OCF 2.0 – BRTG Zigbee Translation – Bridging Task Group CR 2476

Legal Disclaimer

THIS IS A DRAFT SPECIFICATION DOCUMENT ONLY AND HAS NOT BEEN ADOPTED BY THE OPEN CONNECTIVITY FOUNDATION. THIS DRAFT DOCUMENT MAY NOT BE RELIED UPON FOR ANY PURPOSE OTHER THAN REVIEW OF THE CURRENT STATE OF THE DEVELOPMENT OF THIS DRAFT DOCUMENT. THE OPEN CONNECTIVITY FOUNDATION AND ITS MEMBERS RESERVE THE RIGHT WITHOUT NOTICE TO YOU TO CHANGE ANY OR ALL PORTIONS HEREOF, DELETE PORTIONS HEREOF, MAKE ADDITIONS HERETO, DISCARD THIS DRAFT DOCUMENT IN ITS ENTIRETY OR OTHERWISE MODIFY THIS DRAFT DOCUMENT AT ANY TIME. YOU SHOULD NOT AND MAY NOT RELY UPON THIS DRAFT DOCUMENT IN ANY WAY, INCLUDING BUT NOT LIMITED TO THE DEVELOPMENT OF ANY PRODUCTS OR SERVICES. IMPLEMENTATION OF THIS DRAFT DOCUMENT IS DONE AT YOUR OWN RISK AMEND AND IT IS NOT SUBJECT TO ANY LICENSING GRANTS OR COMMITMENTS UNDER THE OPEN CONNECTIVITY FOUNDATION INTELLECTUAL PROPERTY RIGHTS POLICY OR OTHERWISE. IN CONSIDERATION OF THE OPEN CONNECTIVITY FOUNDATION GRANTING YOU ACCESS TO THIS DRAFT DOCUMENT, YOU DO HEREBY WAIVE ANY AND ALL CLAIMS ASSOCIATED HEREWITH INCLUDING BUT NOT LIMITED TO THOSE CLAIMS DISCUSSED BELOW, AS WELL AS CLAIMS OF DETRIMENTAL RELIANCE.

1 Scope

This document specifies a framework for translation between OCF devices and other ecosystems, and specifies the behaviour of a translator that exposes servers in non-OCF ecosystem to OCF clients and/or exposes OCF servers to clients in non-OCF ecosystem. Translation per specific device is left to other specification (deep translation). This document provides generic requirements that apply unless overridden by a more specific document.

2 Normative references

Zigbee 053474, *Zigbee Specification*, August 2015
http://www.zigbee.org/zigbee-for-developers/zigbee-3-0/

Zigbee 075123, *Zigbee Cluster Library Specification*, January 2016
http://www.zigbee.org/zigbee-for-developers/zigbee-3-0/

OCF Core Specification, Open Connectivity Foundation Core Specification, Version 1.3
https://openconnectivity.org/specs/OCF_Core_Specification_v1.3.0.pdf

OCF Resource Type Specification, *Open Connectivity Foundation Security Specification*, Version 1.3
https://openconnectivity.org/specs/OCF_Resource_Type_Specification_v1.3.0.pdf

OCF Bridging Specification, *Open Connectivity Foundation Bridging Specification*, Version 1.3
https://openconnectivity.org/specs/OCF_Bridging_Specification_v1.3.0.pdf

OCF Security Specification, *Open Connectivity Foundation Security Specification*, Version 1.3
https://openconnectivity.org/specs/OCF_Security_Specification_v1.3.0.pdf

IETF RFC 4122, *A Universally Unique IDentifier (UUID) URN Namespace*, July 2005
https://www.rfc-editor.org/info/rfc4122

3 Terms, definitions, symbols and abbreviations

3.1 Terms and definitions

3.1.1

Symmetric, Asymmetric Bridging

In symmetric bridging, a bridge device not only exposes OCF server to other ecosystem but also exposes other ecosystem's server to OCF, on the other hand, in asymmetric bridging a bridge device exposes OCF server to other ecosystem only or exposes other ecosystem's server to OCF only.

3.1.2

Bridged Protocol

Another protocol (e.g., AllJoyn, Zigbee) that is being translated to or from OCF protocols

3.1.3

Zigbee Attribute

A data entity which represents a physical quantity or state. This data is communicated to other devices using commands.

3.1.4

Zigbee Cluster

A cluster is a specification defining one or more attributes, commands, behaviors, and dependencies, which supports an independent utility or application function. The term may also be used for an implementation or instance of such a specification on an endpoint.

3.1.5

Zigbee Cluster Identifier

The cluster identifier is a 16-bit number that maps to (identifies) a single cluster specification. More than one cluster identifier may map to a cluster specification, each defining a different scope and purpose. Cluster identifiers are designated as inputs or outputs in the simple descriptor for use in creating a binding table. In this document, the Zigbee Cluster Identifier maps to one or more OCF Resource Types.

3.1.6

Zigbee Server

A cluster interface which is listed in the input cluster list of the simple descriptor on an endpoint. Typically this interface supports all or most of the attributes of the cluster. A server cluster communicates with a corresponding remote client cluster with the same Zigbee Cluster Identifier.

3.1.7

Zigbee 3.0 Server

Zigbee Server which is built on Zigbee 3.0 stack

3.1.8

Zigbee Client

A cluster interface which is listed in the output cluster list of the simple descriptor on an endpoint. Typically this interface sends commands that manipulate the attributes on the corresponding server cluster. A client cluster communicates with a corresponding remote server cluster with the same Zigbee Cluster Identifier.

3.1.9 Zigbee 3.0 Client

Zigbee Client which is built on Zigbee 3.0 stack

3.1.10 Corresponding cluster:

The opposite side of a cluster (client to a server, or server to a client).

3.1.11 Zigbee Device

A specification which defines a unique device identifier and a set of mandatory and optional clusters to be implemented on a single endpoint. The term may also be used for an implementation or instance of the device specification for on an endpoint. In this document, the unique identifier of Zigbee Device maps to an OCF Device Type.

3.1.12 Zigbee 3.0 Device

Zigbee Device which is built on Zigbee 3.0 stack

3.1.13 Zigbee Node

A ZigBee node (or node) is a single testable implementation of a ZigBee application on a single stack, with a single network address, on a single network.

5 OCF Bridge Device

5.2 Security Considerations

5.2.x Certificate Usage of Virtual Server in Asymmetric Bridging

Since a Virtual OCF Server is treated as a physical OCF Server, the certificates and key pairs for the Virtual OCF server are required to support the certificate based operations defined for a general OCF Server. i.e., Certificate based OTM and Device Authentication with Certificate (Please see OCF Security Specification and OCF Core Specification Extension – CoAPNativeCloud)

A Virtual OCF Server uses the certificates and the key pairs of Bridge to support the certificate based operations. The certificates of Bridge are manufacturer certificate and trust anchor certificate(s) which are installed by Bridge manufacturer. Manufacturer certificate is used for Manufacturer Certificate based OTM (See OCF Security Specification). The both the manufacturer certificate and the trust anchor certificate(s) are used for Device Authentication operation. i.e., Device Connection with the Cloud and Device Registration with the Cloud (See OCF Security Specification and OCF Core Specification Extension – CoAPNativeCloud). The private keys of Bridge's certificates are used to sign handshake messages in both operations.

A Virtual Server should use unique credentials for local connection after the onboarding is complete

10 Zigbee Translation

10.1 Operational Scenarios

The overall goal is to make Bridged Zigbee 3.0 Servers appear to OCF Clients as if they were native OCF Servers in the local network or cloud environment

"Deep translation" is specified in a separate document. "on-the-fly" translation is out of scope. (Please see section **X.X** for the definition of "Deep translation and "on-the-fly")

Figure 1 shows an overview of a Zigbee 3.0 Bridge device and its general topology. It exposes Zigbee 3.0 Servers to OCF Clients and OCF Clouds. Each Bridged Zigbee 3.0 Server is represented as a Virtual OCF Server. The Zigbee 3.0 Translator supports Asymmetric bridging. The scope of this document is the asymmetric bridging to expose the Zigbee Server to OCF. The asymmetric bridging to expose an OCF Server to a Zigbee Client is out of scope.

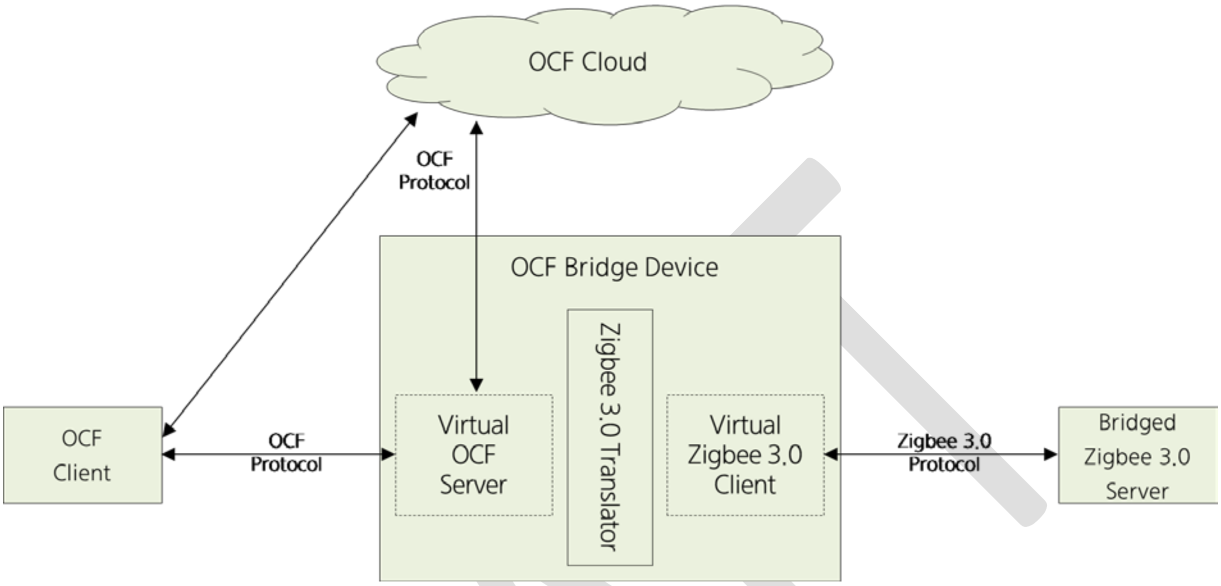

Figure 1 OCF Zigbee Bridge Device Components & Topology

10.1.1 Use case for OCF client and Zigbee 3.0 server

Figure 2 shows a use case for an OCF Client and Zigbee 3.0 Server. An OCF client on a smartphone controls a Zigbee light device through an OCF Zigbee Bridge. Any connectivity that OCF supports is used for communications between the OCF Client and the OCF Zigbee Bridge. The OCF Client can communicate with OCF Zigbee Bridge through OCF Cloud. A Zigbee 3.0 Stack and above is used for communications between the OCF Zig bee Bridge and the Zigbee server.

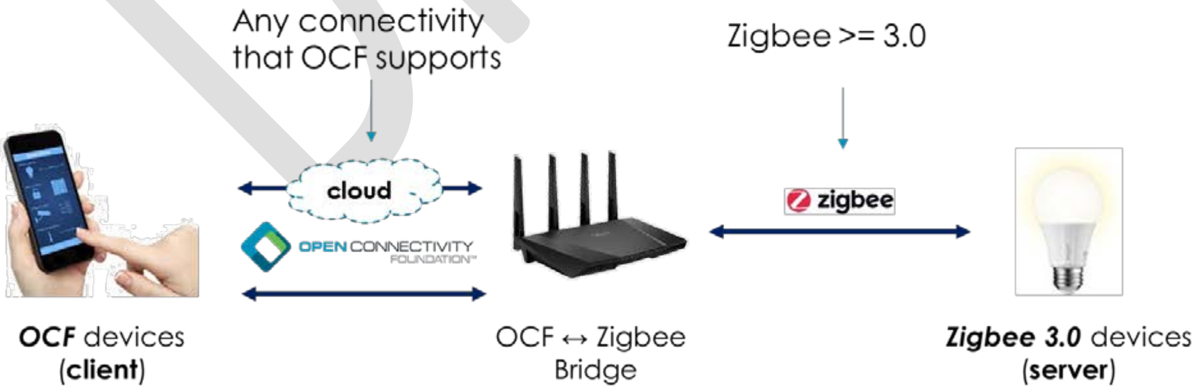
Any connectivity that OCF supports
Zigbee >= 3.0
cloud
OPEN CONNECTIVITY FOUNDATION
zigbee
OCF devices (client)
OCF ↔ Zigbee Bridge
Zigbee 3.0 devices (server)

Figure 2 OCF Client and Zigbee Server

10.2 Requirements specific to Zigbee Translator

10.2.1 Requirements specific to Zigbee

This document refers to Zigbee 3.0 or higher. Zigbee 3.0 is built on Zigbee Pro 2015 or newer, which enhances the IEEE 802.15.4 standard by adding a mesh network and security layers along with an application framework. Low power support is not the scope of this document.

An OCF Zigbee Bridge Device shall act as a Zigbee Coordinator in network layer. A Zigbee Coordinator is responsible for initiating and maintaining the devices on the network. An OCF Zigbee Bridge Device will act as Zigbee Client towards the Zigbee 3.0 Devices in the application layer. Users can expect that a certified OCF Bridge Device will be able to talk to Zigbee 3.0 Devices, without the user having to buy some other device.

Please see section **5.1** for General Requirements.

10.2.2 Exposing Zigbee 3.0 servers to OCF clients

The nature of how Zigbee Devices are structured may be different than how an OCF Device is structured. The mapping of the structure of a Zigbee device on an OCF Device is given by Table 1.

A Zigbee Server cluster maps to one or more OCF Resources. If a specific Zigbee Server cluster has specific commands, one or more OCF Resources corresponding to the specific command attributes are additionally required.

A Zigbee Attribute of a Zigbee Server cluster typically maps to an OCF Resource Property. However, in some special cases, multiple attributes are mapped to a single OCF Resource Property e.g., ‘CurrentX’ and ‘CurrentY’ of the Zigbee color control cluster map to the ‘csc’ Property in the ‘oic.r.colour.csc’ (Colour Space Coordinates) Resource because of the difference in the data types, i.e., ‘csc’ is an array, but CurrentX and CurrentY map to a number.

Table 2 is a mapping example of this rule

Table 1 Translation Rule between Zigbee and OCF Data Models

From Zigbee	mapping count	To OCF	mapping count
Zigbee Device	**1**	OCF Device	**1**
Zigbee Cluster	**1**	OCF Resource	**n**
Zigbee Attribute	**1**	OCF Resource Property	**1**

Table 2 Zigbee → OCF Mapping Example (Color Temperature Light)

From Zigbee		To OCF	
Zigbee 3.0 Device	0x010c (Color Temperature Light)	**OCF Device**	oic.d.light (Light)
Zigbee Server Cluster	0x0006 (On/Off)	**OCF Resource(s)**	oic.r.switch.binary (Binary Switch)
	0x0300 (Color Control Cluster)		oic.r.colour.hs (Colour Hue and Saturation)
			oic.r.colour.csc

			(Colour Space Coordinates)
			oic.r.colour.colourtemperature (Colour Temperature)
Zigbee Attribute	0x0000 (OnOff of On/Off Cluster)	**OCF Resource Property**	value (of Binary Switch Resource)
	0x0003 (CurrentX of Color Control Cluster)		csc (of Colour Space Coordinates)
	0x0004 (CurrentY of Color Control Cluster)		

If a Zigbee 3.0 Device, Zigbee Server Cluster, Zigbee Attribute are enlisted in the well-defined set (Please see OCF to Zigbee Data Model Mapping Specification), the translator follows the specification for translating it to an OCF Device, OCF Resource, or OCF Resource Property (i.e., ”deep translation”).

A Zigbee 3.0 Server Device maps to a single OCF Device Type. The OCF Device Type is provided by using the Device ID of the Zigbee 3.0 Server Device (The Device ID is allocated by the Zigbee Alliance and has the same meaning of the OCF Device Type). The Zigbee 3.0 Translator has a table which includes the mapping information between the Zigbee Device ID and the OCF Device Type. Based on the table, the Zigbee 3.0 Translator finds the OCF Device Type according to the Zigbee Device ID.

A Zigbee Device includes one or more Zigbee Server Clusters. If a Zigbee Cluster maps to multiple OCF Resources, the Zigbee Cluster is translated as a Resource with a Collection Resource Type. The resource mapping between Zigbee Server Cluster and OCF Resources is defined in the OCF to Zigbee Data Model Mapping specification for deep translation.The Zigbee 3.0 Translator has a table which includes the mapping information between the identifier of Zigbee Cluster and OCF Resource Type(s). The Zigbee 3.0 Translator obtains the list of cluster identifiers after the Virtual Zigbee 3.0 Client and Zigbee 3.0 Server Device are bound. Based on the table, the Zigbee 3.0 Translator finds the OCF Resource Type(s) according to the identifier of Zigbee Cluster.

Since a Bridge Device knows all relationships between OCF Resources and Zigbee Server Clusters, the path component of URI can be free to choose. Maintaining relationship information and URI definition is implementation specific.

If a Zigbee operation fails, the translator send an appropriate OCF error response to the OCF Client. it construct an appropriate OCF error message (e.g., diagnostic payload if using CoAP) from the Zigbee enumerated status value and Zigbee enumerated status (if any), using the form "<error name>: <error message>", with the <error name> taken from the Zigbee Status Code field and the <error message> taken from the Zigbee enumerated status, and the error code for the OCF network set to an appropriate value.

10.2.2.1 Translation for well-defined set

If a Zigbee 3.0 Device, Zigbee Server Cluster, Zigbee Attribute are enlisted in the well-defined set (Please see OCF to Zigbee Data Model Mapping Specification), the translator follows the specification for translating it to an OCF Device, OCF Resource, or OCF Resource Property (i.e., ”deep translation”). Table 3 is the list of Zigbee 3.0 devices and mandatory Zigbee Server Clusters with corresponding OCF devices and mandatory OCF Resources. Optional OCF Resources mapped with the specific Zigbee Server Clusters are enlisted in the well-defined set (Please see OCF to Zigbee Data Model Mapping Specification).

Table 3 Zigbee 3.0 Device & Cluster – OCF Device & Resource mapping

Zigbee 3.0 Device Name (Device ID)	Zigbee 3.0 Mandatory Cluster	OCF Mandatory Resource Type	OCF Device Type (“rt”)	OCF Device Name
On/off light (0x0100)	On/off	oic.r.switch.binary,	oic.d.light	Light
Color Temperature Light (0x010c)	On/off, Level Control, Color Control	oic.r.switch.binary,	oic.d.light	Light
Extended Color Light (0x010d)	On/off, Level Control, Color Control	oic.r.switch.binary,	oic.d.light	Light
Dimmable Light (0x0101)	On/off, Level Control	oic.r.switch.binary,	oic.d.light	Light
Color Dimmable Light (0x0102)	On/off Level Control, Color Control	oic.r.switch.binary,	oic.d.light	Light
Temperature Sensor (0x0302)	Temperature Measurement	oic.r.temperature	oic.d.sensor	Generic Sensor
Thermostat (0x0301)	Thermostat	oic.r.temperature(2)	oic.d.thermostat	Thermostat
Window Covering Device (0x0202)	Window Covering	oic.r.openlevel	oic.d.blind	Blind
Smart Plug (0x0051)	On/off, Metering	oic.r.switch.binary,	oic.d.smartplug	Smart Plug
Mains Power Outlet (0x0009)	On/off	oic.r.switch.binary,	oic.d.smartplug	Smart Plug
On/off output (0x0002	On/off	oic.r.switch.binary,	oic.d.smartplug	Smart Plug
IAS Zone (0x0402)	IAS Zone	oic.r.ias.zone	oic.d.sensor	Generic Sensor

Exposing a Zigbee 3.0 Server as a Virtual OCF Server

Table 4 shows how OCF Platform properties, as specified in Table 21 in the OCF Core Specification, shall be derived, typically from fields of Descriptor specified in the Zigbee Specification.

Table 4: oic.wk.p Resource Type definition

To OCF Property title	OCF Property name	OCF Description	OCF Mandatory	From Zigbee 3.0 Field name	Zigbee 3.0 Description	Zigbee 3.0 Mandatory
Platform ID	pi	Unique identifier for the physical platform (UIUID); this shall be a UUID in accordance with IETF RFC 4122. It is recommended that the UUID be created using the random generation scheme (version 4 UUID) specific in the RFC.	Y	(none)	Translator should return a randomly-generated UUID (Please see section 4.4 of IETF RFC 4122 for randomly-generated UUID)	
Manufacturer Name	mnmn	Name of manufacturer (not to exceed 16 characters)	Y	Manufacturer name (in DefaultLanguage, truncated to 16 characters)	Name of the manufacturer as a ZigBee character string Defined in Basic Cluster	Y
Manufacturer Details Link (URL)	mnml	URL to manufacturer (not to exceed 32 characters)	N	(none)	(none)	N
Model Number	mnmo	Model number as designated by manufacturer	N	Model Identifier	Model number (or other identifier) assigned by the manufacturer as a ZigBee character string Defined in Basic Cluster	Y
Date of Manufacture	mndt	Manufacturing date of device	N	DateCode	Date of manufacturer of the device in international date notation according to ISO 8601, i.e., YYYYMMDD, Defined in Basic Cluster	N
Platform Version	mnpv	Version of platform – string (defined by manufacturer)	N	(none)	(none)	N
OS Version	mnos	Version of platform resident OS – string (defined by manufacturer)	N	(none)	(none)	N

To OCF Property title	OCF Property name	OCF Description	OCF Mandatory	From Zigbee 3.0 Field name	Zigbee 3.0 Description	Zigbee 3.0 Mandatory
Hardware Version	mnhw	Version of platform hardware	N	HWVersion	Version number of the hardware of the device. Defined in Basic Cluster	N
Firmware version	mnfv	Version of device firmware	N	(none)	(none)	N
Support link	mnsl	URI that points to support information from manufacturer	N	ProductURL	Link to a web page containing specific product information Defined in Basic Cluster	N
SystemTime	st	Reference time for the device	N	(none)	(none)	N
Vendor ID	vid	Vendor defined string for the platform. The string is freeform and up to the vendor on what text to populate it.	N	(none)	(none)	N

Table 5 shows how OCF Device Properties, as specified in Table 20 in the OCF Core Specification, shall be derived, typically from fields of Descriptor or Attributes of Basic cluster specified in the Zigbee Specification and the Zigbee Cluster Library, respectively.

As specified in the OCF Security Specification, the value of the "di" Property of OCF Devices (including Virtual OCF Devices) shall be established as part of Onboarding of that Virtual OCF Device.

Table 5: oic.wk.d resource type definition

To OCF Property title	OCF Property name	OCF Description	OCF Mandatory	From Zigbee 3.0 Field name	Zigbee 3.0 Description	Zigbee 3.0 Mandatory
(Device) Name	n	Human friendly name For example, "Bob's Thermostat"	Y	User description if it exists, else Model Name if it exists, else translate Application Device Identifier (=Device ID)to Human friendly name by using Application Device Identifier value/description table	User description : Information that allows the user to identify the device using a user-friendly character string, such as "Bedroom TV" Defined in User Descriptor Model Name : character string representing the name of the manufacturer's model of the device Defined in Complex Descriptor Application Device Identifier: device description supported on this endpoint Cluster Defined in Simple Descriptor	User description: N Model Name: N Application Device Identifier: Y

To OCF Property title	OCF Property name	OCF Description	OCF Mandatory	From Zigbee 3.0 Field name	Zigbee 3.0 Description	Zigbee 3.0 Mandatory
Spec Version	icv	Spec version of the core specification this device is implemented to, The syntax is "core.major.minor”]	Y	(none)	Spec version of the core specification that the translator implements should return its own value	
Device ID	di	Unique identifier for Device. This value shall be as defined in OCF Security Specification for DeviceID.	Y	(none)	Use as defined in the the OCF Security Specification	
Protocol-Independent ID	piid	Unique identifier for OCF Device (UUID)	Y	(none)	Translator should return a randomly-generated UUID (Please see section 4.4 of IETF RFC 4122 for randomly-generated UUID)	
Data Model Version	dmv	Spec version(s) of the vertical specifications this device data model is implemented to. The syntax is a comma separated list of "<vertical>.major.minor”]. <vertical> is the name of the vertical (i.e. sh for Smart Home)	Y	(none)	Translator should return its own value.	
Localized Descriptions	ld	Detailed description of the Device, in one or more languages. This property is an array of objects where each object has a ‘language’ field (containing an RFC 5646 language tag) and a ‘value’ field containing the device description in the indicated language.	N	(none)	Zigbee provides Language and Character Set field only which specifies the language and character set used by the character strings by using ISO 639-1 language code	

To OCF Property title	OCF Property name	OCF Description	OCF Mandatory	From Zigbee 3.0 Field name	Zigbee 3.0 Description	Zigbee 3.0 Mandatory
Software Version	sv	Version of the device software.	N	ApplicationVersion	Version number of the application software contained in the device. Defined in Basic Cluster	Y
Manufacturer Name	dmn	Name of manufacturer of the Device, in one or more languages. This property is an array of objects where each object has a 'language' field (containing an RFC 5646 language tag) and a 'value' field containing the manufacturer name in the indicated language.	N	Manufacturer name	Name of the manufacturer as a ZigBee character string Defined in Basic Cluster	Y
Model Number	dmno	Model number as designated by manufacturer.	N	Model Identifier	Model number (or otheridentifier) assigned by the manufacturer as a ZigBee character string Defined in Basic Cluster	Y

Table 6 shows how OCF Device Configuration properties, as specified in Table 15 in OCF Core Specification, shall be derived:

Table 6: oic.wk.con resource type definition

To OCF Property title	OCF Property name	OCF Description	OCF Mandatory	From Zigbee 3.0 Field name	Zigbee 3.0 Description	Zigbee 3.0 Mandatory

To OCF Property title	OCF Property name	OCF Description	OCF Mandatory	From Zigbee 3.0 Field name	Zigbee 3.0 Description	Zigbee 3.0 Mandatory
(Device) Name	n	Human friendly name For example, “Bob’s Thermostat”	Y	User description if it exists, else Model Name if it exists, else translate Application Device Identifier (=Device ID)to Human friendly name by using Application Device Identifier value/description table	User description : Information that allows the user to identify the device using a user-friendly character string, such as “Bedroom TV” Defined in User Descriptor Model Name : character string representing the name of the manufacturer’s model of the device Defined in Complex Descriptor Application Device Identifier: device de-scription supported on this endpoint Cluster Defined in Simple Descriptor	User description: N Model Name: N Application Device Identifier: Y
Location	loc	Provides location information where available.	N	(none)	(none)	
Location Name	locn	Human friendly name for location For example, “Living Room”.	N	(none)	(none)	
Currency	c	Indicates the currency that is used for any monetary transactions	N	(none)	(none)	
Region	r	Free form text Indicating the current region in which the device is located geographically. The free form text shall not start with a quote (").	N	(none)	(none)	
Localized Names	ln	Human-friendly name of the Device, in one or more languages. This property is an array of objects where each object has a ‘language’ field (containing an RFC 5646 language tag) and a ‘value’ field containing the device name in the indicated language. If this property and the Device Name (n) property are both supported, the Device Name (n) value shall be included in this array.	N	User description if it exists, else Model Name if it exists, else translate Application Device Identifier (=Device ID)to Human friendly name by using Application Device Identifier value/description table	User description : Information that allows the user to identify the device using a user-friendly character string, such as “Bedroom TV” Defined in User Descriptor Model Name : character string representing the name of the manufacturer’s model of the device Defined in Complex Descriptor Application Device Identifier: device de-scription supported on this endpoint Cluster Defined in Simple Descriptor	User description: N Model Name: N Application Device Identifier: Y
Default	dl	The default	N	ISO 639-1	Language used for	N

To OCF Property title	OCF Property name	OCF Description	OCF Mandatory	From Zigbee 3.0 Field name	Zigbee 3.0 Description	Zigbee 3.0 Mandatory
Language		language supported by the Device, specified as an RFC 5646 language tag. By default, clients can treat any string property as being in this language unless the property specifies otherwise.		language code (if it exists, else property shall be absent)	character strings.	

10.2.2.2 On-the-fly Translation

If a Zigbee 3.0 Device, Zigbee Server Cluster, Zigbee Attribute are not enlisted in the well-defined set (Please see OCF to Zigbee Data Model Mapping Specification), the translator may not translate it.

10.2.3 Security

Please see section **5.2** for general security requirements.

A Bridge shall be able to block the communication of all OCF Devices with all Bridged Zigbee 3.0 Devices that don’t communicate securely with the Bridge.

- The Zigbee 3.0 stack supports multiple security levels (Please see Zigbee Specification). A security level is supported by both the network (NWK) layer and application support (APS) layer. A security attribute in the Zigbee 3.0 stack, nwkSecurityLevel, represents the security level of a device.

- All Zigbee Servers which want to communicate with an OCF Client through a Zigbee Bridge shall satisfy: nwkSecurityLevel > 0x04. nwkSecurityLevel > 0x04 provides message integrity code (MIC) and/or AES128-CCM encryption (ENC).

Table 7 Zigbee 3.0 Security Levels to the Network, and Application Support layers

Security Level Identifier	Security Level Sub-Field	Security Attributes	Data Encryption	Frame Integrity (Length of M of MIC, in Number of Octets)
0x00	‘000’	None	OFF	NO (M=0)
0x01	‘001’	MIC-32	OFF	YES(M=4)
0x02	‘010’	MIC-64	OFF	YES(M=8)
0x03	‘011’	MIC-128	OFF	YES(M=16)
0x04	‘100’	ENC	ON	NO(M=0)
0x05	‘101’	ENC-MIC-32	ON	YES(M=4)
0x06	‘110’	ENC-MIC-64	ON	YES(M=8)
0x07	‘111’	ENC-MIC-128	ON	YES(M=16)

Figure 3 shows how communications in both ecosystems of OCF Bridge device are secured by their own security.

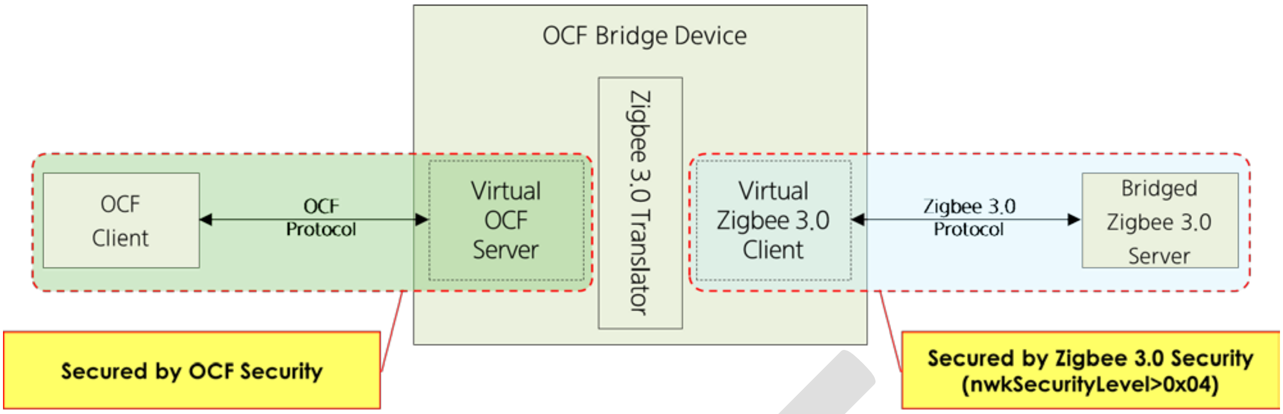

Figure 3 Security for OCF and Zigbee 3.0

12 Device Type definitions

No additional definitions are required.

13 Resource Type definitions

Since on-the-fly translation is out of scope, general Resource Type of Collection Resource for Zigbee Cluster, e.g., oic.r.zigbeecluster is not required.