

DIN EN 50090-6-2

ICS 35.240.67; 97.120

**Elektrische Systemtechnik für Heim und Gebäude (ESHG) –
Teil 6-2: Beschreibung des IoT semantischen Ontologiemodells;
Deutsche Fassung EN 50090-6-2:2021**

Home and Building Electronic Systems (HBES) –
Part 6-2 IoT Semantic Ontology model description;
German version EN 50090-6-2:2021

Systèmes électroniques pour les foyers domestiques et les bâtiments (HBES) –
Partie 6-2: Description du modèle ontologie sémantique IoT;
Version allemande EN 50090-6-2:2021

Gesamtumfang 106 Seiten

DKE Deutsche Kommission Elektrotechnik Elektronik Informationstechnik in DIN und VDE

Anwendungsbeginn

Anwendungsbeginn für die von CENELEC am 2021-09-20 angenommene Europäische Norm als DIN-Norm ist 2023-06-01.

Nationales Vorwort

Vorausgegangener Norm-Entwurf: E DIN EN 50090-6-2:2021-02.

Für dieses Dokument ist das nationale Arbeitsgremium DKE/K 716 „Elektrische Systemtechnik für Heim und Gebäude (ESHG)“ der DKE Deutsche Kommission Elektrotechnik Elektronik Informationstechnik in DIN und VDE (www.dke.de) zuständig.

Aktuelle Informationen zu diesem Dokument können über die Internetseiten von DKE (www.dke.de) und DIN (www.din.de) durch eine Suche nach der Dokumentennummer aufgerufen werden.

Für den Fall einer undatierten Verweisung im normativen Text (Verweisung auf ein Dokument ohne Angabe des Ausgabedatums und ohne Hinweis auf eine Abschnittsnummer, eine Tabelle, ein Bild usw.) bezieht sich die Verweisung auf die jeweils aktuellste Ausgabe des in Bezug genommenen Dokuments.

Für den Fall einer datierten Verweisung im normativen Text bezieht sich die Verweisung immer auf die in Bezug genommene Ausgabe des Dokuments.

Der Zusammenhang der zitierten Dokumente mit den entsprechenden deutschen Dokumenten ergibt sich, soweit ein Zusammenhang besteht, grundsätzlich über die Nummer der entsprechenden IEC-Publikation. Beispiel: IEC 60068 ist als EN 60068 als Europäische Norm durch CENELEC übernommen und als DIN EN 60068 ins Deutsche Normenwerk aufgenommen.

Das Original-Dokument enthält Bilder in Farbe, die in der Papierversion in einer Graustufen-Darstellung wiedergegeben werden. Elektronische Versionen dieses Dokuments enthalten die Bilder in der originalen Farbdarstellung.

Deutsche Fassung

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

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

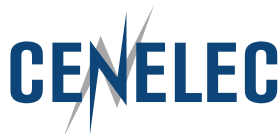
Systèmes électroniques pour les foyers
domestiques et les bâtiments (HBES) – Partie 6-2:
Description du modèle ontologie sémantique IoT

Diese Europäische Norm wurde von CENELEC am 2021-09-20 angenommen. CENELEC-Mitglieder sind gehalten, die CEN/CENELEC-Geschäftsordnung zu erfüllen, in der die Bedingungen festgelegt sind, unter denen dieser Europäischen Norm ohne jede Änderung der Status einer nationalen Norm zu geben ist.

Auf dem letzten Stand befindliche Listen dieser nationalen Normen mit ihren bibliographischen Angaben sind beim CEN-CENELEC Management Centre oder bei jedem CENELEC-Mitglied auf Anfrage erhältlich.

Diese Europäische Norm besteht in drei offiziellen Fassungen (Deutsch, Englisch, Französisch). Eine Fassung in einer anderen Sprache, die von einem CENELEC-Mitglied in eigener Verantwortung durch Übersetzung in seine Landessprache gemacht und dem CEN-CENELEC Management Centre mitgeteilt worden ist, hat den gleichen Status wie die offiziellen Fassungen.

CENELEC-Mitglieder sind die nationalen elektrotechnischen Komitees von Belgien, Bulgarien, Dänemark, Deutschland, Estland, Finnland, Frankreich, Griechenland, Irland, Island, Italien, Kroatien, Lettland, Litauen, Luxemburg, Malta, den Niederlanden, Norwegen, Österreich, Polen, Portugal, der Republik Nordmazedonien, Rumänien, Schweden, der Schweiz, Serbien, der Slowakei, Slowenien, Spanien, der Tschechischen Republik, der Türkei, Ungarn, dem Vereinigten Königreich und Zypern.



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

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

Inhalt

	Seite
Europäisches Vorwort.....	6
1 Anwendungsbereich.....	7
2 Normative Verweisungen.....	7
3 Begriffe und Abkürzungen.....	7
3.1 Begriffe.....	7
3.2 Abkürzungen.....	13
4 ESHG-Informationsmodell.....	14
4.1 Hintergrund und aktuelle Lage.....	14
4.2 Einleitung.....	16
4.2.1 Allgemeines.....	16
4.2.2 KIM – Inhalt.....	17
4.2.3 KIM – Version.....	19
4.2.4 KIM –Verfügbarkeit.....	19
4.2.5 KIM – Datenformat und Datenaustauschformat.....	20
4.2.6 KIM – Ontologie-IRIs und –Namensräume.....	20
4.2.7 KIM – Ontologieklassen.....	22
4.2.8 KIM – semantisches Wörterbuch.....	22
4.3 Ortsmodell.....	27
4.3.1 Einleitung.....	27
4.3.2 Anforderungen.....	29
4.3.3 Klassen und Unterklassen.....	30
4.4 Installationsmodell.....	36
4.4.1 Einleitung.....	36
4.4.2 Klassen und Unterklassen.....	37
4.5 Tag-Modell.....	68
4.5.1 Einleitung.....	68
4.5.2 Tag-Modell – Punkte.....	74
4.5.3 Tag-Modell – Funktionspunkte.....	75
4.5.4 Tag-Modell – Anwendungsfunktion.....	75
4.5.5 Klassen und Unterklassen.....	76
4.5.6 Tag-Kardinalitäten.....	91
4.6 Modellrelationen.....	91
4.6.1 Allgemeines.....	91
4.6.2 Ortsrelationen.....	91
4.6.3 Installationsrelationen.....	93
4.6.4 Tag-Relationen.....	102

Bilder

Bild 1 – Elemente eines ESHG-Systems.....

Bild 2 – Hauptquellen des KIM..... 18

Bild 3 – Wichtigste KIM-Ontologieklassen..... 19

Bild 4 – Ortsmodell..... 28

Bild 5 – Beispiel für Orte und Relationen..... 29

Bild 6 – Installationsmodell..... 37

Bild 7 – Verschiedene Sichtweisen auf Funktionalität..... 40

Bild 8 – Komponenten von ESHG-Geräten..... 53

Bild 9 – ESHG-Gerätemodell..... 54

Bild 10 – Prinzip des Tagging-Modells..... 71

Bild 11 – Tagging-Modell für Katalogdatenpunkte..... 74

Bild 12 – Tagging-Modell für Instanzdatenpunkte..... 75

Bild 13 – Tagging-Modell für Funktionspunkte..... 75

Bild 14 – Tagging-Modell für Anwendungsfunktionen..... 76

Tabellen

Tabelle 1 – KIM-Namensräume, zugehörige IRIs und Ontologiereferenz..... 21

Tabelle 2 – Betrachtete Merkmale und deren OP-Relationsprädikate..... 73

Europäisches Vorwort

Dieses Dokument (EN 50090-6-2:2021) wurde vom CLC/TC 205 "Home and Building Electronic Systems (HBES)" ausgearbeitet.

Nachstehende Daten wurden festgelegt:

- spätestes Datum, zu dem dieses Dokument auf nationaler Ebene durch Veröffentlichung einer identischen nationalen Norm oder durch Anerkennung übernommen werden muss (dop): 2022-09-20
- spätestes Datum, zu dem nationale Normen, die diesem Dokument entgegenstehen, zurückgezogen werden müssen (dow): 2024-09-20

Es wird auf die Möglichkeit hingewiesen, dass einige Elemente dieses Dokuments Patentrechte berühren können. CENELEC ist nicht dafür verantwortlich, einige oder alle diesbezüglichen Patentrechte zu identifizieren.

Rückmeldungen oder Fragen zu diesem Dokument sollten an das jeweilige nationale Komitee des Anwenders gerichtet werden. Eine vollständige Liste dieser Gremien ist auf den Internetseiten des CENELEC abrufbar.

1 Anwendungsbereich

Dieses Dokument legt das ESHG-Informationsmodell und ein zugehöriges Datenaustauschformat für das offene Kommunikationssystem für elektrische Systemtechnik für Heim und Gebäude (ESHG) fest.

2 Normative Verweisungen

Die folgenden Dokumente werden im Text in solcher Weise in Bezug genommen, dass einige Teile davon oder ihr gesamter Inhalt Anforderungen des vorliegenden Dokuments darstellen. Bei datierten Verweisungen gilt nur die in Bezug genommene Ausgabe. Bei undatierten Verweisungen gilt die letzte Ausgabe des in Bezug genommenen Dokuments (einschließlich aller Änderungen).

EN 50090-1:2011, *Elektrische Systemtechnik für Heim und Gebäude (ESHG) - Teil 1: Aufbau der Norm*

EN 50090-3-3, *Elektrische Systemtechnik für Heim und Gebäude (ESHG) - Teil 3-3: Anwendungsaspekte - ESHG-Interworking-Modell und übliche ESHG-Datenformate*

3 Begriffe und Abkürzungen

3.1 Begriffe

Für die Anwendung dieses Dokuments gelten die Begriffe nach EN 50090-1:2011 und die folgenden Begriffe.

ISO und IEC stellen terminologische Datenbanken für die Verwendung in der Normung unter den folgenden Adressen bereit:

- IEC Electropedia: verfügbar unter <https://www.electropedia.org/>
- ISO Online browsing platform: verfügbar unter <https://www.iso.org/obp>

3.1.1

Aktor

en **actuator**

Punkt, der eine *Betätigung* vornimmt (ausgeführt von einem bestimmten *Verfahren* mit einem erwarteten *Ergebnis*), die den Zustand einer Installation zur Laufzeit ändert

Anmerkung 1 zum Begriff: zum Begriff:

- Der Begriff „Actuator“ (Aktor) kann auf `sosa:Actuator` in der SSN-Ontologie abgebildet werden.
- Das Subjekt *Betätigung* kann auf `sosa:Actuation` in der SSN-Ontologie abgebildet werden.
- Das Subjekt *Verfahren* kann auf `sosa:Procedure` in der SSN-Ontologie abgebildet werden.
- Das Subjekt *Ergebnis* kann auf `sosa:Result` in der SSN-Ontologie abgebildet werden.

3.1.2

Anwendungsfunktion

en **application function**

verwendet eine Reihe von Funktionen, um das gewünschte Verhalten eines technischen Systems herbeizuführen, üblicherweise eine Kombination von Geräten, die über ihre Eingangs- und Ausgangsdatenpunkte Informationen untereinander austauschen

Anmerkung 1 zum Begriff: Eine Anwendungsfunktion kann in mehrere Funktionsblöcke mit eigenen Eingangs- und Ausgangsdatenpunkten aufgeteilt sein, die logisch miteinander verbunden sind. Die Funktionsblöcke können in einem oder mehreren Geräten untergebracht sein.

BEISPIEL Beispiele für Anwendungsfunktionen sind „direkte Elektroheizung“, „Elektroheizung mit Speichern“, „Warmwasserheizung“, „Gebläsekonvektor-Klimaanlage“ usw.

Anmerkung 2 zum Begriff: Anwendungsfunktion und Anwendung sind gleichbedeutend. Mit der Einführung des Synonyms

soll eine eindeutige (nachvollziehbare) Bezugnahme von Anwendung/Anwendungsfunktion auf die entsprechende KIM-Klasse :ApplicationFunction oder die Funktion im Management-Client ermöglicht werden.

3.1.3

Aspekt

en **aspect**

im Allgemeinen ein bestimmter Blickwinkel auf ein System, das Dinge mit verschiedenen Eigenschaften enthält; ein Referenzierungsmechanismus für die Organisation von KIM-Elementen unter einem bestimmten Gesichtspunkt

BEISPIEL Ein Funktionspunkt ist definitionsgemäß ein Aspekt mit einem wichtigen spezifischen Blickwinkel. Es handelt sich um einen Referenzierungsmechanismus für die Organisation aller mit einem Funktionspunkt zusammenwirkenden Punkte (aller mit einer GA verknüpften GOs).

3.1.4

BIM

Building Information Model (de: Bauwerksinformationsmodell), ein digitaler Prozess für die Beschreibung und Dokumentation eines Bauwerks über sämtliche Phasen von dessen Lebenszyklus hinweg, von der Planung über den Bau und die Nutzung bis zum Abriss

3.1.5

Kanal

en **channel**

Zusammenstellung von Datenpunkten eines Gerätes, die logisch miteinander zusammenhängen, üblicherweise durch die Zuweisung zu einem Hardwaremerkmal oder einer bestimmten Funktion dieses Gerätes

Anmerkung 1 zum Begriff: Diese Datenpunkte dürfen von einem oder mehreren definierten Funktionsblöcken abgeleitet sein, eine Erweiterung von definierten Funktionsblöcken sein oder unabhängig von einem Funktionsblock sein, wenn für die mit dem Kanal verknüpfte Funktion keiner definiert ist. Das Konzept eines Kanals ist unter den Marktteilnehmern, beispielsweise Installateuren, bekannt.

3.1.6

Datenpunkt

en **datapoint**

Repräsentation einer logischen Eingangsentität eines Gerätes, die als Empfänger von Zustandsdaten einer Installation agiert, während ein logischer Ausgang eines Gerätes als Quelle von Zustandsdaten einer Installation agiert

Anmerkung 1 zum Begriff: Bei einer Implementierung als Gruppenobjekt werden Zustandsdaten durch die Verwendung von Funktionspunkten übertragen.

Anmerkung 2 zum Begriff: Der Begriff „Datenpunkt“ ist ein allgemeiner Begriff; um speziell einen an einer IoT-Drittanbieter-API verfügbaren Datenpunkt zu beschreiben, wird der Begriff „IoT-Datenpunkt“ verwendet.

3.1.7

Gerät

en **device**

physisches Element, das Bestandteil des Netzwerks ist; es handelt sich um ein physisches, konkretes Objekt, das ein Kunde erwerben kann

3.1.8

Endpunkt

en **endpoint**

Zugangspunkt zu einem Dienst, einem Prozess oder einem Queue- oder Topic-Ziel in einer diensteorientierten Architektur

3.1.9

betrachtetes Merkmal

en **feature of interest**

Abstraktion eines Elements aus der realen Welt (Phänomen, Ausrüstung, Person, Ereignis ...), das durch seine

beobachtbaren oder betätigbaren Eigenschaften definiert wird

Anmerkung 1 zum Begriff: Einfach gesagt ist ein FOI ein Träger von Eigenschaften.

Anmerkung 2 zum Begriff: Ein Sensor bezieht sich auf ein FOI mit beobachtbaren (en: observable) Eigenschaften, ein Aktor auf ein FOI mit betätigbaren (en: actuatable) Eigenschaften.

Anmerkung 3 zum Begriff: Ein FOI stellt selbst kein „Klassifizierung/Typ“-Tag dar; „Klassifizierung/Typ“ wird mithilfe von Tags realisiert. Beispiele sind in 4.5.1.4 definiert.

3.1.10

Funktion

en **function**

Teil des vorgesehenen Verhaltens eines FB in einem Bauwerkskontext

3.1.11

Funktionsblock

en **functional block**

besteht aus einer oder mehreren Funktionen, die zusammengehören und nicht auf zwei Geräte aufgeteilt werden können, die aber umfangreich genug sind, dass ein Gerät mit nur einem solchen Funktionsblock vertrieben werden könnte

Anmerkung 1 zum Begriff: Ein Funktionsblock weist ein klar definiertes Black-Box-Verhalten auf.

3.1.12

Funktionspunkt

en **function point**

Laufzeitsystem-Zustandsinformationen einer bestimmten Anwendungsfunktion

Anmerkung 1 zum Begriff: Wird von mindestens zwei Datenpunkten gemeinsam genutzt.

Anmerkung 2 zum Begriff: Verfügt über einen eindeutigen Bezeichner, der eine Gruppe gesteuerter Objekte bezeichnet. Dieser Bezeichner wird als „Gruppenadresse“ bezeichnet.

BEISPIEL <Lichtschalter> in Wohnzimmer ein-/ausgeschaltet, wobei <...> den Namen des Funktionspunktes enthält

3.1.13

Gruppenadresse

en **group address**

numerischer Bezeichner eines Funktionspunktes

3.1.14

Gruppenkommunikation

en **group communication**

Kommunikationsmodell, in dem ein Sender Informationen an einen und typischerweise mehrere Empfänger überträgt

Anmerkung 1 zum Begriff: Beim IoT kann dies durch einfache UDP-Kommunikation, durch ein Message-Broker-System oder anderweitig realisiert werden.

3.1.15

Gruppenobjekt

en **group object**

vorgesehen für Gruppenkommunikation mittels Gruppenadresse(n), der Zugriff darf ohne zugewiesene Gruppenadresse über Punkt-zu-Punkt-Kommunikation erfolgen; mit zugewiesener Gruppenadresse wird es zu einem Mitglied des Funktionspunktes, den die Gruppenadresse repräsentiert

3.1.16

ESHG-Informationsmodell

en HBES information model

ontologiebasiertes Modell der für ein ESHG-System relevanten Teile, einschließlich zusätzlicher semantischer (Wörterbuch-)Informationen

Anmerkung 1 zum Begriff: Die Verwaltung erfolgt durch die KNX Association, daher die Abkürzung KIM.

3.1.17

Industry Foundation Class

offener Standard zur digitalen Beschreibung von BIM-Daten

Anmerkung 1 zum Begriff: Die IFC-Daten und -Modelle werden in ISO 16739-1 festgelegt.

3.1.18

Installation

Zusammenstellung von Materialien und Komponenten (Geräten), die auf eine bestimmte Art und Weise angeordnet werden, um einen bestimmten Dienst zu erbringen

Anmerkung 1 zum Begriff: Eine Installation ist ein eingerichtetes System (z. B. ein HLK-System oder eine Brandschutzanlage) und besteht aus Ausrüstung und Funktionen, die einem bestimmten Zweck dienen.

Anmerkung 2 zum Begriff: Im Zusammenhang mit diesem Begriff beziehen sich erstellte Daten auf das in 4.2 beschriebene Installationsmodell.

[QUELLE: ISO 6707-1:2020, modifiziert – „(Geräten)“ und Anmerkungen zum Begriff hinzugefügt]

3.1.19

IoT-Datenpunkt

en IoT datapoint

Repräsentation eines Endpunktes an einer IoT-Drittanbieter-API, der:

- a) einem oder mehreren Funktionspunkten entspricht, wie eine Zustandsdatendarstellung eines diskreten Zustands im Bauwerkskontext:

BEISPIEL 1 Helligkeit der diskrete Zustand „Helligkeit“ wird durch den Wert 65 (Prozent) repräsentiert

- b) eine vollqualifizierte URL ist, die z. B. von einem IoT-Drittanbieter-Server bereitgestellt wird

BEISPIEL 2 <https://gateway.hbes.local/hbes/api/v1/datapoints/{Id}>

3.1.20

IoT-Funktion

en IoT function

Repräsentation einer Funktion an einer IoT-Drittanbieter-API, die:

- sich wie eine Zusammenstellung von IoT-Datenpunkten verhält, die ein – vom Anwender – gewünschtes Verhalten aufweist

BEISPIEL „Wohnzimmer – Dimmen der Hintergrundbeleuchtung“, „Küche – Fußbodenheizung“

Anmerkung 1 zum Begriff: Bei einem MaC entspricht eine IoT-Funktion instanziierten Daten einer MaC-Funktion in einer Installation bzw. einem MaC-Projekt. Die MaC-Funktion selbst kann auf einer Anwendungsfunktion beruhen.

3.1.21

IoT-Drittanbieter-API

en IoT 3rd party API

Satz von Anforderungen und Vorschriften, in dessen Rahmen ein teilweiser Zugriff auf eine Installation ermöglicht wird, indem eine Zusammenstellung von Endpunkten bereitgestellt wird

3.1.22

IoT-Drittanbieter-Client

en **IoT 3rd party client**

Gerät oder Dienst, das/der über die IoT-Drittanbieter-API von außen mit der Installation interagiert

Anmerkung 1 zum Begriff: Der IoT-Drittanbieter-Client verbindet sich mit einem einzelnen Gerät, das die IoT-Drittanbieter-API bereitstellt, und kann über dieses Gerät vollumfänglich mit der Installation interagieren, gegebenenfalls in Abhängigkeit von einem festgelegten Autorisierungsmechanismus.

BEISPIEL Ein Mobiltelefon (das Teil des Netzwerks oder über eine Internetverbindung verbunden ist), mit dem üblicherweise jeweils nur für kurze Zeitspannen eine Verbindung hergestellt wird.

BEISPIEL Ein Wetterdienst, der die Informationen zum Wetter in einem Feed beständig über die IoT-Drittanbieter-API überträgt.

3.1.23

IoT-Drittanbieter-Server

en **IoT 3rd party server**

Gerät, das die IoT-Drittanbieter-API implementiert

Anmerkung 1 zum Begriff: Hierbei kann es sich um ein eigens dafür vorgesehenes Gerät handeln; es kann eine Funktion eines Gerätes sein, das noch weitere ESHG-IoT-Funktionalitäten und Nicht-ESHG-Funktionalitäten unterstützt; es kann sich innerhalb oder außerhalb des LAN der IoT-Installation befinden.

3.1.24

MaC-Katalogeintrag

en **MaC catalog entry**

erstellte Management-Client-Daten, die sich auf das in 4.2 beschriebene Produktmodell beziehen

3.1.25

MaC-Funktion

en **MaC function**

von dem MaC erstellte Anwendungsfunktion, die einem baulichen Element zugewiesen ist und mit der mehrere Gruppenadressen gruppiert werden

3.1.26

MaC project

en **MaC-Projekt**

von einem MaC erstelltes Projekt, das die Konfiguration einer Installation dokumentiert

3.1.27

Management-Client

Mittel für die Konfiguration und Inbetriebnahme von Geräten sowie für die Planung und Auslegung von und die Diagnosefindung für eine gesamte Installation

Anmerkung 1 zum Begriff: Der MaC dient der Konfiguration und Inbetriebnahme von Geräten sowie der Planung und Auslegung von und der Diagnosefindung für eine gesamte Installation. Im letzten Schritt schreibt der MaC spezifische Konfigurationsdaten wie die Geräteparameter auf die Geräte.

3.1.28

Ontologie

en **ontology**

konzeptuelle Beschreibung von Dingen, die in der realen Welt Gemeinsamkeiten aufweisen, zur Vermittlung von Wissen in einer bestimmten Domäne, wobei die Beschreibung zumeist mittels OWL erfolgt

Anmerkung 1 zum Begriff: Ontologien dienen der strukturierten Beschreibung der Bedeutung von Daten in Ontologieklassen und sollten nicht mit gewöhnlichen Datenmodellstrukturen verwechselt werden.

3.1.29

Objekteigenschaft

en **object property**

in OWL ein eingebautes Konzept, das Paare von Individuen **verbindet**, wobei der Ausdruck einer Objekteigenschaft die (gesamte) Beziehung zwischen den Paaren von Individuen **repräsentiert**

3.1.30

OWL

OWL 2 Web Ontology Language, informell auch „OWL 2“, festgelegt von dem World Wide Web Consortium (W3C), hauptsächlich serialisiert mit XML-Syntax für RDF (RDF/XML)

Anmerkung 1 zum Begriff: In dieser Spezifikation bezieht sich die Abkürzung OWL ausdrücklich stets auf OWL 2.

3.1.31

Punkt

en **point**

Repräsentation einer Schnittstelle zu Daten im System

Anmerkung 1 zum Begriff: In diesem Dokument wird der Begriff „Punkt“ als Sammelbegriff für Daten verwendet, auf die von außerhalb des Gerätes zugegriffen werden kann, beispielsweise, um mit den Punkten anderer Geräte zu interagieren. Der Begriff „Punkt“ ist dementsprechend ein allgemeiner Oberbegriff des Begriffs „Datenpunkt“ (welcher die Techniken, mit denen die „Daten“ im System strukturiert und/oder kodiert werden, genauer beschreibt).

3.1.32

Punkt-API

en **point API**

einfache RESTful-Anwendungsprogrammierschnittstelle (CoAP oder HTTP), die für Beschränkungen unterliegende Geräte der Klasse 2 [RFC 7228] vorgesehen, aber nicht auf diese beschränkt ist und die Individualisierung von Geräten, die Verknüpfung („Linking“) von Geräten und den Zugriff auf die Gerätelaufzeitdaten (z. B. Funktionsblock- oder Kanal-Datenpunkte) unterstützt

3.1.33

Eigenschaftsart

en **quality kind**

Repräsentation einer bestimmten Kombination von beobachtbaren oder betätigbaren Eigenschaften, die als vordefinierter Teil des semantischen Wörterbuchs zur Verfügung steht oder bei der Konfiguration einzeln angelegt wird; Letzteres ist der Fall, wenn eine Eigenschaftsart mit der vorgesehenen Kombination von Eigenschaften bzw. Tags (noch) nicht im Wörterbuch enthalten ist

Anmerkung 1 zum Begriff: Eine QK stellt selbst kein „Klassifizierung/Typ“-Tag dar; „Klassifizierung/Typ“ wird mithilfe von Tags realisiert. Beispiele sind in 4.5.1.4 definiert.

3.1.34

RDF

Resource Description Framework nach <https://www.w3.org/RDF/>

Anmerkung 1 zum Begriff: RDF ist ein Framework für die Darstellung von Informationen im Web mithilfe von Tripeln. Die Informationen können serialisiert und in vielen verschiedenen Formaten wie TURTLE oder JSON(-LD) gespeichert werden. Eine allgemeine Beschreibung des Konzepts des RDF ist unter <https://www.w3.org/TR/rdf11-concepts/> zu finden.

3.1.35

Laufzeit

en **runtime**

Prozess-zu-Prozess-Übertragung von Daten zwischen Geräten, im Gegensatz zu Konfiguration

Anmerkung 1 zum Begriff: Dies betrifft in erster Linie die Übertragung von Datenpunktwerten (Steuer- und Statusinformationen).

3.1.36

semantischer Export en **semantic export**

von dem MaC exportiertes Projekt, das eine Installation in einem Format für verknüpfte Daten widerspiegelt

Anmerkung 1 zum Begriff: Die exportierten Daten:

- werden entsprechend dem KIM strukturiert, beispielsweise mithilfe von im KIM definierten Objekteigenschaften;
- werden mit zusätzlichen semantischen Informationen aus dem semantischen Wörterbuch annotiert;
- referenzieren Konzepte von externen Ontologien.

3.1.37

semantisches Wörterbuch en **semantic dictionary**

Satz von standardisierten Begriffen, mit denen erforderliche Teile einer Installation annotiert werden können

Anmerkung 1 zum Begriff: Für Einzelheiten siehe 4.2.8.

3.1.38

Sensor

Punkt, der eine *Beobachtung* durchführt (ausgeführt von einem bestimmten *Verfahren*, das von einem *Stimulus* ausgelöst wird) und ein *Ergebnis* als einen Installationszustand zur Laufzeit zurückgibt

Anmerkung 1 zum Begriff:

- Der Begriff „Sensor“ kann auf `sosa:Sensor` in der SSN-Ontologie abgebildet werden.
- Das Subjekt „Beobachtung“ kann auf `sosa:Observation` in der SSN-Ontologie abgebildet werden.
- Das Subjekt „Stimulus“ kann auf `ssn:Stimulus` in der SSN-Ontologie abgebildet werden.
- Das Subjekt *Verfahren* kann auf `sosa:Procedure` in der SSN-Ontologie abgebildet werden.
- Das Subjekt *Ergebnis* kann auf `sosa:Result` in der SSN-Ontologie abgebildet werden.

3.1.39

Tag

Art von Annotationsbegriff, mit dem die verfügbaren Daten um (in den meisten Fällen) bekannte standardisierte Informationen aus einem Wörterbuch ergänzt werden (im Gegensatz zu einem benutzerdefinierten, selbst gewählten Begriff)

Anmerkung 1 zum Begriff: Ein Tag ist ein konzeptloser Begriff, der nur über eine begrenzte semantische Bedeutung verfügt, wenn er nicht in ein übergeordnetes Konzept wie das Konzept eines (in einer Anwendungsfunktion verwendeten) Datenpunktes eingebettet wird.

BEISPIEL Der Begriff „Durchfluss“ hat für sich genommen eine schwache semantische Bedeutung, wenn er jedoch in einem FOI mit dem Begriff „Wasser“ in Verbindung gesetzt wird, so kommt zumindest zum Ausdruck, dass der Wasserdurchfluss beobachtet/betätigt wird.

In dieser Spezifikation ist ein Tag beinahe ausnahmslos ein Begriff aus dem semantischen Wörterbuch.

3.1.40

Beschreibung von Dingen en **thing description**

Modell für semantische Metadaten zur Beschreibung (abstrakter oder physischer) Dinge entsprechend der „Thing Description“ (TD) <https://www.w3.org/TR/wot-thing-description/> und der Thing Description Ontology <https://www.w3.org/2019/wot/td>

Anmerkung 1 zum Begriff: TD-relevante Relationen werden in dem Abschnitt zum semantischen Export beschrieben.

3.2 Abkürzungen

Für die Anwendung dieses Dokuments gelten die folgenden Abkürzungen.

DHWC	Steuergerät für das häusliche Warmwasser (en: domestic hot water controller)
FOI	betrachtetes Merkmal (en: feature of interest)
BOC	Kessel-Steuergerät (en: boiler controller)
BUC	Brenner-Steuergerät (en: burner controller)
FTC	Durchflusstemperatur-Steuergerät (en: flow temperature controller)
GA	Gruppenadresse (en: group address)
GO	Gruppenobjekt (en: group object)
FB	Funktionsblock (en: functional block)
FP	Funktionspunkt (en: function point)
ESHG	elektrische Systemtechnik für Heim und Gebäude
HDTRT	Wärmebedarfstransformator Raumtemperatur (en: heat demand transformer room temperature)
HFDM	Wärmestrombedarfsmanager (en: heat flow demand manager)
HIRC	Einzelraumheizungs-Steuergerät (en: heating individual room controller)
HPM	Wärmeerzeugermanager (en: heat producer manager)
HVA	Heizungsstellglied (en: heating valve actuator)
HZC	Heizzone-Steuergerät (en: heating zone controller)
IFC	Industry Foundation Class
IOO	Info ein/aus (en: info on/off)
IO	Eingang/Ausgang (en: input/output)
IoT	Internet der Dinge (en: Internet of Things)
KIM	von der KNX Association verwaltetes ESHG-Informationsmodell
KNXA	KNX Association
MaC	Management-Client
OP	Objekteigenschaft (en: object property)
QK	Eigenschaftsart (en: quality kind)
RSM	Raumsollwert-Manager (en: room setpoint manager)
RTC	Raumtemperatur-Steuergerät (en: room temperature controller)
RTS	Raumtemperatursensor (en: room temperature sensor)
SOO	Schalter ein/aus (en: switch on/off)
TD	Beschreibung von Dingen (en: thing description)

4 ESHG-Informationsmodell

4.1 Hintergrund und aktuelle Lage

Die derzeitigen ESHG-Modell-/Dateninformationen beruhen auf XML, werden von der KNX Association verwaltet und verfügen über die entsprechende Versionierung. Projekt-/Produktdaten, die mit bestehenden ESHG-Management-Clients exportiert werden, stehen im Einklang mit diesem XML-Schema.

Die ESHG-Management-Clients verwenden das XML-Schema in erster Linie, um die entsprechenden Datenstrukturen zum Speichern/Exportieren von Projekt- und Produktdaten zu definieren. Dementsprechend wird das XML-Schema jedes Mal aktualisiert, wenn eine neue Version von ESHG-Management-Clients veröffentlicht wird, und neue Projekt- und/oder Produkteigenschaften erfordern eine Änderung der Datenstrukturen.

Wenngleich XML selbst ein bekanntes, weit verbreitetes Format ist, hat es im Hinblick auf die folgenden Punkte doch einige Nachteile:

- Freigeben von Modell-/Projektdateninformationen für externe Clients, die diese Daten verwenden; wenn eine neue XML-Schema-Version angekündigt wird, muss hier eine Synchronisierung erfolgen;
- Beschreibung, Ausdruck, Abbildung und Freigabe von (semantischen) Informationen in der IoT-Domäne.

Ziel und Motivation ist es, ein ESHG-Informationsmodell und ein zugehöriges Datenaustauschformat zu definieren:

- Das Modell bringt nur die aktuellen – von externen Clients angeforderten – Informationen zum Ausdruck.
- Das Modell lässt sich zudem einfach aktualisieren.
- Die exportierten Daten verwenden ein weit verbreitetes Datenaustauschformat, das auch für Menschen lesbar sein sollte, also ein textbasiertes Format.

Dieses Modell- und Datenaustauschformat ist stabiler als die ständige Weiterentwicklung von ESHG-Management-Clients.

- Das ESHG-Informationsmodell wird als Ontologie in einem oder mehreren Formaten wie z. B. Turtle-Dateien zur Verfügung stehen.
- Die von den ESHG-Management-Clients exportierten Daten werden als verknüpfte Daten wie z. B. JSON-LD-Dateien zur Verfügung stehen.

Die ESHG-IoT-Protokoll-Suite muss semantische Informationen für die Laufzeit und für die Konfiguration unterstützen.

Diese Informationen müssen den Systemkomponenten auf einem datengestützten Weg übermittelt werden, von der ESHG-Management-Client-Software und möglichen anderen Quellen. Sie müssen also auf den vom Nutzer des ESHG-Management-Clients bereitgestellten Informationen aufbauen, damit sie nicht bei der Drittanbieter-Client-Konfiguration erneut eingegeben werden müssen.

Die semantischen Informationen müssen dem ESHG-Standard entsprechen und öffentlich verfügbar sein. Technisch betrachtet werden diese Informationen als verknüpfte Daten zur Verfügung stehen, ausgedrückt als Tripel.

Die Verwendung von Semantik ermöglicht die Formalisierung, Beschränkung und Überprüfung der Verwendung von ESHG-Subjekten zur Beschreibung von Entitäten, Relationen und Tags (Entitäten verfügen über ein Relationsprädikat zu anderen Entitäten – im Grunde eine gerichtete Kante in einem Graphen).

Diese einfache (Tripel-)Struktur ermöglicht eine prägnante und elegante Zusammensetzung großer, miteinander verbundener Strukturen von Untersystemen von Einrichtungen (Gebäuden).

Das KIM muss in der Lage sein, Beziehungen in einer Installation auszudrücken, so dass die folgenden Anwendungsfälle für das Anfordern/Abfragen von Daten unterstützt werden.

- Wie ist ein Bauwerk strukturiert (Stockwerke, Räume usw.)?
- Wo an einem Ort (Ort des Zusammenbaus) befindet sich eine Ausrüstung (z. B. ein Gerät)?
- Welche übergeordnete Funktion wird auf einem Gerät gehostet?
- Welche Schnittstelle (z. B. ein Punkt) wird von einer Funktionalität oder einem Gerät (Kanal, FB) bereitgestellt?
- Welche Anwendungsfunktion betrifft welche baulichen Elemente wie z. B. einen Raum?
- Was beobachtet/betätigt eine Anwendungsfunktion oder ein Punkt in der realen Welt (Stichwort → betrachtetes Merkmal mit Stoff, Phänomen und Ausrüstung)?
- In welcher Betriebsdomäne oder für welche Sparten arbeitet eine Anwendungsfunktion (Stichwort → Zuweisung von Sparten für Anwendungsfunktionen)?
- Welche Punkte gehören zu einer Anwendungsfunktion und was ist der Zweck des Punktes (Stichwort → Tag-Informationen: *Logischer Eingang / Logischer Ausgang, Sollwert oder Parameterpunkt*).

- Über welchen Außenleiter wird der Verbraucher (z. B. *Leuchte*, *Motor*) versorgt?
- Was sind die Soll- und Isttemperaturwerte an einem bestimmten Ort (Raum, Zone) / einer Reihe von Orten?
- Welche aktuelle Betriebsart (Heizen, Kühlen, Belüften, Warmwasserversorgung) weisen bestimmte Primärsysteme allgemein oder am Ort x auf?
- Wie lautet der Bezeichner eines bestimmten Heizkreises (= Verbindung von der Wärmequelle bis zur Wärmesenke) und in welchem Heizkreis befindet sich ein bestimmter Heizkörper?

ANMERKUNG Heizkreis = Es kann ausgedrückt werden, dass n Punkten eine Lokalität „Circuit“ (Kreis) und ein Phänomen „Heat“ (Wärme) zugeordnet ist, dass sie über verschiedene Geräte verteilt sind, sich an verschiedenen Orten befinden, für verschiedene Anwendungsfunktionen arbeiten und unter einem Aspekt gruppiert sind.

- Welchen Stromverbrauch weist ein bestimmtes Gerät auf, das über Punkte verfügt, die für eine *Leuchte* / *Klimaanlage* / einen *Kühlschrank* arbeiten?
- Welche Funktionspunkte steuern eine bestimmte *Leuchte* / einen bestimmten *Heizkreis* (siehe oben) usw.?
- Welche Betriebsdauer weisen bestimmte *Leuchten* zum aktuellen Zeitpunkt auf? Hinweis: Der Client kann diese Dauer anschließend mit dem Betriebsdauerschwellwert für die Instandhaltung vergleichen, der nicht im KIM ausgedrückt wird.
- Welche Funktionsinformationen, die in den Tag-Informationen ausgedrückt werden, werden von einem Punkt repräsentiert oder sind diesem zugewiesen?

Beispiele können 4.4.2.5.2 oder 4.5.1.4 entnommen werden.

4.2 Einleitung

4.2.1 Allgemeines

Das ESHG-System ist für den direkten Austausch von Informationen (d. h. für *Kommunikation*) zwischen über ein Netzwerk verbundenen *Geräten* vorgesehen, die Anwendungen in und um Gebäude bzw. *Orte* steuern (siehe 3.1).

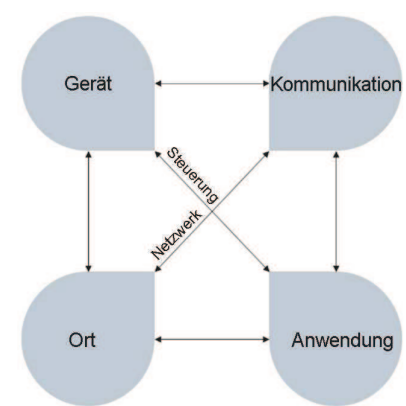


Bild 1 – Elemente eines ESHG-Systems

Um das gesamte ESHG-System zu beschreiben, werden mehrere Modelle benötigt, die die verschiedenen Aspekte des Systems abdecken:

1. Produktmodell

- für die Modellierung von Produkten mit ihrem typ-/anwendungsbezogenen Verhalten, ihrer Konfiguration und ihren Kataloginformationen, unabhängig von der tatsächlichen Verwendung in einer Installation;
- nutzt in erster Linie das Geräte- und das Tag-Modell aus diesem Dokument.

Die Modelldaten werden umgangssprachlich auch als (nicht instanziierte) **Katalogdaten** bezeichnet.

BEISPIEL 1 Gerätemodellaspekt → Speicherbedarf von Konfigurationsparametern und Datenpunkten

2. Systemmodell

- für die Modellierung von Elementen und Abhängigkeiten der physikalischen Vorgänge, siehe „Kommunikation“, (über ein Netzwerk verbundene) „Geräte“ und „Anwendung“ in Bild 1 oben;
- nutzt in erster Linie das Geräte- und das Anwendungsfunktionsmodell aus diesem Dokument. Nutzt außerdem einige Bestandteile des Installationsmodells, wie z. B. Punkte.

BEISPIEL 2 Topologiemodellaspekt → die physische Netzwerkconfiguration eines Gerätes

3. Installationsmodell

- für die Modellierung von Elementen und Abhängigkeiten, die instanziierten Daten einer realen Installation entsprechen; nutzt das Produktmodell (als Eingabe) und verhält sich wie das (und entsprechend dem) Systemmodell;
- nutzt in erster Linie das Orts-, das Anwendungsfunktions- und das Tag-Modell aus diesem Dokument. Die Modelldaten werden umgangssprachlich auch als **Instanzdaten** bezeichnet.

BEISPIEL 3 Ortsmodellaspekt → die tatsächliche räumliche bauliche Struktur einer Installation, die unabhängig von dem ESHG-System ist. Dieselbe Installation könnte aus technischer Sicht auch in einem anderen Bauwerk betrieben werden / kommunizieren, auch wenn das ursprünglich angestrebte Ziel dort nicht notwendigerweise erreicht werden würde.

Hinweis: In diesem Dokument wird in erster Linie das Installationsmodell behandelt, derjenige Aspekt einer Installation, der in einem Projekt eines Management-Clients dokumentiert wird, mit instanziierten ESHG-Geräten oder -Funktionen usw.

4.2.2 KIM – Inhalt

Der Inhalt des KIM baut auf den ESHG-Definitionen, aber auch auf anderen Arbeiten auf:

- er wird, wo zutreffend, durch Haystack-Tags erweitert;
- die Darstellung der KIM-Klasse :Point oder :Equipment sowie deren Relationen ist an W3C-Konzepte angelehnt;
- er nutzt die Ortskonzepte („Location“) der IFC.

Komponenten

Aus technischer Sicht baut das KIM auf verschiedenen Informationsquellen auf, nämlich auf ESHG-relevanten Definitionen und – wie vorstehend beschrieben – auf verwandtem externem Inhalt, wobei Letzterer unter 4.2.6 ausführlich beschrieben wird.

Die ESHG-relevanten Definitionen wiederum setzen sich aus zwei logischen Teilen zusammen, dem ESHG-System und dem semantischen Wörterbuch. Diese Trennung ist erforderlich, um einen statischen (versionierten) mit dem ESHG-System zusammenhängenden Teil der KIM-Ontologie ausdrücken zu können, der sich nicht ändert, wenn Inhalte in dem Teil mit dem semantischen Wörterbuch hinzugefügt oder aktualisiert werden. Dieses Verhalten findet auch in der Verwendung unterschiedlicher Namensräume Ausdruck. Bezüglich Einzelheiten zu diesem Aufbau wird auf 4.2.8 verwiesen.

KIM = ESHG-Domänen-Modell (bestehend aus externem Inhalt + ESHG-System) und ein zugehöriges semantisches Wörterbuch

Dieses Prinzip ist in Bild 2 dargestellt.

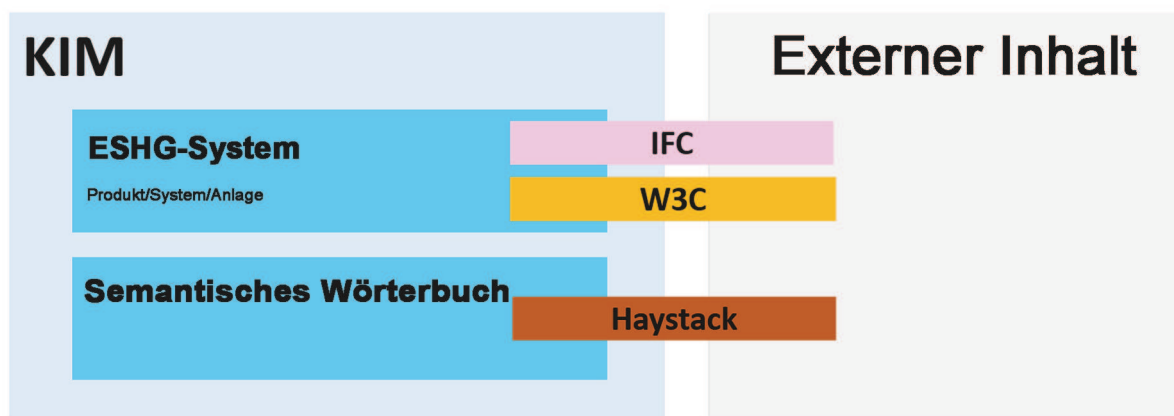


Bild 2 – Hauptquellen des KIM

Wie in 3.1 festgelegt, handelt es sich bei dem KIM um eine Ontologie; in einer Ontologie wird für jeden einzelnen Aspekt eine Ontologiekategorie festgelegt. Eine Kategorie wird in einer Ontologie auch als Konzept bezeichnet.

BEISPIEL Die/das Ontologiekategorie/-konzept :Room spiegelt den Aspekt „verfügbare Räume in einer Installation“ wider. Der Aspekt „Punkte von Geräten“ wird durch die/das Ontologiekategorie/-konzept :Point dargestellt.

ESHG-System:

- definiert das Konzept-Framework;
- setzt sich aus dem vorgenannten System-, Installations- und Produktmodell zusammen, wird jedoch nicht direkt aus diesen zusammengesetzt; vielmehr handelt es sich um eine durch Zusammenführung geschaffene Kombination dieser Modelle;
- wird von dem semantischen Wörterbuch genutzt;
- spiegelt den statischen Teil des KIM wider, die Konzepte sind ein feststehender Bestandteil des KIM;
- stellt Relationen/Abbildungen spezifischer KIM-Konzepte zu/auf andere(n) Ontologien her, siehe diesbezüglich 4.2.6.

Semantisches Wörterbuch:

- definiert die Konzept-Framework-Daten;
- bringt semantische Begriffe zum Ausdruck; eine allgemeine Definition kann 3.1, Einzelheiten 4.2.8 entnommen werden;
- spiegelt den dynamischen Teil des KIM wider, der **Inhalt** kann ergänzt/aktualisiert werden.

In Bild 3 sind die wichtigsten Ontologiekategorien des KIM für die beiden vorstehend beschriebenen Teile dargestellt.

- | | |
|------------------------|--|
| — :Aspect | ein bestimmter Blickwinkel auf / Aspekt von dem System; zur Definition siehe 3.1 |
| — :Equipment | repräsentiert materielle Hardware wie Schränke, Geräte usw. |
| — :ApplicationProgram | gleichbedeutend mit Anwendungsprogramm, siehe 3.1 |
| — :Point | Schnittstelle für die Interaktion mit Datenpunkten – Bereitstellung in erster Linie durch Geräte |
| — :Functionality | Gruppierung von Punkten als Bestandteil eines Anwendungsprogramms bzw. Gerätes |
| — :ApplicationFunction | bauliches Element, das eine vorgesehene Anwendung zum Ausdruck bringt |

- :Location Wurzelement von baulichen Konzepten
- :Tag Wurzelement der meisten Begriffe im semantischen Wörterbuch

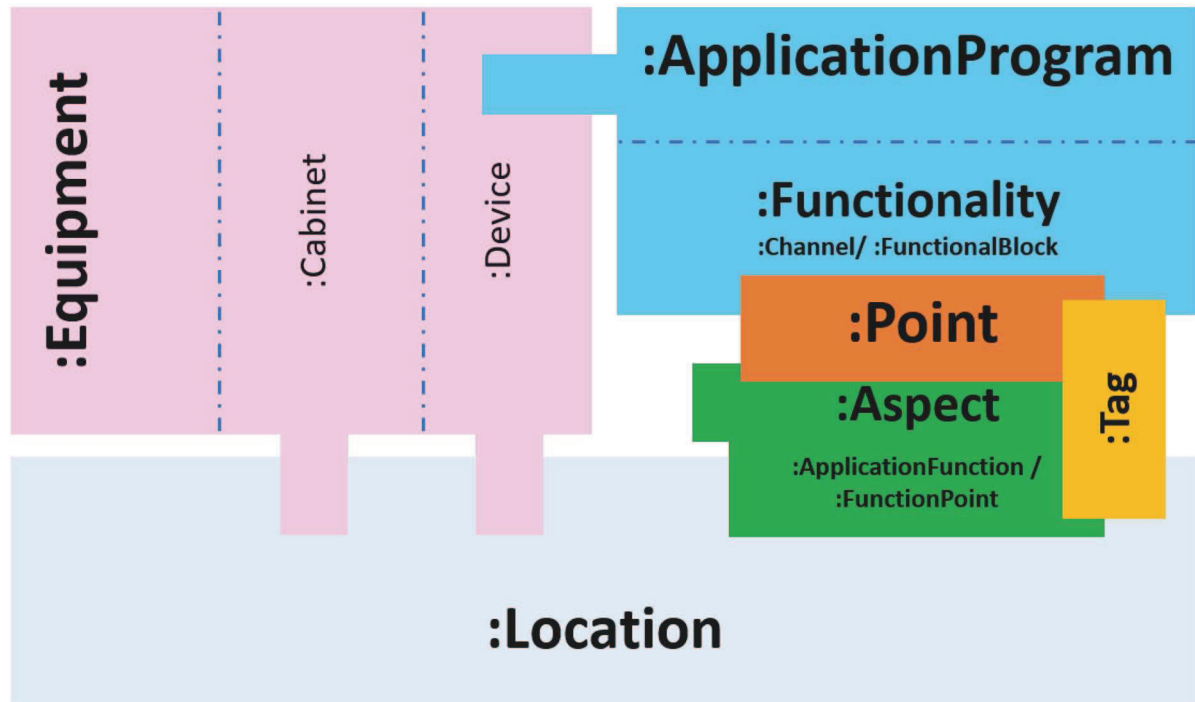


Bild 3 – Wichtigste KIM-Ontologieklassen

4.2.3 KIM – Version

Um zwischen Aktualisierungen des KIM unterscheiden zu können, benötigt dieses explizite Versionierungsangaben. Die einzige Anforderung an eine neue/aktualisierte (KIM-)Ontologie besteht darin, dass der IRI der Ontologieversion ein anderer sein muss.

Die Versionierung der KIM-Ontologie folgt den Empfehlungen des W3C für die OWL-Versionierung (siehe https://www.w3.org/TR/owl2-syntax/#Versioning_of_OWL_2_Ontologies). Üblicherweise wird der IRI aus dem Versionsbezeichner, einem für Menschen lesbaren Text und einem zusätzlichen Zeitstempel zusammengesetzt.

Der IRI der KIM-Ontologie einschließlich des Versionsbezeichners wird in 4.2.6 definiert.

4.2.4 KIM –Verfügbarkeit

Es wird erwartet, dass mithilfe des unten angegebenen IRI der KIM-Ontologie die neueste Ontologie als elektronische Datei abgerufen werden kann, indem der IRI als URL verwendet wird.

Um eine bestimmte Version der KIM-Ontologie abzurufen, muss stattdessen der IRI der entsprechenden Version der KIM-Ontologie als URL verwendet werden.

- <https://schema.knx.org/2020/ontology> gibt die neueste Version der KIM-Ontologie zurück, und zwar als vollständige Ontologie mit allen festgelegten Ontologieteilern wie dem Ortskonzept, dem ESHG-System-Konzept oder semantischen Wörterbüchern der KNX und etwaiger KNX-Hersteller (siehe Tabelle unten);
- <https://schema.knx.org/2020/ontology/v1> gibt die Version v1 der KIM-Ontologie zurück, wenn der oben angegebene Ontologie-IRI, ergänzt um die benötigte Version (in diesem Beispiel „v1“), verwendet wird.

4.2.5 KIM – Datenformat und Datenaustauschformat

Das KIM steht in einem Format für verknüpfte Daten wie TURTLE oder RDF/XML zur Verfügung.

Das KIM tauscht seine Daten im Rahmen des und unterstützt von dem semantischen Export aus.

Der semantische Export verwendet ein weit verbreitetes Datenaustauschformat, das auch für Menschen lesbar sein sollte und also textbasiert ist. Es steht in Form von verknüpften Daten wie als JSON-LD zur Verfügung.

4.2.6 KIM – Ontologie-IRIs und –Namensräume

Der IRI der KIM-Ontologie wird entsprechend dem folgenden (vereinfachten) Prinzip festgelegt:

$$\{\text{Schema}\} + „/“ + \{\text{Organisation}\} + „/“ + \{\text{Pfad}\} + „/“ + \{\text{Version}\} + „\#“$$

Über den oben angegebenen Teil aus {Schema} und {Organisation} wird der IRI der KIM-Ontologie definiert als „https://schema.knx.org/2020/ontology“.

Der {Pfad} ist eine Zeichenkette, die einzelne KIM-Konzepte beschreibt und auch einen eigenen Namensraum verwendet; siehe diesbezüglich nachstehende Tabelle.

Die {Version} muss eine Zeichenkette aus „v“ und einer nicht mit „0“ beginnenden Dezimalzahl sein, beispielsweise „v2“ oder „v249“. Dieser Versionierungsansatz lässt demnach nur die Benummerung von Hauptversionen der Ontologie zu.

Eine weitere Anforderung an das KIM ist die Übersetzung relevanter KIM-Konzepte in andere (externe) Ontologien zum Zwecke einer einfachen semantischen Abbildung. Um diese Anforderung zu erfüllen, wird vorrangig die Relation zwischen OWL-Klassen und -Unterklassen oder eine explizite Referenz verwendet (es gibt jedoch noch weitere Möglichkeiten). Im Folgenden werden die wichtigsten Prinzipien erläutert, sie gelten auch für die anderen Konzepte.

1. KIM-Ortskonzepte mit der gleichen konzeptuellen Bedeutung wie spezifische IFC-Ortskonzepte, dies wird durch die Verwendung der Referenz *owl:sameAs* ausgedrückt.

Diese Referenz definiert keine Abbildung der OWL-(Unter-)Klasse auf das entsprechende IFC-Konzept, sie bringt lediglich zum Ausdruck, dass **IFC-Konzepte auf** ihre semantischen **ESHG-Konzeptentsprechungen** abgebildet bzw. in diese übersetzt werden können.

- Die ESHG-Konzepte selbst müssen nicht notwendigerweise alle Daten des IFC-Konzepts zum Ausdruck bringen. Wenn allerdings ein ESHG-Konzept Informationen benötigt, die **nicht** (weder direkt noch indirekt) für das entsprechende IFC-Konzept benötigt werden, ist eine Übersetzung von IFC zu ESHG aufgrund der fehlenden Informationen nicht möglich.

2. KIM-Datenpunkttypkonzepte sind Unterklassen eines JSON-Schema-Konzepts, beispielsweise ist *dic-tag:dpt.1.x* eine Unterklasse von *jsonschema:BooleanScheme*.

Dies bringt zum Ausdruck, dass ein **ESHG-Datenpunkttyp (DPT) auf** seine semantische **JSON-Datenschema-Konzeptentsprechung (Typ)** abgebildet bzw. in diese übersetzt werden kann.

- Ein ESHG-DPT ist zugleich ein JSON-Typ. Der ESHG-DPT erbt die Merkmale des JSON-Typs, kann jedoch Ergänzungen vornehmen. Ein ESHG muss also mindestens die Informationen erfordern, die auch ein JSON-Typ erfordert.

Die folgende Tabelle 1 enthält die in KIM verwendeten Namensräume, ihre IRIs und ihre Ontologiereferenz.

Tabelle 1 – KIM-Namensräume, zugehörige IRIs und Ontologiereferenz

Ontologienname	Namensraumbezeichner in KIM	IRI ^a	Bemerkung
JSON-Schema	jsonSchema:	https://www.w3.org/2019/wot/json-schema#	OPs und OWL-Klassen zum Ausdrücken von JSON-Typ/-Format und -Daten
QUDT	qudt:	http://qudt.org/schema/qudt/	OWL-(SI-)Einheitenklasse für FPs und Datenpunkte
	qudt-unit:	http://qudt.org/vocab/unit/	Individuen zum Ausdrücken von (SI-)Einheiten
	qudt-quantity:	http://qudt.org/vocab/quantity-kind/	Individuen zum Ausdrücken von Größen-Tags
VCARD	vcard:	http://www.w3.org/2006/vcard/ns#	OPs und OWL-Klassen zum Ausdrücken von Geo-Adressen
DCTERMS	dcterms:	http://purl.org/dc/terms	OPs für Beschreibung/Titel
HTTP	http:	http://www.w3.org/2011/http#	OPs für den URL-Organisations-Teil
KIM ^b	loc:	https://schema.knx.org/2020/ontology/loc{/Version}#	ESHG-Ortskonzepte {Pfad} = /loc
	knx:	http://schema.knx.org/2020/ontology/knx{/Version}#	ESHG-Systemkonzepte {Pfad} = /knx
	dic:	http://schema.knx.org/2020/ontology/dictionaries/dic{/Version}#	ESHG-Inhalt des semantischen Wörterbuchs, Ausdruck von Konzepten, die keine Individuen beinhalten. Siehe 4.2.8.4. {Pfad} = dictionaries/dic
	dic-tag:	http://schema.knx.org/2020/ontology/dictionaries/dic-tag{/Version}#	ESHG-Inhalt des semantischen Wörterbuchs, Ausdruck von Konzepten, die Individuen beinhalten. Siehe 4.2.8.4. {Pfad} = dictionaries/dic-tag
	–	https://schema.knx.org/2020/ontology	Über den Ontologie-IRI wird die vollständige Ontologie abgerufen, siehe 4.2.4.
	<u>Beispiel</u> abb: gira: ...	<u>Beispiel</u> http://gira.de/2022/dictionaries/iotfunctions{/Version}#	Inhalt des semantischen Wörterbuchs, der Inhalte eines KNX-Herstellers widerspiegelt. Siehe 4.2.8.3.3.
^a Die IRIs müssen nicht notwendigerweise dereferenziert werden, wie dies bei einer URL der Fall ist, um auf Inhaltserläuterungen einer Website zuzugreifen. Bezüglich des Abrufens von KIM-Inhalten siehe 2.2.3. ^b Aus Gründen der Lesbarkeit beziehen sich alle KIM-Ontologieklassen in diesem Dokument auf den Namensraum :knx, sofern kein expliziter Namensraumbezeichner angegeben ist.			

4.2.7 KIM – Ontologieklassen

Wie in 4.2.2 erläutert, wird der gesamte KIM-Inhalt einschließlich der ESHG-Systemaspekte als Ontologieklassen modelliert. Im KIM ist die folgende Struktur aus Klassen und Unterklassen definiert.

- owl:Thing
- jsonSchema:DataScheme
- knx:Thing
 - ...
 - knx:Point
 - ...
 - knx:Location
 - ...
- qudt:Unit
- vcard:Address

Alle mit ESHG zusammenhängenden Ontologieklassen sind unter dem Knoten *knx:Thing* angeordnet, der Knoten hat keine bestimmte semantische Bedeutung.

Er wurde eingeführt, um ESHG-Ontologiekonzepte klar von anderen verwendeten (externen) Ontologien (strukturellen ebenso wie UI-bezogenen) zu trennen. Hintergrund ist, dass einige ESHG-Ontologiekonzepte Relationen zu anderen Ontologien benötigen, siehe 4.2.6.

Die Knoten *knx:Thing*, *qudt:Unit*, *vcard:Address* und *jsonSchema:DataScheme* dürfen nicht disjunkt sein, andernfalls kann ein Datenpunkt, der Teil von *knx:Thing* ist, keine Unterklasse von *jsonSchema:BooleanSchema* sein.

Wurzelklassen eines *spezifischen* Konzepts in einer Ontologie – wie beispielsweise alle mit dem ESHG-System zusammenhängenden Ontologieunterklassen auf der ersten Ebene unter dem Knoten *knx:Thing* – sind in fast allen Fällen disjunkt; im KIM sind sie disjunkt.

ANMERKUNG In der Mathematik werden zwei Klassen als „disjunkt“ bezeichnet, wenn die Schnittmenge der beiden Mengen, die sie beschreiben, leer ist. In der Logik bzw. OWL bedeutet die Zugehörigkeit zu einer von zwei disjunkten Klassen, dass keine Zugehörigkeit zu der anderen Klasse besteht.

4.2.8 KIM – semantisches Wörterbuch

4.2.8.1 Allgemeines

Wie in 4.2 erläutert, werden die erforderlichen ESHG-Systemaspekte im KIM mithilfe von Ontologieklassen ausgedrückt. Es muss sichergestellt werden, dass alle diese als Ontologie ausgedrückten Aspekte in unterschiedlichen Domänen und insbesondere von allen Personen, die an der (Vor-/Nach-)Bearbeitung dieser Informationen beteiligt sind, semantisch einheitlich verstanden und interpretiert werden.

Hierzu wird ein Wörterbuch mit semantisch vordefinierten Begriffen benötigt. Dieses Wörterbuch erfüllt zwei wesentliche Aufgaben.

1. Es beinhaltet Begriffe zum Annotieren der vorgenannten Aspekte, wobei jeder Begriff bestimmte Domäneninformationen enthält.
2. Es kann um neue Begriffe ergänzt werden oder die vorhandenen Begriffe können aktualisiert werden, jeweils ohne dass dies den ESHG-System-Teil des KIM ändert/betrifft (weil dadurch bei jeder Aktualisierung oder Änderung eine neue KIM-Version veröffentlicht werden müsste). Der Mechanismus zum Aktualisieren des Wörterbuchs wird in 4.2.8.5 beschrieben.

4.2.8.2 Aufbau des Wörterbuchs

- Während die Begriffe in einem Lexikon alphabetisch geordnet sind, sind die Begriffe in einem semantischen Wörterbuch:
 - nicht alphabetisch, sondern nach ihrem Zweck geordnet;
 - sofern zutreffend, in einer Taxonomie organisiert, wobei jeder Abschnitt einer Taxonomie eine andere relevante Domäne beschreibt (beispielsweise ein Abschnitt mit allen Protokollen, ein Abschnitt mit allen Sparten usw.);
 - über mehrere KIM-Ontologieklassen/-unterklassen verteilt.
- Eine Taxonomie lässt sich mit einer erweiterbaren Struktur mit unterschiedlichen Abschnitten und optional Unterabschnitten vergleichen. Ergebnis ist die Definition semantisch bedeutsamer Abschnitte, die in manchen Fällen nur einen Individuen-Begriff enthalten.

BEISPIEL Ein Beispiel für eine Taxonomie ist die Klassifikation von **feline** (Katzen) (Abschnittsebene der Familie), die weiter in **wildCat** (Wildkatzen) und **petCat** (Hauskatzen) (optionale Abschnittsebene der Art) unterteilt werden. Der Unterabschnitt **petCat** wiederum enthält Individuen wie „Tom“ oder „Garfield“. Hinweis: Dass zu der (menschengemachten) Bedeutung von „Garfield“ die Zuschreibung eines bestimmten Gerichts gehört, kann durch eine (Objekteigenschaft-)Relation zu Nahrungsmittelbegriffen im Wörterbuch ausgedrückt werden.

4.2.8.3 Inhalt des Wörterbuchs

4.2.8.3.1 Allgemeines

Das Wörterbuch ist der dynamische Teil des KIM (siehe 4.2.2).

Das Wörterbuch enthält Begriffe zum Annotieren einer Installation mit semantischem Inhalt, genauer gesagt Begriffe zum Annotieren der folgenden KIM-Konzepte:

- Punkte (en: points)
- Aspekte (en: aspects), diese enthalten Anwendungsfunktionen und Funktionspunkte
- Orte (en: locations)
- Ausrüstung (en: equipment)

Hinsichtlich der Namenskonvention für Wörterbuchbegriffe im KIM muss Folgendes beachtet werden: Jeder Begriff im KIM verfügt über einen Ontologienamen, kann aber auch ein Label (in Textform) haben. Für Nutzer der Ontologie – einschließlich Software – wird der Begriff in Form dieses Labels dargestellt. Für ein und dasselbe Konzept kann es im KIM mehrere Labels geben, die in verschiedene Sprachen übersetzt und mit einem Sprachbezeichner versehen sind. Zu den Beispielen in diesem Dokument wird jeweils der Label-Name angegeben.

BEISPIEL Der Wörterbuchbegriff des Konzepts „Größe“ (en: quantity) ist ein Individuum mit dem Namen dic-tag:flow, das Label lautet „Durchfluss“.

Das Wörterbuch enthält ESHG-Inhalte sowie Inhalte von einem KNX-Hersteller. Für beide Fälle wird jeweils ein eigener Namensraum verwendet, siehe 4.2.8.3.

4.2.8.3.2 ESHG-Inhalt

Dies ist der gesamte standardisierte Inhalt, der im ESHG-System dank der Standardisierungsarbeit der KNX Association bereits definiert ist.

- Der Inhalt ist ein direkter Bestandteil der KIM-Ontologie und wird von der KNX Association gepflegt.
- Der Inhalt nutzt die definierten KIM-Ontologie-IRIs bzw. die zugehörigen Namensräume für den Inhalt des semantischen Wörterbuchs, entsprechend der Beschreibung in 4.2.6.

4.2.8.3.3 Inhalt von KNX-Herstellern

Hierbei handelt es sich um sämtlichen von einem KNX-Hersteller definierten Inhalt.

Rein technisch betrachtet kann ein KNX-Hersteller herstellerspezifischen Inhalt für alle vorgenannten KIM-Konzepte definieren. Ungeachtet dieser technischen Möglichkeit muss standardisierter ESHG-Inhalt wie Funktionsblöcke, annotierte/typisierte Punkte oder Tags in dem Wörterbuch im Rahmen des Standardisierungsprozesses von der KNX Association verwaltet werden.

- Ein KNX-Hersteller kann in der Klasse :ApplicationFunction oder in einer von deren Unterklassen seine eigenen Anwendungsfunktionen definieren. Dabei muss das in 4.4.2.3.2.4 beschriebene Namensschema für die Definition neuer Anwendungsfunktionen berücksichtigt werden. Außerdem wird dringend empfohlen, den Herstellerinhalt auf verfügbare Anwendungsfunktionen des semantischen Wörterbuchs aufzubauen.
- Der KNX-Hersteller-Inhalt bildet eine eigene, vom KNX-Hersteller erstellte Ontologie, die von der KNX Association gemeinsam mit allen anderen Teilen der KIM verwaltet wird.
- Für den Herstellerinhalt wird ein eigener IRI verwendet, der disjunkt zu den IRIs sein muss, die im KIM für bereits definierten ESHG-Inhalt verwendet werden (siehe 4.2.8.3.2).

ANMERKUNG Das Referenzieren von einer (eigenen KNX-Hersteller-)Ontologie auf eine externe (KIM-)Ontologie stellt ein standardmäßiges OWL-(Abbildungs-)Verfahren dar. Als Beispiel: Definitionen von KNX-Hersteller-spezifischen Anwendungsfunktionen referenzieren mindestens auf die generische KIM-Klasse :ApplicationFunction; aussagekräftiger ist eine Zuweisung zu einer der in der KIM-Klasse verfügbaren spezifischeren Anwendungsfunktionen wie :lightingCtrl.

- Der KNX-Hersteller-Inhalt muss für seinen jeweiligen IRI einen entsprechenden Namensraumbezeichner für den Hersteller ausdrücken, der aus einem eindeutigen KNX-Hersteller-Namen und „:“ am Ende bestehen muss.

Der Name des KNX-Herstellers darf nicht mehr als 10 Zeichen umfassen und darf nur aus ASCII-Zeichen bestehen, wobei „:“ ausgenommen ist.

Der Namensbezeichner darf nicht mit anderen Namensraumbezeichnern, einschließlich jener für von der KNX Association standardisierten Inhalt, in Konflikt stehen.¹

BEISPIELE

- | | | | |
|---|------------|---------|------------|
| — | Hersteller | SIEMENS | → siemens: |
| — | Hersteller | GIRA | → gira: |
| — | Hersteller | ABB | → abb: |

4.2.8.3.4 URN-Schema für Hersteller- und ESHG-Inhalt

Hersteller- und ESHG-Inhalt kann direkt als IRI oder als URN-basierter Ressourcenbezeichner ausgedrückt werden (Beispiele für eine IoT-Funktion können 4.4.2.3.2.3 entnommen werden). Der URN muss wie folgt formatiert werden:

„urn:“ + „knx:“ + \$TYPE + \$NAME

Die Teile „urn:“ und „knx:“ sind unveränderlich; „urn:“ ist der – in RFC – standardisierte URN-Namensraumbezeichner und „knx:“ ist der offizielle KNX-Namensraumbezeichner.

\$TYPE ist ein Bezeichner, der den Typ des Inhalts des nachfolgenden \$NAME-Teils beschreibt. Die folgenden \$TYPE-Bezeichner sind definiert:

- mft. Das nächste Element hinter \$TYPE muss ein Bezeichner für den Namen eines KNX-Herstellers sein. Dieser Typbezeichner muss immer dann verwendet werden, wenn KNX-Mitglied-spezifischer Inhalt ausgedrückt wird. Für von der KNX standardisierten Inhalt darf dieser \$TYPE-Bezeichner nicht verwendet werden.

Der Name des KNX-Herstellers muss der Name sein, der auch für den KNX-Hersteller-Namensraum

¹ Die Namensraumbezeichner kommen auch zum Einsatz, wenn KNX-Hersteller-spezifische API-Referenz-Mitglieder ausgedrückt werden, siehe dazu 3.4.5.6.1.

verwendet wird, ohne den Doppelpunkt („.“). Am Ende des Namens des KNX-Herstellers wird „.“ angehängt.

- urn:knx:fct.switching // eine von der KNXA standardisierte Anwendungsfunktion
- urn:knx:dpt.switch // ein von der KNXA standardisierter DPT (typisierter Punkt)
- urn:knx:mft.siemens.switching // eine von einem KNX-Mitglied standardisierte Anwendungsfunktion
- fct. Das nächste Element hinter \$TYPE muss ein Bezeichner für eine Anwendungsfunktion sein.
- dpt. Das nächste Element hinter \$TYPE muss ein Bezeichner für einen typisierten Punkt sein.
- dpa. Das nächste Element hinter \$TYPE muss ein Bezeichner für einen annotierten Punkt sein.

Der \$TYPE-Bezeichner darf nicht mehr als 5 Zeichen umfassen (einschließlich des „.“) und darf nur aus ASCII-Zeichen bestehen.

Der \$NAME-Teil kann frei gewählt werden, wobei die Benennungsregeln zu beachten sind, beispielsweise im Hinblick auf IoT-Datenpunkt-Namen als Teil des standardisierten ESHG-Inhalts oder auf IoT-Funktionen von Mitgliedern. Er darf sich nur aus den unten aufgeführten Zeichen zusammensetzen und höchstens 40 Zeichen umfassen, so dass sichergestellt ist, dass der gesamte URN nicht länger als 64 Zeichen ist.

Zulässige Zeichen aus der IRI- und der URN-Spezifikation, beschränkt auf den folgenden expliziten Zeichensatz:

ALPHA / DIGIT / „-“ / „.“ / „_“ / „~“

Es wird empfohlen, den \$NAME-Teil so zu wählen, dass er eine Länge von 16 Zeichen nicht überschreitet (ausgehend von der Annahme, dass auf einen KNX-Hersteller-Namen 10 Zeichen entfallen), so dass sichergestellt ist, dass der gesamte URN nicht länger als 40 Zeichen ist, wodurch wiederum sichergestellt ist, dass er nicht größer als 40 Byte ist und somit in einer einzelnen UDP-Nutzlast (Nutzdaten) über CoAP untergebracht werden könnte (siehe <https://tools.ietf.org/html/rfc7252#section-4.6>).

Hinweis: Ein URN muss zum Abrufen einer Ressource nicht dereferenziert werden, wie dies bei einer URL und in manchen Fällen auch bei einem IRI – beispielsweise für die KIM-Ontologie – der Fall ist, siehe 4.2.6.

Anwendungsfall

Es ist möglich, in einer bestimmten Version einer Ontologie mithilfe eines (für Menschen lesbaren) URN nach einer Anwendungsfunktion zu suchen, deren IRI möglicherweise kryptisch (nicht lesbar) ist, z. B. dic:x13\$yz. Selbst in unterschiedlichen Versionen einer Ontologie kann nach **ein und derselben** Anwendungsfunktion mit ein und demselben URN gesucht werden.

4.2.8.4 Wörterbuchklassen

4.2.8.4.1 Allgemeines

Wörterbuchinhalte und eine mögliche Taxonomie werden im Allgemeinen mithilfe von Ontologieklassen dargestellt. Der Ansatz im KIM besteht darin, spezifische Ontologieklassen zu definieren, die jeweils eine spezifische Bedeutung im Gebäudebereich haben, wie das Konzept von Punkt-Datentypen (ausgedrückt durch die KIM-Klasse knx:PointType) oder das Konzept (die Klasse) physikalischer Phänomene, die von einem Sensor gemessen werden. Aus diesem Grund kann nicht der gesamte Wörterbuchinhalt direkt unter ein und derselben Ontologiekategorie angeordnet werden.

Der Wörterbuchinhalt wird anschließend dafür verwendet, Relationen/Verknüpfungen zwischen Installations(instanz)daten und dem erforderlichen Wörterbuchinhalt auszudrücken. Installations(instanz)daten dürfen entweder durch die Aussage, dass sie einem bestimmten Wörterbuch(-Ontologieklassen)typ angehören, oder durch die Aussage, dass sie eine(n) OP-Relation/-Verknüpfung zu einem Wörterbuch(-Ontologieklassen)individuum ausdrücken, semantisch definiert werden. Welches Verfahren gewählt wird, hängt von dem vorgesehenen Anwendungsfall ab, deshalb setzt sich das Wörterbuch aus Ontologieklassen mit und ohne Individuen zusammen.

Der gesamte ESHG-Inhalt des Wörterbuchs setzt sich aus den folgenden Ontologieklassen zusammen:

1. Ontologieklassen, die keine Individuen beinhalten
 - :ApplicationFunction
 - :FunctionalBlock
 - :Point
2. Ontologieklassen, die Individuen beinhalten
 - :Tag
 - :QualityKind

4.2.8.4.2 Anwendungsfunktionsklasse (:ApplicationFunction)

Dies ist die oberste Taxonomiekategorie, die das Konzept von Anwendungsfunktionen enthält (siehe 4.4.2.3.2).

Die Klasse verfügt über Unterklassen für die Beschreibung der verfügbaren Anwendungsfunktionen wie standardisierte ESHG-Anwendungsfunktionen oder individuelle Anwendungsfunktionen eines KNX-Herstellers. Hierfür greift sie auf einige der unten beschriebenen Konzepte wie Punkte oder Tags zurück.

4.2.8.4.3 Funktionsblockklasse (:FunctionalBlock)

Dies ist die oberste Taxonomiekategorie, die das Konzept eines Funktionsblocks enthält.

Die Klasse verfügt über Unterklassen für die Beschreibung der verfügbaren FBs wie standardisierte ESHG-FBs. Hierfür greift sie auf einige der unten beschriebenen Konzepte wie Punkte oder Tags zurück.

4.2.8.4.4 Punktklasse (:Point)

Dies ist die oberste Taxonomiekategorie, die das Konzept von Punkten enthält (siehe 4.4.2.8).

Die Klasse verfügt über Unterklassen für die Beschreibung der verfügbaren annotierten Punkte oder typisierten Punkte wie standardisierte ESHG-Punkte, die in Funktionsblöcken verwendet werden, oder Punkte, die mit einem standardisierten ESHG-DPT typisiert werden.

4.2.8.4.5 Tag-Klasse (:Tag)

Dies ist die oberste Taxonomiekategorie, die die Konzepte von Tags enthält, die für Punkte, Funktionspunkte, Anwendungsfunktionen und Orte anwendbar sind (siehe 4.5.5.1).

ANMERKUNG Der Klassenname „:Tag“ wurde gewählt, um eine Harmonisierung der Bezeichnung mit anderen Systemen, in denen Tags bereits etabliert sind (wie z. B. Haystack), zu erzielen.

Die Klasse verfügt über Unterklassen für die Beschreibung verschiedener Arten von Orten oder verschiedener Punkteigenschaften. Mit Ausnahme von Punkttypen und der Klasse :ObjectKind werden die Unterklassen als flache Listen von Begriffen strukturiert.

Unterklassen :Locality, :PointFunctionType, ...

- Mit Tags wird die Semantik von Punkten, Funktionspunkten und Anwendungsfunktionen beschrieben, beispielsweise ob ein Punkt ein Ausgang oder ein Eingang ist.

Unterkategorie :LocationUsage

- Mit Tags wird die Verwendungsart von Orten beschrieben.

4.2.8.4.6 Eigenschaftsartklasse (:QualityKind)

Dies ist die oberste Taxonomiekategorie, die das Konzept einer Eigenschaftsart enthält (siehe 4.4.2.10).

Sie enthält Individuen für die Beschreibung einer Bedeutung tragenden Kombination von Tags, mit der eine Betätigung oder eine Beobachtung durch einen Punkt oder ein FOI mit spezifischen semantischen Daten ausgedrückt wird.

4.2.8.5 Aktualisierung des Wörterbuchs

Der Wörterbuchinhalt kann online ergänzt oder aktualisiert werden.

Voraussetzung für das Hinzufügen von Inhalt ist, dass die Kompatibilität mit dem (statischen) KIM-Teil des ESHG-Systems nicht beeinträchtigt wird. In diesem Fall wäre ein neues KIM mit einer aktualisierten Version erforderlich.

Mithilfe der Annotations-OP-Relation *dic:dictionaryVersion* wird die Version des semantischen Wörterbuchs separat ausgedrückt. Sie verwendet einen (positiven) ganzzahligen Wert und ist bei der Knotenklasse *knx:Thing* angeordnet; für Einzelheiten zu dieser Klasse siehe 4.2.7.

Das semantische Wörterbuch wird als Ganzes versioniert; durch jede Änderung infolge einer Anfrage eines KNX-Herstellers oder durch KNX selbst erhöht sich die Versionsnummer.

4.3 Ortsmodell

4.3.1 Einleitung

Die Ortsontologie enthält eine Formalisierung des Ortsmodells entsprechend der Darstellung in Bild 4.

Notationsstil

- Das blaue Kästchen steht für den obersten generischen Ort (*:Location*), von dem alle Unterklassen abgeleitet werden. Diese Klasse kann als Brückenklasse zu anderen Ontologien verwendet werden.
- Die grün gestrichelten Kästchen stehen für KIM-Klassen, die in Industry Foundation Classes nicht existieren.
- Die grünen Kästchen stehen für KIM-Klassen, die in IFC existieren und abgebildet werden können, siehe dazu 4.2.6.
- Die weißen Kästchen beschreiben Klassen, die unabhängig von den *:Location*-Klassen sind.
- Kästchen mit der Beschriftung „subClassOf“ definieren spezifische (untergeordnete) Klassen von *:Location*.
- Die gestrichelten roten Linien und deren Richtung beschreiben OP-Relationen zwischen Klassen; das hier verwendete OP-Relationsprädikat ist der Name, der der gestrichelten Linie zugewiesen ist.

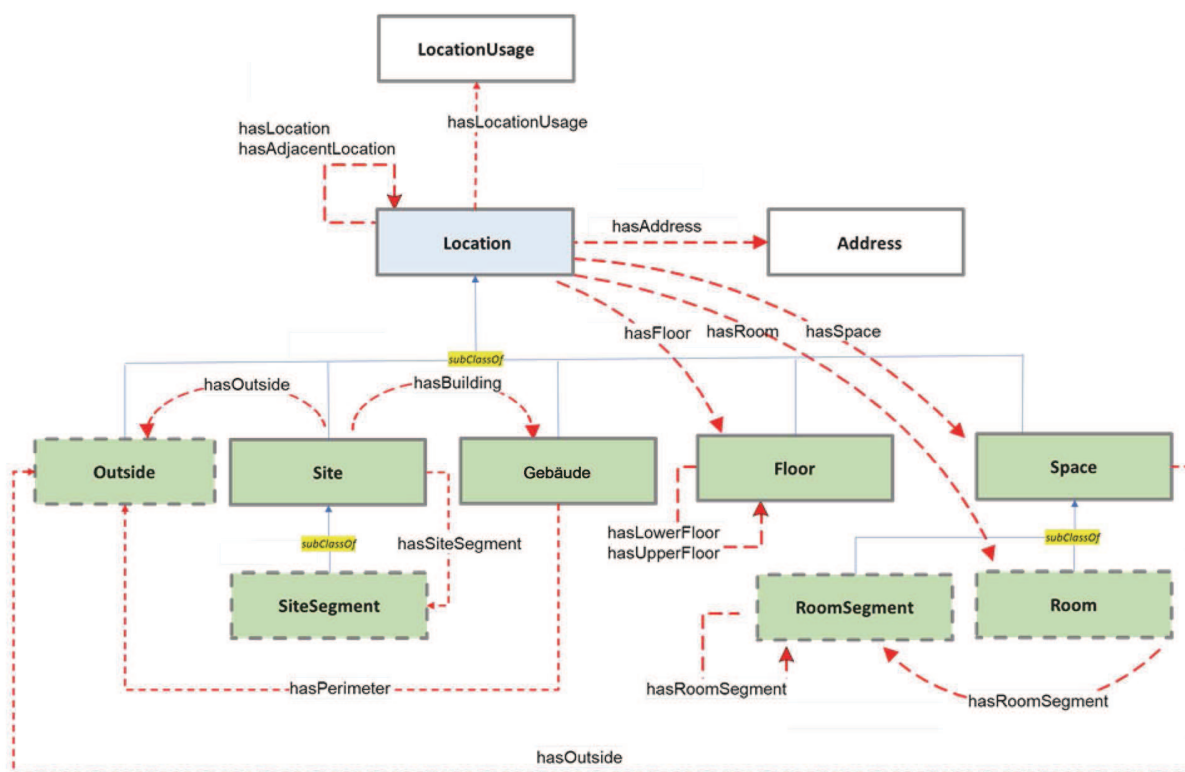


Bild 4 – Ortsmodell

ANMERKUNG 1 Aus Gründen der Lesbarkeit wurde der Namensraumbezeichner „:“ in diesem Bild weggelassen.

Die Klasse *loc:Location* verfügt über die Objekteigenschaften *loc:hasLocationUsage* und *:hasAddress*, über die ihr eine Ortsverwendung bzw. eine Anschrift (mit den Dateneigenschaften Land, Ort, Straßenname und Hausnummer) zugewiesen werden kann.

Auf diese Weise kann bei Bedarf allen Orten, die als Unterklasse von *loc:Location* festgelegt werden, eine Ortsverwendung und/oder Anschrift zugewiesen werden. Üblicherweise wird ein *loc:Site* oder ein *loc:Building* jedoch vorzugsweise mit einer Anschrift beschrieben.

Eine Ortsverwendung für einen Raum wie *Badezimmer* wird möglicherweise vorzugsweise für einen *loc:Room* oder ein *:RoomSegment* verwendet, darf jedoch auch für *loc:Floor* oder *loc:Building* oder sogar *loc:Site* verwendet werden, wenn dieses/dieser einem bestimmten Zweck dient (z. B. *Krankenhaus* o. Ä.).

Die Klassen *loc:Site*, *loc:Building*, *loc:Space* und *loc:Outside* sind die primären Unterklassen von *loc:Location*. *loc:Site* ist üblicherweise am oberen Ende einer Ortshierarchie angeordnet, wohingegen *loc:Space* einen generischen Ort bezeichnet, der einen beliebigen Teil eines Gebäudes bezeichnen kann. Die Klasse *loc:Outside* dient der Beschreibung von Orten, die sich nicht innerhalb, sondern außerhalb eines Gebäudes befinden.

Es stehen zusätzliche *loc:hasOutside*-Eigenschaften zur Verfügung, mit denen außen liegende Orte genauer beschrieben und (bei Bedarf) einem Gebäude zugeordnet werden können.

BEISPIEL 1 Bei außen liegenden Orten wie „Westfassade“ oder „Dach“ ist logischerweise die Zuweisung zu einem bestimmten Gebäude und teilweise (im Falle der Fassade) auch die Angabe einer Himmelsrichtung (z. B. „Westen“) erforderlich.

Darüber hinaus kann der Klassentyp *loc:Space* mit dem Klassentyp *loc:Outside* kombiniert werden.

BEISPIEL 2 Ein Balkon ist sowohl vom Typ *loc:Outside* als auch vom Typ *loc:Space* (dies wird auch als „Class Assertion“ bezeichnet).

In Bild 5 werden diese Möglichkeiten in einem Beispiel erläutert.

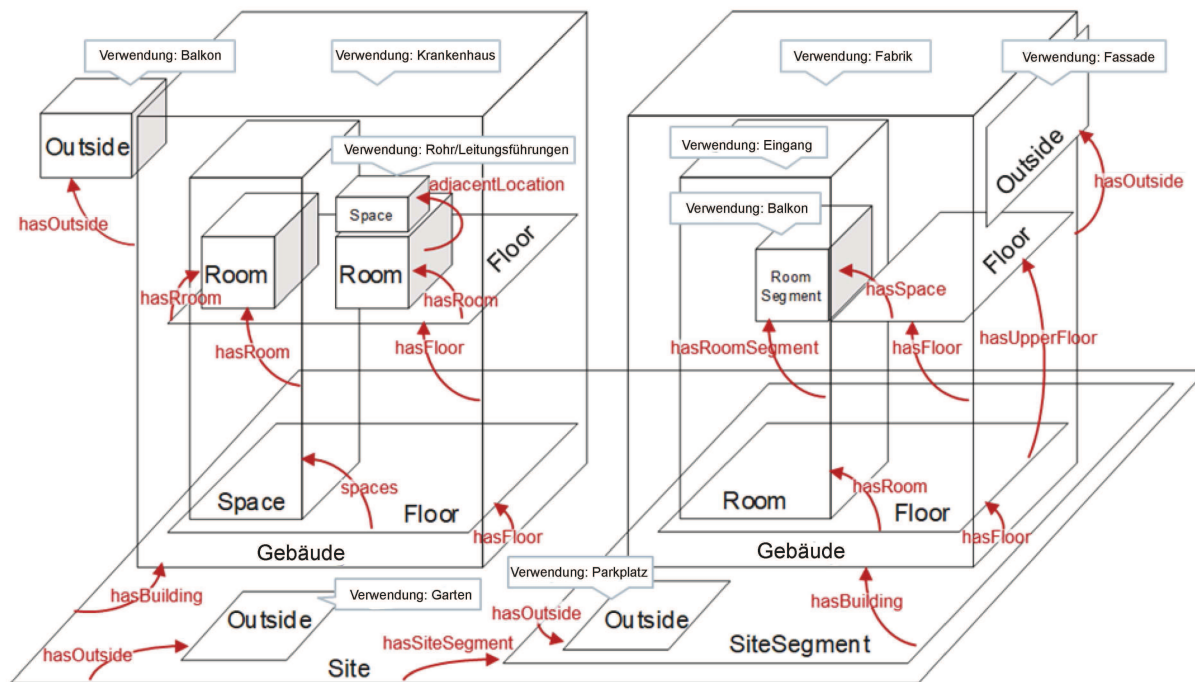


Bild 5 – Beispiel für Orte und Relationen

ANMERKUNG 2 Aus Gründen der Lesbarkeit wurde der Namensraumbezeichner „loc:“ in obigem Bild weggelassen.

4.3.2 Anforderungen

Das Ortsmodell erfüllt die folgenden Anforderungen.

Verbindung zum BIM

Viele Konzepte des Ortsmodells entsprechen Konzepten der IFC-Ontologie, die für BIM (Building Information Modeling)-Modelle entwickelt wurde. Insbesondere bringen *loc:Building*, *loc:Floor*, *loc:Site* und *loc:Space* der ESHG eine Referenz zu *:IfcBuilding*, *:IfcBuildingStorey*, *:IfcSite* und *:IfcSpace* aus IFC zum Ausdruck, so dass eine Verbindung zu einem BIM-Modell möglich wird.

Alternative Hierarchien

Die gemeinsame übergeordnete Klasse *loc:Location* des KIM ermöglicht die Festlegung einer alternativen Ortshierarchie bzw. ein Abweichen von der üblichen Hierarchie, sollte dies aus irgendeinem Grund gewünscht sein.

BEISPIEL 1 Ein KIM-Element *loc:Room* kann direkt einem *loc:Building* zugeordnet werden (und nicht, wie sonst üblich, einem *loc:Floor*). Dies erfolgt mittels eines starken semantischen Ausdrucks in Form der Objekteigenschaft „*loc:hasRoom*“ oder mittels eines schwachen semantischen Ausdrucks in Form der Objekteigenschaft „*loc:hasLocation*“, ohne dass dadurch Beschränkungen des Ortsmodells verletzt werden.

Erweiterbarkeit

Das Konzept *loc:Space* repräsentiert einen generischen Ort innerhalb eines Gebäudes. Deshalb kann es dazu verwendet werden, Typen von Orten zu definieren, die sich nicht ohne Weiteres in die Struktur der bestehenden Klassen des Ortsmodells einfügen lassen. Hierfür können entweder zusätzliche Eigenschaften definiert werden (in einer Erweiterung des bestehenden KIM) oder neue Unterklassen von *loc:Space* definiert werden, die besser für den jeweiligen Zweck geeignet sind.

BEISPIEL 2 Wenn ein Flügel als Teil eines Gebäudes modelliert werden soll, so kann eine Klasse *loc:Wing* mit speziellen Flügeigenschaften hinzugefügt werden.

4.3.3 Klassen und Unterklassen

ANMERKUNG Nachstehend werden nur die wichtigsten Ontologieklassen und deren wichtigste Eigenschaften beschrieben.

4.3.3.1 Location (Ort)

Es wird die folgende Struktur aus Klassen und Unterklassen definiert; vorhandene Über-/Unterordnungsbeziehungen zwischen Unterklassen werden durch Einrückung der Klassen dargestellt.

- *:Location* // generischer Platzhalter für ein (beliebiges) Ortskonzept
 - *loc:Location* // übergeordnete Klasse des ESHG-Ortskonzepts
 - *loc:Building*
 - *loc:Floor*
 - *loc:Outside*
 - *loc:Site*
 - *loc:SiteSegment*
 - *loc:Space*
 - *loc:Room*
 - *loc:RoomSegment*
- *vcard:Address*

Klasse	Definition
<i>loc:Location</i>	Beschreibung Ein <i>loc:Location</i> ist ein physischer, benannter geografischer Ort (wie z. B. „see-seitiger Besprechungsraum“, „Oskars Büro“, „der fünfte Stock“ oder „Wohnhaus mit Blick auf die Berge“ usw.), mit dem ein Punkt, ein Bereich oder ein Raum innerhalb oder außerhalb eines Gebäudes identifiziert wird. Wichtigste Objekt-/Dateneigenschaften / disjunkt zu — <i>loc:hasLocation</i>, <i>loc:hasAdjacentLocation</i>, <i>loc:hasLocationUsage</i>, <i>vcard:hasAddress</i>, <i>loc:hasOutside</i>, <i>:hasApplicationFunction</i>, <i>:hasFeatureOfInterest</i>, <i>loc:containsEquipment</i> — - — - Bemerkungen Die Klasse kann als Brückenklasse zu anderen Ontologien (z. B. dem Installationsmodell, siehe 4.4) verwendet werden. Das gesamte ESHG-Ortskonzept – für das der Namensraum <i>:loc</i> verwendet wird – ist als Unterklasse der generischen Platzhalterklasse <i>:Location</i> eingebettet. Dadurch kann das derzeitige ESHG-Ortskonzept problemlos durch eine alternative Ortsontologie ersetzt werden.

4.3.3.2 Building (Gebäude)

/loc:Building	Beschreibung Ein /loc:Building repräsentiert ein gesamtes Gebäude. Ein reales Gebäude beherbergt mehrere weitere reale Elemente wie z. B. Stockwerke, Flächen oder Räume. Wichtigste Objekt-/Dateneigenschaften / disjunkt zu — /loc:hasFloor, /loc:hasRoom, /loc:hasSpace — - — /loc:Site, /loc:Floor, /loc:Space, /loc:Outside Bemerkungen Es kann sinnvoll sein, eine Ortsverwendung und/oder eine Anschrift zu einem Gebäude anzugeben. Das Konzept kann auf das IFC-Konzept <i>IfcBuilding</i> abgebildet werden.
---------------	---

4.3.3.3 Floor (Stockwerk)

/loc:Floor	Beschreibung Ein /loc:Floor ist ein Ebenenkonzept in einem Gebäude. Ein Stockwerk unterteilt ein Gebäude in horizontale Flächen. Einem Stockwerk können Flächen zugewiesen sein, insbesondere ein Raum. Wichtigste Objekt-/Dateneigenschaften / disjunkt zu — /loc:hasLowerFloor, /loc:hasUpperFloor, /loc:hasRoom, /loc:hasSpace — - — /loc:Site, /loc:Building, /loc:Space, /loc:Outside Bemerkungen Es kann sinnvoll sein, eine Ortsverwendung für ein Stockwerk anzugeben; eine Anschrift ist üblicherweise bereits für das Gebäude angegeben, das das Stockwerk enthält. Das Konzept kann auf das IFC-Konzept <i>IfcBuildingStorey</i> abgebildet werden.
------------	--

4.3.3.4 Outside (Draußen)

/loc:Outside	Beschreibung <i>loc:Outside</i> ist eine Klassendefinition, die beschreibt, dass sich etwas von einem Gebäude aus betrachtet draußen befindet. <i>loc:Outside</i> ist eine Klassendefinition ohne Individuen. Ein außen liegender Ort kann mittels „Class Assertion“ festgelegt werden. — Ein Ortsindividuum wird per Assertion der Klasse <i>loc:Outside</i> zugewiesen. — Individuen, deren Typ disjunkt zu der Klasse <i>loc:Outside</i> ist, dürfen <i>loc:Outside</i> nicht per Assertion zugewiesen werden. — Sämtliche Individuen, die der Klasse per Assertion zugewiesen sind, befinden sich grundsätzlich außerhalb von Gebäuden, nicht nur außerhalb eines bestimmten Gebäudes. Die Relation, die angibt, außerhalb welchen konkreten Gebäudes sich das Individuum befindet, kann nicht allein per „Class Assertion“ ausgedrückt werden. Eine Relation zu einem außen liegenden Ort kann mittels „Property Assertion“ festgelegt werden. — Individuen, die per Assertion der OP-Relation <i>loc:hasOutside</i> zugewiesen sind, verfügen über Beziehungen von innerhalb eines bestimmten Gebäudes nach draußen. — Die Relation, die angibt, zu welchem Gebäude ein Individuum gehört, wird über die Abhängigkeit von seinem räumlichen Gebäudekontext geregelt, wie ein Raum, der als Teil eines Gebäudes ausgedrückt wird (mögliche Fehlanwendungen und Grenzfälle werden hier nicht betrachtet). — Für Einzelheiten zur Domäne / zum Bereich von OP-Relationen siehe 4.6.2. Wichtigste Objekt-/Dateneigenschaften / disjunkt zu — - — - — <i>loc:Room</i>, <i>loc:RoomSegment</i>, <i>loc:Building</i>, <i>loc:Floor</i> Bemerkungen Als außerhalb eines Gebäudes liegend gilt sowohl eine Fassade innerhalb des Umfangs der Gebäudehülle als auch eine Garage, die weit von dem Gebäude entfernt liegt. Beispiele für außen liegende Orte sind „Westfassade“ oder „Dach“ als Teile einer Installation (Instanzdaten); logischerweise ist hier die Zuweisung zu einem bestimmten Gebäude und teilweise (im Falle einer Fassade als <i>loc:Space</i>) auch die Angabe einer Himmelsrichtung (z. B. „Westen“) erforderlich.
--------------	---