OCF Matter Bridging Derived Model Specification

VERSION 2.2.7 | August 24, 2023

OPEN CONNECTIVITY FOUNDATION®

CONTACT admin@openconnectivity.org

Legal Disclaimer

Introduction

This document, and all the other parts associated with this document, were developed in response to worldwide demand for smart home focused Internet of Things (IoT) devices, such as appliances, door locks, security cameras, sensors, and actuators; these to be modelled and securely controlled, locally and remotely, over an IP network.

While some inter-device communication existed, no universal language had been developed for the IoT. Device makers instead had to choose between disparate frameworks, limiting their market share, or developing across multiple ecosystems, increasing their costs. The burden then falls on end users to determine whether the products they want are compatible with the ecosystem they bought into, or find ways to integrate their devices into their network, and try to solve interoperability issues on their own.

In addition to the smart home, IoT deployments in commercial environments are hampered by a lack of security. This issue can be avoided by having a secure IoT communication framework, which this standard solves.

The goal of these documents is then to connect the next 25 billion devices for the IoT, providing secure and reliable device discovery and connectivity across multiple OSs and platforms. There are multiple proposals and forums driving different approaches, but no single solution addresses the majority of key requirements. This document and the associated parts enable industry consolidation around a common, secure, interoperable approach.

The OCF specification suite is made up of nineteen discrete documents, the documents fall into logical groupings as described herein:

- Core framework
  - Core Specification
  - Security Specification
  - Onboarding Tool Specification
- Bridging framework and bridges
  - Bridging Specification
  - OCF Resource to Alljoyn Interface Mapping Specification
  - OCF Resource to oneM2M Resource Mapping Specification
  - OCF Resource to BLE Mapping Specification
  - OCF Resource to EnOcean Mapping Specification
  - OCF Resource to LWM2M Mapping Specification
  - OCF Resource to UPlus Mapping Specification
  - OCF Resource to Zigbee Cluster Mapping Specification
  - OCF Resource to Z-Wave Mapping Specification
- Resource and Device models
  - Resource Type Specification
  - Device Specification
- Core framework extensions
  - Easy Setup Specification

  - Core Optional Specification
- OCF Cloud
  - Cloud API for Cloud Services Specification
  - Device to Cloud Services Specification
  - Cloud Security Specification

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

Matter Core Specification 1.0
https://csa-iot.org/wp-content/uploads/2022/11/22-27349-001_Matter-1.0-Core-Specification.pdf

Matter Device Library Specification 1.0
https://csa-iot.org/wp-content/uploads/2022/11/22-27351-001_Matter-1.0-Device-Library-Specification.pdf

Matter Application Cluster Specification 1.0
https://csa-iot.org/wp-content/uploads/2022/11/22-27350-001_Matter-1.0-Application-Cluster-Specification.pdf

ISO/IEC 30118-1 Information technology -- Open Connectivity Foundation (OCF) Specification -- Part 1: Core specification
https://www.iso.org/standard/82127.html
Latest version available at: https://openconnectivity.org/specs/OCF_Core_Specification.pdf

ISO/IEC 30118-2 Information technology – Open Connectivity Foundation (OCF) Specification – Part 2: Security specification
https://www.iso.org/standard/82128.html
Latest version available at: https://openconnectivity.org/specs/OCF_Security_Specification.pdf

ISO/IEC 30118-3 Information technology – Open Connectivity Foundation (OCF) Specification – Part 3: Bridging specification
https://www.iso.org/standard/82129.html
Latest version available at: https://openconnectivity.org/specs/OCF_Bridging_Specification.pdf

ISO/IEC 30118-4 Information technology – Open Connectivity Foundation (OCF) Specification – Part 4: Resource Type specification
https://www.iso.org/standard/82130.html
Latest version available at:
https://openconnectivity.org/specs/OCF_Resource_Type_Specification.pdf

ISO/IEC 30118-5 Information technology – Open Connectivity Foundation (OCF) Specification – Part 5: Device specification
https://www.iso.org/standard/82131.html
Latest version available at: https://openconnectivity.org/specs/OCF_Device_Specification.pdf

Derived Models for Interoperability between IoT Ecosystems, Stevens & Merriam, March 2016
https://www.iab.org/wp-content/IAB-uploads/2016/03/OCF-Derived-Models-for-Interoperability-Between-IoT-Ecosystems_v2-examples.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

For the purposes of this document, the terms and definitions given in Error: Reference source not found, ISO/IEC 30118-2 Information technology – Open Connectivity Foundation (OCF) Specification – Part 2: Security specification , and ISO/IEC 30118-3 and the following apply.

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

– ISO Online browsing platform: available at https://www.iso.org/obp
– IEC Electropedia: available at http://www.electropedia.org/

3.1.1 Attribute

A data entity which represents a physical quantity or state. This data is communicated to other Nodes using commands. (a term of Matter)

3.1.2 Command

Requests for action on a value with an expected response which may have parameters and a response with a status and parameters. (a term of Matter)

3.1.3 Bridged Protocol

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

3.1.4 Client

A Cluster interface that typically sends commands that manipulate the attributes on the corresponding server cluster. A client cluster communicates with a corresponding remote server cluster with the same cluster identifier. (a term of Matter)

(OCF and Matter define "Client" but its meaning is different. If "Client" is used in the context of Matter, it follows this definition)

3.1.5 Cluster

A specification defining one or more attributes, commands, behaviors and dependencies, that 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. (a term of Matter)

3.1.6 Endpoint

A particular component within a Node that is individually addressable. (a term of Matter)

(OCF and Matter define "Endpoint" but its meaning is different. If "Endpoint" is used in the context of Matter, it follows this definition)

3.1.7 Node

An addressable entity which supports the Matter protocol stack and (once Commissioned) has its own Operational Node ID and Node Operational credentials. A Device MAY host multiple Nodes. (a term of Matter)

3.1.8 Server

A Cluster interface that typically supports all or most of the attributes of the Cluster. A Server Cluster communicates with a corresponding remote Client Cluster with the same Cluster identifier. (a term of Matter)

(OCF and Matter define "Server" but its meaning is different. If "Server" is used in the context of Matter, it follows this definition)

3.1.9 Symmetric, Asymmetric Bridging

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

3.2 Symbols and abbreviations

CBOR	Concise Binary Object Representation
CoAP	Constrained Application Protocol
CoAPs	Secure Constrained Application Protocol
DTLS	Datagram Transport Layer Security
IP	Internet Protocol
TLV	Tag Length Value
VOD	Virtual OCF Device

7 Theory of Operation

7.1 Interworking Approach

The mapping between Matter-defined Cluster data model and OCF-defined Resource data model is described using the derived model syntax outlined in "Derived Models for Interoperability between IoT Ecosystems".

7.2 Mapping Syntax

In the syntax defined for derived modelling in this document, there are two blocks that define the actual Property-Property equivalence or mapping. These blocks are identified by the keywords "x-to-ocf" and "x-from-ocf". Derived Models for Interoperability between IoT Ecosystems do not define a rigid syntax for these blocks; they are free-form string arrays that contain pseudo-coded mapping logic.

In this document, Python (version >= 3.0) syntax is used to describe translation rules.

The JSON skeleton shows typical translation block used in the derived models.

```
"<Matter Cluster Name>" : {
   "type": "object",
   "properties": {
      "<an attribute name of the Matter Cluster>" : {
         "x-ocf-conversion" : {
            "x-ocf-alias": "<corresponding Property of the OCF Resource>",
```

```
            "x-to-ocf": [
               ...
            ],
            "x-from-ocf": [
               ...
            ]
         }
      }
   }
}
```

– <Matter Cluster Name>: this is the name of a Matter Cluster. If a name hasa space, it shall be removed (e.g. "Fan Control" → "FanControl"). It follows the names defined in "Matter Application Cluster Specification 1.0".

– <an attribute name of the Matter Cluster>: “an attribute name of the Matter Cluster” is a name of an attribute which belongs to the <Matter Cluster Name>. It follows the names defined in "Matter Application Cluster Specification 1.0".

– <corresponding Property of the OCF Resource>: a Property name of the OCF Resource type which is corresponding to the <an attribute name of the Matter Cluster>. Notation rule for this follows clause 4.4 of "ISO/IEC 30118-1".

8 Translation rules per device type

8.1 Introduction

This clause provides mapping descriptions per each Device Type. Mapping descriptions are based on the derived modelling syntax outlined in "Derived Models for Interoperability between IoT Ecosystems".

8.2 On/Off Light

8.2.1 Derived model

The derived model: "OnOffLight.json"

8.2.2 Property definition

Table 1 shows mapping between Matter On/Off Cluster and OCF binary switch Resource. The "OnOff" attribute of On/Off Cluster is read access only, so writing to the "OnOff" attribute can only be done through "On" or "Off" command provided by On/Off Cluster.

Table 1 Mapping for Matter On/Off Cluster

Matter Cluster Attribute name	OCF Resource Property name	To OCF	From OCF

OnOff	oic.r.switch.binary:value	oic.r.switch.binary:value = OnOff	oic.r.switch.binary:value = new_value
			if oic.r.switch.binary:value == True:
			  invoke(On)
			else:
			  invoke(Off)

Table 2 Corresponding attributes of On/Off Cluster

Matter Cluster Attribute name	Type	Required	Description
OnOff	bool	Yes	The OnOff attribute indicates whether the device type implemented on the endpoint is turned off or turned on, in these cases the value of the OnOff attribute equals FALSE, or TRUE respectively.

8.2.3 Derived model definition

```
{
  "id": "https://github.com/openconnectivityfoundation/OCF-Matter/OnOffLight.json#",
  "$schema": "http://json-schema.org/draft-04/schema#",
  "description" : "Copyright (c) 2023 Open Connectivity Foundation, Inc. All rights reserved.",
  "title": "On/Off Light Device Type",
  "definitions": {
    "OnOff" : {
      "description": "On/Off Cluster includes Attributes and commands for turning devices on and off",
      "type" : "object",
      "properties": {
        "OnOff": {
          "description": "OnOff attribute indicates whether the device type implemented on the
endpoint is turned off or turned on",
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.switch.binary:value",
            "x-to-ocf": [
              "oic.r.switch.binary:value = OnOff"
            ],
            "x-from-ocf": [
              "oic.r.switch.binary:value = new_value",
              "if oic.r.switch.binary:value == True:",
              "  invoke(On)",
              "else:",
              "  invoke(Off)"
            ]
          }
        }
      }
    }
  },

  "type": "object",

  "required": [
    "OnOff"
  ]
}
```

8.3 On/Off Plug-in Unit

8.3.1 Derived model

The derived model: "OnOffPluginUnit.json"

8.3.2 Property definition

Table 3 shows mapping between Matter On/Off Cluster and OCF binary switch Resource. The "OnOff" attribute of On/Off Cluster is read access only, so writing to the "OnOff" attribute can only be done through "On" or "Off" command provided by On/Off Cluster.

Table 3 Mapping for Matter On/Off Cluster

Matter Cluster Attribute name	OCF Resource Property name	To OCF	From OCF

OnOff	oic.r.switch.binary:value	oic.r.switch.binary:value = OnOff	oic.r.switch.binary:value = new_value
			if oic.r.switch.binary:value == True:
			  invoke(On)
			else:
			  invoke(Off)

Table 4 Corresponding attributes of On/Off Cluster

Matter Cluster Attribute name	Type	Required	Description
OnOff	bool	Yes	The OnOff attribute indicates whether the device type implemented on the endpoint is turned off or turned on, in these cases the value of the OnOff attribute equals FALSE, or TRUE respectively.

8.3.3 Derived model definition

```
{
  "id": "https://github.com/openconnectivityfoundation/OCF-Matter/OnOffPluginUnit.json#",
  "$schema": "http://json-schema.org/draft-04/schema#",
  "description" : "Copyright (c) 2023 Open Connectivity Foundation, Inc. All rights reserved.",
  "title": "On/Off Plug-in Unit Device Type",
  "definitions": {
    "OnOff" : {
      "description": "On/Off Cluster includes Attributes and commands for turning devices on and off",
      "type" : "object",
      "properties": {
        "OnOff": {
          "description": "OnOff attribute indicates whether the device type implemented on the
endpoint is turned off or turned on",
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.switch.binary:value",
            "x-to-ocf": [
              "oic.r.switch.binary:value = OnOff"
            ],
            "x-from-ocf": [
              "oic.r.switch.binary:value = new_value",
              "if oic.r.switch.binary:value == True:",
              "  invoke(On)",
              "else:",
              "  invoke(Off)"
            ]
          }
        }
```

```
      }
    }
  },

  "type": "object",

  "required": [
    "OnOff"
  ]
}
```

8.4 Dimmable Light

8.4.1 Derived model

The derived model: "DimmableLight.json"

8.4.2 Property definition

Table 5 shows mapping between Matter On/Off Cluster and OCF binary switch Resource. The "OnOff" attribute of On/Off Cluster is read access only, so writing to the "OnOff" attribute can only be done through "On" or "Off" command provided by On/Off Cluster.

Table 5 Mapping for Matter On/Off Cluster

Matter Cluster Attribute name	OCF Resource Property name	To OCF	From OCF

OnOff	oic.r.switch.binary:value	oic.r.switch.binary:value = OnOff	if oic.r.switch.binary:value == True:
			invoke(On)
			else:
			invoke(Off)

Table 6 Corresponding attributes of On/Off Cluster

Matter Cluster Attribute name	Type	Required	Description
OnOff	bool	Yes	The OnOff attribute indicates whether the device type implemented on the endpoint is turned off or turned on, in these cases the value of the OnOff attribute equals FALSE, or TRUE respectively.

Table 7 shows mapping between Matter Level Control Cluster and OCF dimming Resource. The "CurrentLevel" attribute of Level Control Cluster is read access only, so writing to the "CurrentLevel" attribute can only be done through "MoveToLevel" command provided by Level Control Cluster. "range" Property of "oic.r.light.dimming" Resource is read only, so there is no mapping from OCF. There is no corresponding OCF Property for "OnLevel" attribute, but "OnLevel" is mandatory attribute of Level Control Cluster, so it is updated when "dimmingSetting" of OCF dimming Resource is updated.

Table 7 Mapping for Matter Level Control Cluster

Matter	OCF Resource Property name	To OCF	From OCF

Cluster Attribute name			
CurrentLevel	oic.r.light.dimming:dimmingSetting	oic.r.light.dimming:dimmingSetting = CurrentLevel	oic.r.light.dimming:dimmingSetting = new_dimmingSetting invoke(MoveToLevel(oic.r.light.dimming:dimmingSetting))
MinLevel	oic.r.light.dimming:range[0]	oic.r.light.dimming:range[0] = MinLevel	N/A
MaxLevel	oic.r.light.dimming:range[1]	oic.r.light.dimming:range[1] = MaxLevel	N/A
OnLevel	oic.r.light.dimming:dimmingSetting	N/A	oic.r.light.dimming:dimmingSetting = new_dimmingSetting OnLevel = oic.r.light.dimming:dimmingSetting

Table 8 Corresponding attributes of Level Control Cluster

Matter Cluster Attribute name	Type	Required	Description
CurrentLevel	uint8	Yes	The CurrentLevel attribute represents the current level of this device. The meaning of 'level' is device dependent.
MinLevel	uint8	No	The MinLevel attribute indicates the minimum value of CurrentLevel that is capable of being assigned.
MaxLevel	uint8	No	The MaxLevel attribute indicates the maximum value of CurrentLevel that is capable of being assigned.
OnLevel	uint8	Yes	The OnLevel attribute determines the value that the CurrentLevel attribute is set to when the OnOff attribute of an On/Off cluster on the same endpoint is set to TRUE, as a result of processing an On/Off cluster command.

8.4.3 Derived model definition

```
{
  "id": "https://github.com/openconnectivityfoundation/OCF-Matter/DimmableLight.json#",
  "$schema": "http://json-schema.org/draft-04/schema#",
  "description" : "Copyright (c) 2023 Open Connectivity Foundation, Inc. All rights reserved.",
  "title": "On/Off Light Device Type",
  "definitions": {
    "OnOff" : {
      "description": "On/Off Cluster includes Attributes and commands for turning devices on and off",
      "type" : "object",
      "properties": {
        "OnOff": {
          "description": "OnOff attribute indicates whether the device type implemented on the
endpoint is turned off or turned on",
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.switch.binary:value",
            "x-to-ocf": [
              "oic.r.switch.binary:value = OnOff"
            ],
            "x-from-ocf": [
              "oic.r.switch.binary:value = new_value",
```

```
              "if oic.r.switch.binary:value == True:",
              "  invoke(On)",
              "else:",
              "  invoke(Off)"
            ]
          }
        }
      }
    },

    "LevelControl" : {
      "description": "Level Control cluster provides an interface for controlling a characteristic of
a device that can be set to a level, for example the brightness of a light, the degree of closure of a
door, or the power output of a heater. This Cluster is optional",
      "type" : "object",
      "properties": {
        "CurrentLevel": {
          "description": "CurrentLevel attribute represents the current level of this device. The
meaning of 'level' is device dependent.",
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.light.dimming:dimmingSetting",
            "x-to-ocf": [
              "oic.r.light.dimming:dimmingSetting = CurrentLevel"
            ],
            "x-from-ocf": [
              "oic.r.light.dimming:dimmingSetting = new_dimmingSetting",
              "invoke(MoveToLevel(oic.r.light.dimming:dimmingSetting))"
            ]
          }
        },
        "MinLevel": {
          "description": "MinLevel attribute indicates the minimum value of CurrentLevel that is
capable of being assigned.",
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.light.dimming:range",
            "x-to-ocf": [
              "oic.r.light.dimming:range[0] = MinLevel"
            ],
            "x-from-ocf": [
              "N/A"
            ]
          }
        },
        "MaxLevel": {
          "description": "MaxLevel attribute indicates the maximum value of CurrentLevel that is
capable of being assigned.",
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.light.dimming:range",
            "x-to-ocf": [
              "oic.r.light.dimming:range[1] = MaxLevel"
            ],
            "x-from-ocf": [
              "N/A"
            ]
          }
        },
        "OnLevel": {
          "description": "OnLevel attribute determines the value that the CurrentLevel attribute is
set to when the OnOff attribute of an On/Off cluster on the same endpoint is set to TRUE,",
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.light.dimming:dimmingSetting",
            "x-to-ocf": [
              "N/A"
            ],
            "x-from-ocf": [
```

```
              "oic.r.light.dimming:dimmingSetting = new_dimmingSetting",
              "OnLevel = oic.r.light.dimming:dimmingSetting"
            ]
          }
        }
      }
    }
  },

  "type": "object",

  "required": [
    "OnOff",
    "CurrentLevel",
    "MinLevel",
    "OnLevel"
  ]
}
```

8.5 Thermostat

8.5.1 Derived model

The derived model: "Thermostat.json"

8.5.2 Property definition

Table 9 shows mapping between Matter Thermostat Cluster and OCF temperature Resource. "temperature" Property of OCF temperature Resource plays 2 roles (temperature setting or measurement), but Thermostat Cluster has separate attribute for each of them.

The "LocalTemperature" attribute of Thermostat Cluster is read access only, but it is a measured temperature so no mapping for "from OCF" is necessary. On the other hand, "*Setpoint" attributes have read/write access, therefore they can be updated without using command.

Table 9 Mapping for Matter Thermostat Cluster

Matter Cluster Attribute name	OCF Resource Property name	To OCF	From OCF
LocalTemperature	oic.r.temperature:temperature	oic.r.temperature:temperature = LocalTemperature/100	N/A
OccupiedCoolingSetpoint	oic.r.temperature:temperature	oic.r.temperature:temperature = LocalTemperature/100	oic.r.temperature:temperature = new_temperature OccupiedCoolingSetpoint = oic.r.temperature:temperature * 100
OccupiedHeatingSetpoint	oic.r.temperature:temperature	oic.r.temperature:temperature = LocalTemperature/100	oic.r.temperature:temperature = new_temperature OccupiedCoolingSetpoint = oic.r.temperature:temperature * 100
UnoccupiedCoolingSetpoint	oic.r.temperature:temperature	oic.r.temperature:temperature = LocalTemperature/100	oic.r.temperature:temperature = new_temperature

			OccupiedCoolingSetpoint = oic.r.temperature:temperature * 100
UnoccupiedHeatingSetpoint	oic.r.temperature:temperature	oic.r.temperature:temperature = LocalTemperature/100	oic.r.temperature:temperature = new_temperature OccupiedCoolingSetpoint = oic.r.temperature:temperature * 100

Table 10 Corresponding attributes of Thermostat Cluster

Matter Cluster Attribute name	Type	Required	Description
LocalTemperature	int16 value = (temperature in °C) x 100 this value represents a temperature on the Celsius scale with a resolution of 0.01°C.	Yes	This attribute represents the temperature, as measured locally or remotely (over the network)
OccupiedCoolingSetpoint	int16 value = (temperature in °C) x 100 this value represents a temperature on the Celsius scale with a resolution of 0.01°C.	No	OccupiedCoolingSetpoint attribute specifies the cooling mode setpoint when the room is occupied
OccupiedHeatingSetpoint	int16 value = (temperature in °C) x 100 this value represents a temperature on the Celsius scale with a resolution of 0.01°C.	No	OccupiedHeatingSetpoint attribute specifies the heating mode setpoint when the room is occupied
UnoccupiedCoolingSetpoint	int16 value = (temperature in °C) x 100 this value represents a temperature on the Celsius scale with a resolution of 0.01°C.	No	UnoccupiedCoolingSetpoint attribute specifies the cooling mode setpoint when the room is unoccupied
UnoccupiedHeatingSetpoint	int16 value = (temperature in °C) x 100 this value represents a temperature on the Celsius scale with a resolution of 0.01°C.	No	UnoccupiedHeatingSetpoint attribute specifies the heating mode setpoint when the room is unoccupied.

8.5.3 Derived model definition

```
{
  "id": "https://github.com/openconnectivityfoundation/OCF-Matter/Thermostat.json#",
  "$schema": "http://json-schema.org/draft-04/schema#",
  "description" : "Copyright (c) 2023 Open Connectivity Foundation, Inc. All rights reserved.",
  "title": "Thermostat Device Type",
  "definitions": {
    "Thermostat" : {
      "description": "Thermostat cluster provides an interface to the functionality of a thermostat",
      "type" : "object",
      "properties": {
        "LocalTemperature": {
          "description": "LocalTemperature attribute represents the temperature, as measured locally
or remotely",
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.temperature:temperature",
            "x-to-ocf": [
              "oic.r.temperature:temperature = LocalTemperature/100"
            ],
            "x-from-ocf": [
              "N/A"
            ]
          }
        },
        "OccupiedCoolingSetpoint": {
          "description": "OccupiedCoolingSetpoint attribute specifies the cooling mode setpoint when
the room is occupied",
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.temperature:temperature",
            "x-to-ocf": [
              "oic.r.temperature:temperature = OccupiedCoolingSetpoint/100"
            ],
            "x-from-ocf": [
              "oic.r.temperature:temperature = new_temperature",
              "OccupiedCoolingSetpoint = oic.r.temperature:temperature * 100"
            ]
          }
        },
        "OccupiedHeatingSetpoint": {
          "description": "OccupiedHeatingSetpoint attribute specifies the heating mode setpoint when
the room is occupied",
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.temperature:temperature",
            "x-to-ocf": [
              "oic.r.temperature:temperature = OccupiedHeatingSetpoint/100"
            ],
            "x-from-ocf": [
              "oic.r.temperature:temperature = new_temperature",
              "OccupiedHeatingSetpoint = oic.r.temperature:temperature * 100"
            ]
          }
        },
        "UnoccupiedCoolingSetpoint": {
          "description": "UnoccupiedCoolingSetpoint attribute specifies the cooling mode setpoint when
the room is unoccupied",
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.temperature:temperature",
            "x-to-ocf": [
              "oic.r.temperature:temperature = UnoccupiedCoolingSetpoint/100"
            ],
            "x-from-ocf": [
```

```
              "oic.r.temperature:temperature = new_temperature",
              "UnoccupiedCoolingSetpoint = oic.r.temperature:temperature * 100"
            ]
          }
        },
        "UnoccupiedHeatingSetpoint": {
          "description": "UnoccupiedHeatingSetpoint attribute specifies the heating mode setpoint when
the room is unoccupied.",
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.temperature:temperature",
            "x-to-ocf": [
              "oic.r.temperature:temperature = UnoccupiedHeatingSetpoint/100"
            ],
            "x-from-ocf": [
              "oic.r.temperature:temperature = new_temperature",
              "UnoccupiedHeatingSetpoint = oic.r.temperature:temperature * 100"
            ]
          }
        }
      }
    }
  },

  "type": "object",

  "required": [
    "LocalTemperature"
  ]
}
```

8.6 Fan

8.6.1 Derived model

The derived model: "Fan.json"

8.6.2 Property definition

Table 11 shows mapping between Matter Fan Control Cluster and OCF binary switch Resource. The "FanMode" attribute of Fan Control Cluster has read/write access, therefore it can be updated without using command. However "FanMode" attribute is enum type, so it is necessary that default mapping rule for "FanMode" when "oic.r.switch.binary:value" is True. In this mapping, a default value of 2 (Medium Speed) is used to map to the True value of "oic.r.switch.binary:value".

Table 11 Mapping for Matter Fan Control Cluster

Matter Cluster Attribute name	OCF Resource Property name	To OCF	From OCF

FanMode	oic.r.switch.binary:value	if FanMode > 0:
		  oic.r.switch.binary:value = True
		else:
		  oic.r.switch.binary:value = False
			oic.r.switch.binary:value = new_value
			
			if oic.r.switch.binary:value > 0:
			  FanMode = 2
			else:
			  FanMode = oic.r.switch.binary:value

Table 12 Corresponding attributes of Fan Control Cluster

Matter Cluster Attribute name	Type	Required	Description
FanMode	enum8	Yes	This attribute indicate the current speed mode of the fan. This attribute is written by the client to indicate a new speed mode of the fan.

8.6.3 Derived model definition

```
{
  "id": "https://github.com/openconnectivityfoundation/OCF-Matter/Fan.json#",
  "$schema": "http://json-schema.org/draft-04/schema#",
  "description" : "Copyright (c) 2023 Open Connectivity Foundation, Inc. All rights reserved.",
  "title": "Fan Device Type",
  "definitions": {
    "FanControl" : {
      "description": "Fan Control cluster specifies an interface to control the speed of a fan",
      "type" : "object",
      "properties": {
        "FanMode": {
          "description": "FanMode attribute indicates the the current speed mode of the fan",
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.switch.binary:value",
            "x-to-ocf": [
              "if FanMode > 0:",
              "  oic.r.switch.binary:value = True",
              "else:",
              "  oic.r.switch.binary:value = False"
            ],
            "x-from-ocf": [
              "oic.r.switch.binary:value = new_value",
              "if oic.r.switch.binary:value > 0:",
              "  FanMode = 2",
              "else:",
              "  FanMode = oic.r.switch.binary:value"
            ]
          }
        }
      }
    }
  },

  "type": "object",

  "required": [
    "FanMode"
  ]
}
```