1] *R*
Tracking_Value	BACnetLifeSafetyState	~~O~~ *R*[1]
...	...	...

[Change **12.16.4**, p. 201]

12.16.4 Present_Value

This property, of type BACnetLifeSafetyState, reflects the state of the Life Safety Zone object. The means of deriving the Present_Value shall be a local matter. Present_Value may latch non-NORMAL state values until reset. ~~The Present_Value property shall be writable when Out_Of_Service is TRUE.~~

[Change **12.16.5**, p. 201]

12.16.5 Tracking_Value

This ~~optional~~ property, of type BACnetLifeSafetyState, reflects the non-latched state of the Life Safety Zone object. The means of deriving the state shall be a local matter. Unlike Present_Value, which may latch non-NORMAL state values until reset, Tracking_Value shall continuously track changes in the state. *The Tracking_Value property shall be writable when Out_Of_Service is TRUE.*

[Change **12.16.11**, p. 202]

12.16.11 Out_Of_Service

The Out_Of_Service property, of type BOOLEAN, is an indication whether (TRUE) or not (FALSE) the input(s) or process the object represents is not in service. This means that changes to the ~~Present_Value~~ *Tracking_Value* property are decoupled from the input(s) or process when the value of Out_Of_Service is TRUE. In addition, the Reliability property and the corresponding state of the FAULT flag of the Status_Flags property shall be decoupled when Out_Of_Service is TRUE. While the Out_Of_Service property is TRUE, the ~~Present_Value~~ *Tracking_Value* and Reliability properties may be changed to any value as a means of simulating specific fixed conditions or for testing purposes. Other functions that depend on the state of the ~~Present_Value~~ *Tracking_Value* or Reliability properties shall respond to changes made to these properties while Out_Of_Service is TRUE, as if those changes had occurred to the input(s) or process.

[Change **Annex C**, p.459]

LIFE-SAFETY-POINT :: = SEQUENCE {

…		
present-value	[85]	BACnetLifeSafetyState,
tracking-value	[164]	BACnetLifeSafetyState ~~OPTIONAL~~,
description	[28]	CharacterString OPTIONAL,
…		
}		

[Change **Annex C**, p.460]

LIFE-SAFETY-ZONE :: = SEQUENCE {

…		
present-value	[85]	BACnetLifeSafetyState,
tracking-value	[164]	BACnetLifeSafetyState ~~OPTIONAL~~,
description	[28]	CharacterString OPTIONAL,
…		
}		

135-2004*c*-1. Adding BACnet/WS Web Services Interface

Rationale
"Web services" is emerging as the predominant technology for the integration of a wide variety of enterprise information. This addendum defines a standard means of using Web services to integrate facility data from disparate data sources, including BACnet networks, with a variety of business enterprise applications.

Addendum 135-2004*c*-1

[Add new **Annex N**]

ANNEX N - BACnet/WS WEB SERVICES INTERFACE (NORMATIVE)

(This annex is part of this standard and is required for its use.)

This annex defines a data model and Web service interface for integrating facility data from disparate data sources with a variety of business management applications. The data model and access services are generic and can be used to model and access data from any source, whether the server owns the data locally or is acting as a gateway to other standard or proprietary protocols.

Implementations of the services described in this standard shall conform to the Web Services Interoperability Organization (WS-I) Basic Profile 1.0, which specifies the use of Simple Object Access Protocol (SOAP) 1.1 over Hypertext Transfer Protocol -- HTTP/1.1 (RFC2616) and encodes the data for transport using Extensible Markup Language (XML) 1.0 (Second Edition), which uses the datatypes and the lexical and canonical representations defined by the World Wide Web Consortium XML Schema.

Clients may determine the version of the BACnet/WS standard that a server implements by querying a specific numerical value as defined in clause N.9. The numerical value for the version described in this document is 1.

There are three distinct usages of datatype names in this standard. Datatype names beginning with a lowercase letter, such as "string", and "nonNegativeInteger", refer to datatypes defined by the XML Schema standard. Datatype names beginning with an uppercase letter, such as "Real" or "Multistate" refer to the value types defined in Clause N.8.9. Datatype names used in a "typical language binding signature" are arbitrary and are for illustrative purposes only.

N.1 Data Model

The data structures and methods used to store information internally in a BACnet/WS server are a local matter. However, in order to exchange that information using Web services, this standard establishes a minimal set of requirements for the structuring and association of data exchanged with a BACnet/WS server.

A node is the fundamental primitive data element in the BACnet/WS data model. Nodes are arranged into a hierarchy in the data model. The topmost node in the hierarchy is known as the root node. A root node has children, but no parent. Every other node has a single parent and may optionally have children. The network visible state of a node is exposed as a collection of attributes.

Any node may have a value. The possible types for a node's value are limited to the primitive datatypes "String", "OctetString", "Real", "Integer", "Multistate", "Boolean", "Date", "Time", "DateTime", and "Duration". Nodes that have a value may also have other attributes related to that value, such as minimum, writable, etc.

An attribute is a single aspect or quality of a node, such as its value or its writability. Every node exposes a collection of attributes. Some attributes are required for all nodes, and some are conditionally required based on the value of other attributes. Some of the attributes are localizable and may return different values based on an option in a service request. Attributes are described more fully in Clause N.8.

Attributes may themselves have attributes that define a single aspect or quality of the original attribute. This standard supports this recursion syntactically, but does not define or require that any of the standardized attributes have attributes themselves at this time. Servers may provide proprietary attributes for any node or attribute at any level in the hierarchy.

A path is a character string that is used to identify a node or an attribute of a node. The hierarchy of nodes is reflected in a path as a hierarchy of identifiers arranged as a delimited series, similar to the arrangement of identifiers in a Uniform Resource Locator (URL) for the World Wide Web. A path like "/East Wing/AHU #5/Discharge Temp" identifies a node, and a path like "/East Wing/AHU #5/Discharge Temp:InAlarm" identifies the InAlarm attribute of that node. Paths are described more fully in Clause N.2.

To allow for an arbitrary number of logical arrangements of nodes, a single node may logically appear to be in more than one place in the hierarchy through the use of a reference node. Reference nodes may be used to build alternate logical arrangements of nodes since the children of a reference node may differ from that of its referent node. Reference nodes are described more fully in Clause N.4.

The arrangement of data nodes into hierarchies and the naming of those nodes is generally a local matter. However, this standard also defines a number of standardized nodes with standardized names and locations that allow clients to obtain basic information about the server itself. These standardized nodes are described more fully in clause N.9.

N.2 Paths

A path is a character string that is used to identify a node or a specific attribute. The hierarchy of nodes is reflected in a path as a hierarchy of node identifiers arranged as a delimited series separated by forward slash ("/") characters. Similarly, the hierarchy of attributes is reflected in a path as a hierarchy of attribute identifiers arranged as a delimited series separated by colon (":") characters.

Certain services accept an optional attribute path on the end of a node path. If an attribute path is not specified to those services, the Value attribute is assumed. The attribute path is separated from the node path with a colon.

The concatenated path form is:

[/node-identifier[/node-identifier]…][:attribute-identifier[:attribute-identifier]…]

where square brackets indicate optionality and "…" indicates repetition of the previous element.

Examples: "/aaa" "/aaa/bbb" "/aaa/bbb/ccc:Description" "/aaa/bbb/ccc:Description:.foo"

All identifiers are case sensitive and shall be of non-zero length. Identifiers are not localizable and are not affected by the "locale" or "canonical" service options. A path with no node identifier ("") refers to the root of the hierarchy, and ":attribute-identifier" is the syntax for accessing the attributes of the root node.

Only printable characters may be used to construct path identifiers, and, as an additional restriction, all characters equivalent to the ANSI X3.4 "control characters" (those less than X'20') are not allowed, and neither are any characters equivalent to the following ANSI X3.4 characters: / \ : ; | < > * ? " [ ] { }

Node identifiers beginning with a period (".") character and attribute identifiers not beginning with a period (".") character are reserved for use by ASHRAE. This restriction separates node and attribute identifiers that are defined by this standard from those that are defined by the server, perhaps based on user input. Server defined node identifiers shall not start with a period, so that "/aaa/.first-floor" is invalid but "/aaa/first-floor" is valid. Conversely, all server defined attribute identifiers shall start with a period, so that "/aaa:MyNewAttribute" is invalid but "/aaa:.MyNewAttribute" is valid. This asymmetry is based on the expected common usage where most node identifiers will be server defined and most attributes are standard, making the use of periods the exception rather than the norm.

Space characters are allowed and are significant in identifiers; however, it is recommended that identifiers should not begin or end with space characters.

N.3 Normalized Points

Most building automation protocols, both standard and proprietary, have the concept of organizing data into "points" that have "values." In addition to their values, points often contain data such as "point description" or "point is in alarm." But these data may be named, structured, and/or accessed differently in different protocols.