SLOVENSKI STANDARD
SIST EN 50090-6-2:2022

01-februar-2022

Stanovanjski in stavbni elektronski sistemi (HBES) - 6-2. del: Semantični opis ontološkega modela

Home and Building Electronic Systems (HBES)- Part 6-2 IoT Semantic Ontology model description

Systèmes électroniques pour les foyers domestiques et les bâtiments (HBES) Partie 6-2: Semantic Ontology Model Description pour l'internet des objets

Ta slovenski standard je istoveten z: **EN 50090-6-2:2021**

ICS:

35.240.67	Uporabniške rešitve IT v gradbeništvu	IT applications in building and construction industry
97.120	Avtomatske krmilne naprave za dom	Automatic controls for household use

SIST EN 50090-6-2:2022 **en,fr**

EUROPEAN STANDARD

NORME EUROPÉENNE

EUROPÄISCHE NORM

EN 50090-6-2

December 2021

ICS 97.120; 35.240.67

English Version

Home and Building Electronic Systems (HBES)- Part 6-2 IoT Semantic Ontology model description

Systèmes électroniques pour les foyers domestiques et les bâtiments (HBES) - Partie 6-2: Description du modèle ontologie sémantique IoT

Elektrische Systemtechnik für Heim und Gebäude (ESHG) - Teil 6-2: Beschreibung des IoT semantischen Ontologiemodells

This European Standard was approved by CENELEC on 2021-09-20. CENELEC members are bound to comply with the CEN/CENELEC Internal Regulations which stipulate the conditions for giving this European Standard the status of a national standard without any alteration.

Up-to-date lists and bibliographical references concerning such national standards may be obtained on application to the CEN-CENELEC Management Centre or to any CENELEC member.

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

CENELEC members are the national electrotechnical committees of Austria, Belgium, Bulgaria, Croatia, Cyprus, the Czech Republic, Denmark, Estonia, Finland, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Lithuania, Luxembourg, Malta, the Netherlands, Norway, Poland, Portugal, Republic of North Macedonia, Romania, Serbia, Slovakia, Slovenia, Spain, Sweden, Switzerland, Turkey and the United Kingdom.

European Committee for Electrotechnical Standardization
Comité Européen de Normalisation Electrotechnique
Europäisches Komitee für Elektrotechnische Normung

CEN-CENELEC Management Centre: Rue de la Science 23, B-1040 Brussels

Contents

Page

European foreword

This document (EN 50090-6-2:2021) has been prepared by CLC/TC 205 “Home and Building Electronic Systems (HBES)”.

The following dates are fixed:

- latest date by which this document has to be implemented at national level by publication of an identical national standard or by endorsement (dop) 2022-09-20
- latest date by which the national standards conflicting with this document have to be withdrawn (dow) 2022-09-20

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

Any feedback and questions on this document should be directed to the users’ national committee. A complete listing of these bodies can be found on the CENELEC website.

1 Scope

This document defines the HBES Information Model and a corresponding data exchange format for the Home and Building HBES Open Communication System.

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

EN 50090-1:2011, *Home and Building Electronic Systems (HBES) - Part 1: Standardization structure*

EN 50090-3-3, *Home and Building Electronic Systems (HBES) - Part 3-3: Aspects of application - HBES Interworking model and common HBES data types*

3 Terms, definitions and abbreviations

3.1 Terms and definitions

For the purposes of this document, the terms and definitions given in EN 50090-1:2011 and the following apply.

ISO and IEC maintain terminological databases for use in standardization at the following addresses:

— IEC Electropedia: available at https://www.electropedia.org/

— ISO Online browsing platform: available at https://www.iso.org/obp

3.1.1
actuator
point performing an *actuation* (executed by a specific *procedure,* with an expected *result*) that changes an Installation state during Runtime

Note 1 to entry:

— The term Actuator can be mapped to sosa:Actuator in the SSN Ontology.

— The subject *actuation* can be mapped to sosa:Actuation in the SSN Ontology.

— The subject *procedure* can be mapped to sosa:Procedure in the SSN Ontology.

— The subject *result* can be mapped to sosa:Result in the SSN Ontology.

3.1.2
Application Function
uses a set of Functions to achieve the desired behaviour of a technical system, typically using a combination of devices exchanging information via their input and output Datapoints

Note 1 to entry: An Application Function may be split into several Functional Blocks with their input and output Datapoints that are logically connected to each other. The Functional Blocks may be located in one or more devices.

EXAMPLE Application Functions examples are “direct electrical heating”, “electrical heating with accumulators”, “warm water heating”, “fan coil air-conditioning” …

Note 2 to entry: The Application Function and Application are meant to be the same. Reason to introduce an alias term is to use a clear (understandable) reference from Application/ Application Function to the corresponding KIM class :ApplicationFunction or to the Function in the Management Client.

3.1.3
aspect
generally, a specific perspective on a system that contains things with different properties; a referencing mechanism to organize KIM elements in a specific perspective

EXAMPLE A Function Point is an ex officio Aspect with an important specific perspective. It is a referencing mechanism to organize together all to a Function Point interoperating Points (all GOs linked to a GA).

3.1.4
BIM
Building Information Model, a digital process to describe and document a building in all its life cycle phases, from its planning, construction, operation up to its demolition

3.1.5
channel
collection of Datapoints of a device that are logically related to each other typically by association with a hardware feature or a specific function of that device

Note 1 to entry: These Datapoints may be derived from one or more defined Functional Blocks or may be an expansion above and beyond defined Functional Blocks or may be independent of a Functional Block if none is defined for the function associated with the Channel. The concept of a Channel is well-understood by the market participant, e.g. installers.

3.1.6
datapoint
represents a logical input entity of a device acting as recipient of Installation state data, whereas a logical output of a device acts as source of Installation state data

Note 1 to entry: In case of implementation as a Group Object, state data is communicated with the use of Function Points.

Note 2 to entry: The term Datapoint is the common term; to specifically denote a Datapoint available on an IoT 3rd Party API, the term IoT Datapoint is used.

3.1.7
device
physical element that is part of the network; it is a physical, concrete object that a customer can buy

3.1.8
endpoint
entry point to a service, a process, or a queue or topic destination in service-oriented architecture

3.1.9
Feature of Interest
abstraction of a real-world thing (phenomenon, equipment, person, event…) defined by its observable or actuatable properties

Note 1 to entry: In colloquial terms, a FOI is a property carrier.

Note 2 to entry: A Sensor operates on a FOI with observable properties, an Actuator with actuatable properties.

Note 3 to entry: A FOI is not a “classification/type” tag itself; the “classification/ type” is accomplished with the help of tags. Examples are defined in 4.5.1.4.

3.1.10
function
describes a part of the intended behaviour of a FB in a building context

3.1.11
Functional Block
consists of one or more Functions that belong together and that cannot be separated across two devices but big enough that a device with only one such Functional Block could be marketed

Note 1 to entry: A Functional Block has a well-defined black box behaviour.

3.1.12
Function Point
runtime system state information of a specific Application Function

Note 1 to entry: Shared by at least two Datapoints.

Note 2 to entry: Has a unique identifier that addresses a group of controlled objects. This identifier is called a Group Address.

EXAMPLE < Light Switch > in living room on/off, whereas the < … > is the Function Point name

3.1.13
Group Address
numerical identifier of a Function Point

3.1.14
Group Communication
communication model in which one sender communicates information to one and typically more receivers

Note 1 to entry: In IoT, this can be realized by simple UDP communication or by using a message broker system or other.

3.1.15
Group Object
foreseen for Group Communication using Group Address(es), may be accessed via point-to-point communication without an assigned Group Address; with assigned Group Address, it becomes a member of that Function Point represented by the Group Address

3.1.16
HBES Information Model
ontology based model of HBES System relevant parts, including additional semantic (dictionary) information

Note 1 to entry: It is managed by the KNX Association, hence the abbreviation KIM.

3.1.17
Industry Foundation Classes
open standard to describe BIM data in a digital way

Note 1 to entry: IFC data and models are specified in ISO 16739-1.

3.1.18
installation
assembly of materials and components (devices) placed in position to provide a service

Note 1 to entry: An Installation is a deployed system (e.g. HVAC system or fire protection system) and consists of equipment and Functions that are used for a particular purpose.

Note 2 to entry: In relation to this term created data correlates to the installation model, described in 4.2.

[SOURCE: ISO 6707-1:2020, modified – added "(devices)" and Notes to entry.]

3.1.19
IoT Datapoint
represents an Endpoint at an IoT 3rd Party API that:

a) corresponds to one or more Function Points, such as a state data representation of a discrete state in a building context:

EXAMPLE 1 brightness → discrete state “brightness” is represented by the value 65 (percent)

b) is a fully qualified URL e.g. provided by an IoT 3rd Party Server

EXAMPLE 2 https://gateway.hbes.local/hbes/api/v1/datapoints/{Id}

3.1.20
IoT Function
represents a Function at an IoT 3rd Party API that:

— is as a collection of IoT Datapoints that fulfils a – by the user – intended behaviour

EXAMPLE “living room – rear light dimming”, “kitchen – floor heating”

Note 1 to entry: In a Mac, an IoT Function is instantiated data of a MaC Function in an Installation respectively MaC project. The MaC Function itself may base on an Application Function.

3.1.21
IoT 3rd Party API
set of requirements and regulations through which partial access to an Installation can be gained by offering a collection of Endpoints

3.1.22
IoT 3rd Party Client
device or service interacting with the Installation from outside using the IoT 3rd Party API

Note 1 to entry: The IoT 3rd Party Client connects to a single device that provides the IoT 3rd Party API and can use this single device to fully interact with the Installation, possibly depending on a specified authorization mechanism.

EXAMPLE 1 A mobile phone (from inside the network, or from an Internet connection) with typically short period connections.

EXAMPLE 2 A weather service permanently feeding in its weather information using the IoT 3rd Party API.

3.1.23
IoT 3rd Party Server
device that implements the IoT 3rd Party API

Note 1 to entry: This can be a dedicated device; this can be a function of a device that supports other HBES IoT and non HBES functionalities; it may be located within the local LAN of the IoT installation or outside.

3.1.24
MaC Catalog Entry
created management client data correlating to the product model, described in 4.2

3.1.25
MaC Function
Application Function created by the MaC and assigned to a building structure element, grouping several Group Addresses

3.1.26
MaC Project
project created by a MaC documenting the Configuration of an Installation

3.1.27
Management Client
means to configure and commission Devices as well as to plan, design and diagnose an entire Installation

Note 1 to entry: The MaC is used to configure and commission Devices, as well as to plan, design and diagnose an entire Installation. As a final step the MaC writes specific configuration data such as Device parameters to the Devices.

3.1.28
ontology
conceptual descriptions of things that have a real-world commonality sharing the knowledge of a domain, mainly expressed with OWL

Note 1 to entry: Ontologies are a structured way to describe the meaning of data in ontology classes and should not be mixed up with common data model structures.

3.1.29
Object Property
in OWL a built-in concept that **connects** pairs of individuals, an object property expression **represents** the (entire) relationship between the pairs of individuals

3.1.30
OWL
OWL 2 Web Ontology Language, informally OWL 2, specified by the World Wide Web Consortium (W3C), mainly serialized with XML syntax for RDF (RDF/XML)

Note 1 to entry: In this specification the abbreviation OWL is always an explicit reference to OWL 2.

3.1.31
point
represents an interface to data in the system

Note 1 to entry: This document uses the term Point as an umbrella for data that can be accessed from outside of the Device, for instance to interact with other Points from other Devices. Consequently, term Point is a generic superset of the term Datapoint (which describes more precisely the technics how the “data” in the system are structured and/or coded).

3.1.32
Point API
simple RESTful (CoAP or HTTP) application programming interface designed for, but not limited to, constrained class 2 devices [RFC7228] supporting device individualization, device linking and accessing device runtime data (e.g. Functional Block or Channel Datapoints)

3.1.33
Quality Kind
represents a certain combination of observable or actuatable properties, available as predefined parts of the Semantic Dictionary or created individually during Configuration; the latter is the case when a Quality Kind with the intended combination of properties respectively tags is not (yet) part of the dictionary

Note 1 to entry: A QK is not a “classification/type” tag itself; the “classification/ type” is accomplished with the help of tags. Examples are defined in 4.5.1.4.

3.1.34
RDF
Resource Description Framework, as specified by the https://www.w3.org/RDF/

Note 1 to entry: RDF is a framework to represent information in the web by using triples. The information can be serialized and stored in many formats such as the TURTLE or JSON(-LD) format. The general RDF concept description can be found under https://www.w3.org/TR/rdf11-concepts/

3.1.35
runtime
process-to-process communication of data between devices, opposing to Configuration

Note 1 to entry: This concerns mainly the communication of Datapoint values (control and status information).

3.1.36
Semantic Export
project exported by the MaC reflecting an Installation in a linked data format

Note 1 to entry: The exported data is:

— structured according to the KIM, such as using Object Properties defined in KIM;

— annotated with additional semantic information from the Semantic Dictionary;

— referencing concepts of external Ontologies.

3.1.37
semantic dictionary
set of standardized terms allowing to annotate required parts of an Installation

Note 1 to entry: For details, see 4.2.8.

3.1.38
sensor
point performing an *observation* (executed by a specific *procedure*, triggered by a *stimulus*), responding a *result* as an Installation state during Runtime

Note 1 to entry:

— The term Sensor can be mapped to sosa:Sensor in the SSN Ontology.

— The subject observation can be mapped to sosa:Observation in the SSN Ontology.

— The subject stimulus can be mapped to ssn:Stimulus in the SSN Ontology.

— The subject *procedure* can be mapped to sosa:Procedure in the SSN Ontology.

— The subject *result* can be mapped to sosa:Result in the SSN Ontology.

3.1.39
tag
kind of annotation term used to extend available data with (in most cases) well known standardized information from a dictionary (in contrast to user defined, arbitrary term)

Note 1 to entry: A Tag is a concept-less term, without an integration in a broader concept such as the concept of a Datapoint (used in an Application Function), it has a limited semantic meaning.

EXAMPLE Term “flow” has a weak meaning on its own, but if you relate it in a FOI with the other term “water” this expresses at least that you observe/ actuate the water flow.

In this specification a Tag is almost exclusively a term from the Semantic Dictionary.

3.1.40
thing description
semantic metadata model to describe (abstract or physical) things, as specified by the thing description https://www.w3.org/TR/wot-thing-description/ and thing Ontology https://www.w3.org/2019/wot/td

Note 1 to entry: TD relevant relations are described in the clause of Semantic Export.

3.2 Abbreviations

For the purposes of this document, the following abbreviations apply.

DHWC	Domestic Hot Water Controller
FOI	Function of Interest
BOC	Boiler Controller
BUC	Burner Controller
FTC	Flow Temperature Controller
GA	Group Address
GO	Group Object
FB	Functional Block
FP	Function Point
HBES	Home and Building Electronic Systems
HDTRT	Heat Demand Transformer Room Temperature
HFDM	Heat Flow Demand Manager
HIRC	Heating Individual Room Controller
HPM	Heat Producer Manager
HVA	Heating Valve Actuator
HZC	Heating Zone Controller
IFC	Industry Foundation Classes
IOO	Info On off
IO	Input Output
IoT	Internet of Things
KIM	HBES Information Model managed by KNX Association
KNXA	KNX Association
MaC	Management Client
OP	Object Property
QK	Quality Kind
RSM	Room Setpoint Manager
RTC	Room Temperature Controller
RTS	Room Temperature Sensor
SOO	Switch on/off
TD	Thing Description

4 HBES Information Model

4.1 Motivation and current situation

The current HBES model/ data information is based on XML, is managed by the KNX Association and has its corresponding versioning. Project/ product data exported by existing HBES management clients are in line with this XML schema.

The HBES Management clients use the XML schema mainly to define the corresponding data structures to store/export project and product data. Consequently, the XML schema is always updated when new versions of the HBES management clients are published, and new project and/or product features demand a change of the data structures.

Though XML itself is a well-known and widespread format it has its drawbacks as regards:

— sharing model/project data information with external clients that use this data need to be synchronized when a new XML schema version is announced;

— describing, expressing, mapping and sharing (semantic) information in the IoT domain.

The aim and motivation is to define a HBES Information Model and a corresponding data exchange format:

— The model expresses only the current - by external clients requested – information.

— The model can be also easily updated.

— The exported data uses a widely used data exchange format, which should also be readable by humans, meaning it is text based.

This model and data exchange format is more stable, compared to the frequent HBES management client evolution.

— The HBES Information Model will be available as ontology in one or more formats, such as turtle files.

— The data exported by HBES management clients will be available as linked data, such as JSON-LD files.

The HBES IoT protocol suite shall support semantic information, both for runtime as well as for configuration.

This information shall be brought to the system components in a data driven way, by the HBES management clients software and possible other sources. It shall thus build on the information provided by the HBES management clients user, to avoid having to be entered again in the 3rd Party Client configuration.

The semantic information shall comply with the HBES standard and be available as public information. Technically this information will be available as linked data, expressed with triples.

The use of semantics itself allows to formalize, restrict, and verify the usage of HBES subjects to describe entities, relationships, and tags (entities have some relationship predicate to some other entities— essentially a directed edge in a graph).

This simple (triple) structure enables the succinct and elegant composition of large, interconnected structures of facility (building) subsystems.

The KIM shall be able express relations in an Installation to support the following use cases to request/ query data.

— How is a building structured with floors, rooms and other?

— Where is an equipment (e.g. a Device) installed in a location (assembly place)?

— Which overall functionality is hosted on a Device?

— Which interface (e.g. a Point) is provided by a functionality or Device (Channel, FB)?

— Which Application Function affects which building structure elements, such as a room?

— What does an Application Function or Point observe or actuate upon in the real world (key word → feature of interest with substance, phenomenon and equipment)?

— In which operational domain or for which trades does an Application Function operate (key word → assignments of trades for Application Functions)?

— Which Points belong to an Application Function and what is the purpose of the Point (key word → tag information: *Logical Input*/ *Logical Output*, *Set Value* or *Parameter Point*).

— Which phase feeds the load (e.g. *Light Fixture/ Luminaire, Motor*)?

— What are the setpoint and current temperature values in a specific location (room, zone) / a list of locations?

— What is the current operation mode of primary systems (heating, cooling, ventilation, hot water supply) in general or in location x?

— What is the identifier of a specific heating circuit (= connection from heating source to heating sink) and in which heating circuit is a radiator located?

NOTE Heating Circuit = It is possible to express that n Points have a Locality “Circuit” and a “Phenomenon” Heat assigned, they are distributed over several Devices and located within several locations, operating on several Application Functions, finally grouped as an Aspect.

— What is the power consumption of a specific Device containing Points operating on a *Luminaire* / *Air Conditioner* / *Fridge*?

— Which Function Points control a specific *Luminaire* / *Heating Circuit* (see above) …?

— What is the current operation time of a certain *Luminaires? As a note, a* Client can then compare this time to the maintenance threshold operation time, not expressed in KIM.

— Which functional information as expressed by the Tag Information is represented by or associated with a Point?

For examples, see 4.4.2.5.2 or 4.5.1.4.

4.2 Introduction

4.2.1 General

The HBES System is designed for direct exchange of information (i.e. *communication*) between networked *devices* controlling *applications* in and around buildings respectively *locations* (see 3.1).

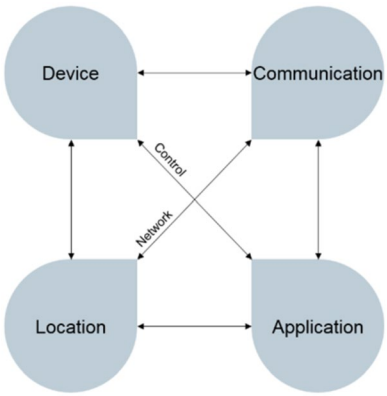

Figure 1 — HBES System elements

To describe the entire HBES System several models are needed to reflect the different aspects of it:

1. **Product model**

   — to model products with their type/applicative behaviour, their configuration, their catalog information, independent from an actual usage in an Installation;

   — mainly makes use of the Device and Tag Model from this document.

   Model data are also colloquially called (not instantiated) **catalog data**.

   EXAMPLE 1 device model aspect → memory footprint for configuration parameters and Datapoints

2. **System model**

   — to model elements and dependencies of the physical processes, see words “Communication”, (networked) “Devices” and “Application” described in above Figure 1;

   — mainly makes use of Device and Application Function Model from this document. Also uses some parts of the Installation Model such as Points.

   EXAMPLE 2 topology model aspect → the physical network configuration of a device

3. **Installation model**

   — to model elements and dependencies correlating to instantiated data of a real- world Installation, uses as (input) the product model, behaves according to (and respecting) the system model.

   — Mainly makes use of Location, Application Function and Tag Model from this document. Model data are also called colloquially **instance data**.

   EXAMPLE 3 location model aspect → The actual spatial building structure of an Installation which is independent from the HBES System. The same Installation would technically operate/ communicate also in a different building, regardless if original goal is met.

As a note, this document mainly addresses the installation model, the Aspect of an Installation documented in a Project from a Management Client, with instantiated HBES Devices or Functions etc.