
**Information technology — Open
Connectivity Foundation (OCF)
Specification —**

**Part 3:
Bridging specification**

*Technologies de l'information — Specification de la Fondation pour la
connectivité ouverte (Fondation OCF) —*

Partie 3: Spécification de pontage





COPYRIGHT PROTECTED DOCUMENT

© ISO/IEC 2021

All rights reserved. Unless otherwise specified, or required in the context of its implementation, no part of this publication may be reproduced or utilized otherwise in any form or by any means, electronic or mechanical, including photocopying, or posting on the internet or an intranet, without prior written permission. Permission can be requested from either ISO at the address below or ISO's member body in the country of the requester.

ISO copyright office
CP 401 • Ch. de Blandonnet 8
CH-1214 Vernier, Geneva
Phone: +41 22 749 01 11
Email: copyright@iso.org
Website: www.iso.org

Published in Switzerland

Contents

Page

Foreword	iv
Introduction	v
1 Scope	1
2 Normative references	1
3 Terms, definitions, and abbreviated terms.....	1
3.1 Terms and definitions	1
3.2 Symbols and abbreviated terms	4
4 Document conventions and organization.....	4
4.1 Conventions	4
4.2 Notation.....	4
5 Introduction	5
5.1 Translation between OCF and non-OCF ecosystem - primitive concept of Bridging.....	5
5.2 Bridge platform	5
5.3 Symmetric vs. asymmetric bridging	7
5.4 General requirements.....	8
5.4.1 Requirements common to all bridge platforms	8
5.4.2 Requirements specific to symmetric bridge platforms	8
5.5 VOD list	9
5.6 Resource discovery	9
5.7 "Deep translation" vs. "on-the-fly"	14
5.8 Security	14
6 Device type definitions	14
7 Resource type definitions	14
7.1 List of resource types	14
7.2 VOD list	15
7.2.1 Introduction.....	15
7.2.2 Example URI.....	15
7.2.3 Resource type	15
7.2.4 OpenAPI 2.0 definition	15
7.2.5 Property definition	17
7.2.6 CRUDN behaviour	17

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of document should be noted (see www.iso.org/directives or www.iec.ch/members_experts/refdocs).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents) or the IEC list of patent declarations received (see patents.iec.ch).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT), see www.iso.org/iso/foreword.html. In the IEC, see www.iec.ch/understanding-standards.

This document was prepared by the Open Connectivity Foundation (OCF) (as OCF Bridging Framework Specification, version 2.2.0) and drafted in accordance with its editorial rules. It was adopted, under the JTC 1 PAS procedure, by Joint Technical Committee ISO/IEC JTC 1, *Information technology*.

This second edition cancels and replaces the first edition (ISO/IEC 30118-3:2018), which has been technically revised.

The main changes compared to the previous edition are as follows:

- bridging specification has been made more generic;
- text moved from AllJoyn mapping to the resource to Resource to AllJoyn interface mapping specification;
- addition of clarifications throughout.

A list of all parts in the ISO/IEC 30118 series can be found on the ISO and IEC websites.

Any feedback or questions on this document should be directed to the user's national standards body. A complete listing of these bodies can be found at www.iso.org/members.html and www.iec.ch/national-committees.

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.

ISO/IEC 30118 consists of eighteen parts, under the general title Information technology — Open Connectivity Foundation (OCF) Specification. The parts fall into logical groupings as described herein:

- Core framework
 - Part 1: Core Specification
 - Part 2: Security Specification
 - Part 13: Onboarding Tool Specification
- Bridging framework and bridges
 - Part 3: Bridging Specification
 - Part 6: Resource to Alljoyn Interface Mapping Specification
 - Part 8: OCF Resource to oneM2M Resource Mapping Specification
 - Part 14: OCF Resource to BLE Mapping Specification
 - Part 15: OCF Resource to EnOcean Mapping Specification
 - Part 16: OCF Resource to UPlus Mapping Specification
 - Part 17: OCF Resource to Zigbee Cluster Mapping Specification
 - Part 18: OCF Resource to Z-Wave Mapping Specification
- Resource and Device models
 - Part 4: Resource Type Specification
 - Part 5: Device Specification

- Core framework extensions
 - Part 7: Wi-Fi Easy Setup Specification
 - Part 9: Core Optional Specification
- OCF Cloud
 - Part 10: Cloud API for Cloud Services Specification
 - Part 11: Device to Cloud Services Specification
 - Part 12: Cloud Security Specification

Information technology — Open Connectivity Foundation (OCF) Specification —

Part 3: Bridging specification

1 Scope

This document specifies a framework for translation between OCF Devices and other ecosystems, and specifies the behaviour of a Bridging Function 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 documents (deep translation). This document provides generic requirements that apply unless overridden by a more specific document.

2 Normative references

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

ISO/IEC 30118-1 *Information technology -- Open Connectivity Foundation (OCF) Specification -- Part 1: Core specification*

<https://www.iso.org/standard/53238.html>

ISO/IEC 30118-2 *Information technology -- Open Connectivity Foundation (OCF) Specification -- Part 2: Security specification*

<https://www.iso.org/standard/74239.html>

JSON Schema Core, *JSON Schema: core definitions and terminology*, January 2013

<http://json-schema.org/latest/json-schema-core.html>

OpenAPI Specification, Version 2.0

<https://github.com/OAI/OpenAPI-Specification/blob/master/versions/2.0.md>

3 Terms, definitions, and abbreviated terms

3.1 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO/IEC 30118-1 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

Asymmetric Client Bridge

Bridge (3.1.3) that exposes another ecosystem's clients into the OCF ecosystem as *Virtual OCF Clients* (3.1.21). This is equivalent to exposing *OCF Servers* (3.1.18) into the other ecosystem.

3.1.2

Asymmetric Server Bridge

Bridge that exposes another ecosystem devices into the OCF ecosystem as *Virtual OCF Servers* (3.1.18). How this is handled in each ecosystem is specified on a per ecosystem basis in this document.

3.1.3

Bridge

OCF Device that has a Device Type of "oic.d.bridge", provides information on the set of *Virtual OCF Devices* (3.1.22) that are resident on the same Bridge Platform.

3.1.4

Bridge Platform

entity on which the *Bridge* (3.1.3) and *Virtual OCF Devices* (3.1.22) are resident

3.1.5

Bridged Client

logical entity that accesses data via a *Bridged Protocol* (3.1.7)

3.1.6

Bridged Device

Bridged Client (3.1.5) or *Bridged Server* (3.1.10).

3.1.7

Bridged Protocol

another protocol that is being translated to or from OCF protocols

3.1.8

Bridged Resource

artefact modelled and exposed by a *Bridged Protocol* (3.1.7)

3.1.9

Bridged Resource Type

schema used with a *Bridged Protocol* (3.1.7)

3.1.10

Bridged Server

logical entity that provides data via a *Bridged Protocol* (3.1.7). More than one Bridged Server can exist on the same physical platform.

3.1.11

Bridging Function

logic resident on the *Bridge Platform* (3.1.4) that performs that protocol mapping between OCF and the *Bridged Protocol* (3.1.7); a *Bridge Platform* (3.1.4) may contain multiple Bridging Functions dependent on the number of *Bridged Protocols* (3.1.7) supported.

3.1.12

OCF Bridge Device

OCF Device (3.1.14) that can represent devices that exist on the network but communicate using a *Bridged Protocol* (3.1.7) rather than OCF protocols.

3.1.13**OCF Client**

logical entity that accesses an *OCF Resource* (3.1.15) on an *OCF Server* (3.1.18), which might be a *Virtual OCF Server* (3.1.24) exposed by the *OCF Bridge Device* (3.1.12)

3.1.14**OCF Device**

logical entity that assumes one or more OCF. More than one OCF Device can exist on the same physical platform

3.1.15**OCF Resource**

represents an artefact modelled and exposed by the OCF Framework

3.1.16**OCF Resource Property**

significant aspect or notion including metadata that is exposed through the *OCF Resource* (3.1.15)

3.1.17**OCF Resource Type**

OCF Resource Property (3.1.16) that represents the data type definition for the *OCF Resource* (3.1.15)

3.1.18**OCF Server**

logical entity with the role of providing resource state information and allowing remote control of its resources

3.1.19**Virtual Bridged Client**

logical representation of an *OCF Client* (3.1.13), which an *OCF Bridge Device* (3.1.12) exposes to *Bridged Servers* (3.1.10).

3.1.20**Virtual Bridged Server**

logical representation of an *OCF Server* (3.1.18), which an *OCF Bridge Device* (3.1.12) exposes to *Bridged Clients* (3.1.5).

3.1.21**Virtual OCF Client**

logical representation of a *Bridged Client* (3.1.5), which an *OCF Bridge Device* (3.1.12) exposes to *OCF Servers* (3.1.18)

3.1.22**Virtual OCF Device**

Virtual OCF Client (3.1.21) or *Virtual OCF Server* (3.1.24).

3.1.23**Virtual OCF Resource**

logical representation of a *Bridged Resource* (3.1.8), which an *OCF Bridge Device* (3.1.12) exposes to *OCF Clients* (3.1.13)

3.1.24**Virtual OCF Server**

logical representation of a *Bridged Server* (3.1.10), which an *OCF Bridge Device* (3.1.12) exposes to *OCF Clients* (3.1.13).

3.2 Symbols and abbreviated terms

CRUDN Create, Read, Update, Delete, and Notify

4 Document conventions and organization

4.1 Conventions

In this document a number of terms, conditions, mechanisms, sequences, parameters, events, states, or similar terms are printed with the first letter of each word in uppercase and the rest lowercase (e.g., Network Architecture). Any lowercase uses of these words have the normal technical English meaning.

In this document, to be consistent with the IETF usages for RESTful operations, the RESTful operation words CRUDN, CREATE, RETRIVE, UPDATE, DELETE, and NOTIFY will have all letters capitalized. Any lowercase uses of these words have the normal technical English meaning.

4.2 Notation

In this document, features are described as required, recommended, allowed or DEPRECATED as follows:

Required (or shall or mandatory).

- These basic features shall be implemented to comply with this document. The phrases “shall not”, and “PROHIBITED” indicate behaviour that is prohibited, i.e. that if performed means the implementation is not in compliance.

Recommended (or should).

- These features add functionality supported by this document and should be implemented. Recommended features take advantage of the capabilities of this document, usually without imposing major increase of complexity. Notice that for compliance testing, if a recommended feature is implemented, it shall meet the specified requirements to be in compliance with these guidelines. Some recommended features could become requirements in the future. The phrase “should not” indicates behaviour that is permitted but not recommended.

Allowed (or allowed).

- These features are neither required nor recommended by this document, but if the feature is implemented, it shall meet the specified requirements to be in compliance with these guidelines.
- Conditionally allowed (CA)The definition or behaviour depends on a condition. If the specified condition is met, then the definition or behaviour is allowed, otherwise it is not allowed.

Conditionally required (CR)

- The definition or behaviour depends on a condition. If the specified condition is met, then the definition or behaviour is required. Otherwise the definition or behaviour is allowed as default unless specifically defined as not allowed.

DEPRECATED

- Although these features are still described in this document, they should not be implemented except for backward compatibility. The occurrence of a deprecated feature during operation of an implementation compliant with the current document has no effect on the implementation's operation and does not produce any error conditions. Backward compatibility may require that a feature is implemented and functions as specified but it shall never be used by implementations compliant with this document.

Strings that are to be taken literally are enclosed in "double quotes".

Words that are emphasized are printed in *italic*.

5 Introduction

5.1 Translation between OCF and non-OCF ecosystem - primitive concept of Bridging

The details of Bridging may be implemented in many ways, for example, by using a Bridge Platform with an entity handler to interface directly to a Non-OCF device as shown in Figure 1.



Figure 1 – Server bridging to Non- OCF device

On start-up the Bridge Platform runs the entity handlers which discover the non-OCF systems (e.g., Heart Rate Sensor Device) and create Resources for each Device or functionality discovered. The entity handler creates a Resource for each discovered Device or functionality and binds itself to that Resource. These Resources are made discoverable by the Bridge Platform.

Once the Resources are created and made discoverable, then the Client Device can discover these Resources and operate on them using the mechanisms described in ISO/IEC 30118-1. The requests to a Resource on the Bridge Platform are then interpreted by the entity handler and forwarded to the non-OCF device using the protocol supported by the non-OCF device. The returned information from the non-OCF device is then mapped to the appropriate response for that Resource.

Current OCF Bridging architecture implements the entity handler in the form of VOD.

5.2 Bridge platform

This clause describes the functionality of a Bridge Platform; such a device is illustrated in Figure 2.

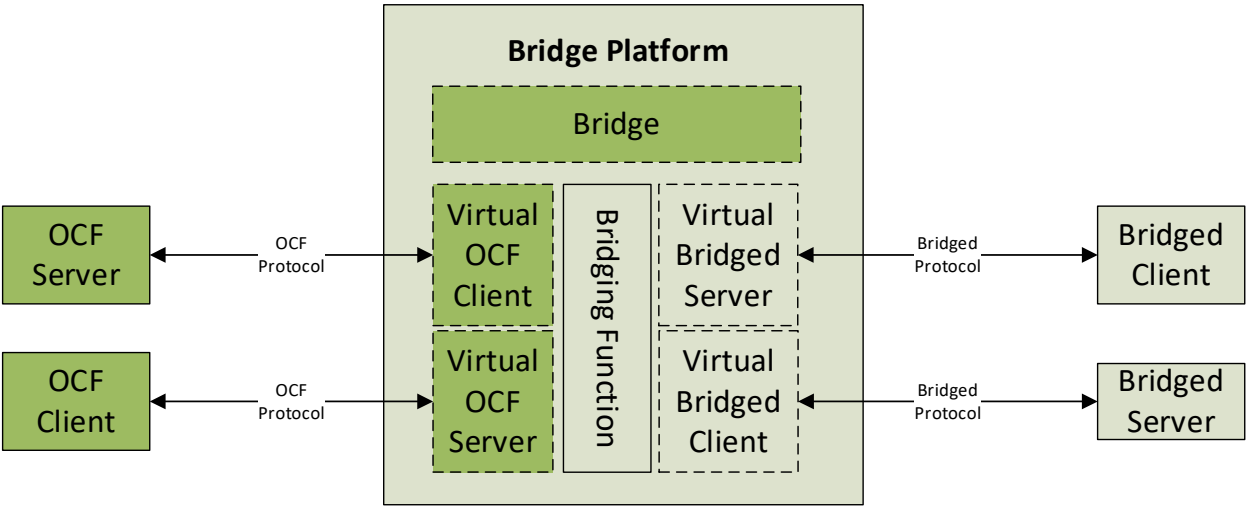


Figure 2 – Bridge Platform components

A Bridge Platform enables the representation of one or more Bridged Devices as Virtual OCF Devices (VODs) on the network and/or enables the representation of one or more OCF Devices as Virtual OCF Devices using another protocol on the network. The Bridged Devices themselves are out of the scope of this document. The only difference between a native OCF Device and a VOD from the perspective of an OCF Client is the inclusion of "oic.d.virtual" in the "rt" of "/oic/d" of the VOD.

A Bridge Platform exposes a Bridge Device which is an OCF Device with a Device Type of "oic.d.bridge". This provides to an OCF Client an explicit indication that the discovered Device is performing a bridging function. This is useful for several reasons; 1) when establishing a home network, the Client can determine that the bridge is reachable and functional when no bridged devices are present, 2) allows for specific actions to be performed on the bridge considering the known functionality a bridge supports, 3) allows for explicit discovery of all devices that are serving a bridging function which benefits trouble shooting and maintenance actions on behalf of a user. When such a device is discovered the exposed Resources on the OCF Bridge Device describe other devices. For example, as shown in Figure 3.

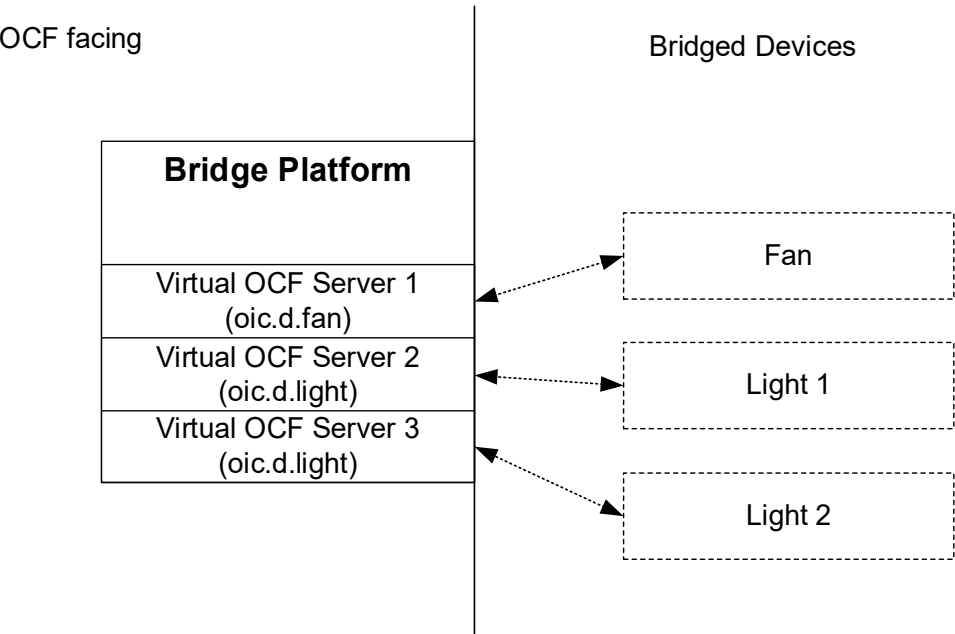


Figure 3 – Schematic overview of a Bridge Platform bridging non-OCF devices

It is expected that the Bridge Platform creates a set of devices during the start-up of the Bridge Platform, these being the Bridge and any known VODs. The exposed set of VODs can change as Bridged Devices are added or removed from the bridge. The adding and removing of Bridged Devices is implementation dependent.

5.3 Symmetric vs. asymmetric bridging

There are two kinds of bridging: symmetric and asymmetric. In symmetric bridging, a bridge device exposes OCF server(s) to another ecosystem and exposes other ecosystem's server(s) to OCF. In asymmetric bridging, a bridge device exposes OCF server(s) to another ecosystem or exposes another ecosystem's server(s) to OCF, but not both. The former case is called an Asymmetric Server Bridge (see Figure 4), the latter case is called an Asymmetric Client Bridge (see Figure 5)

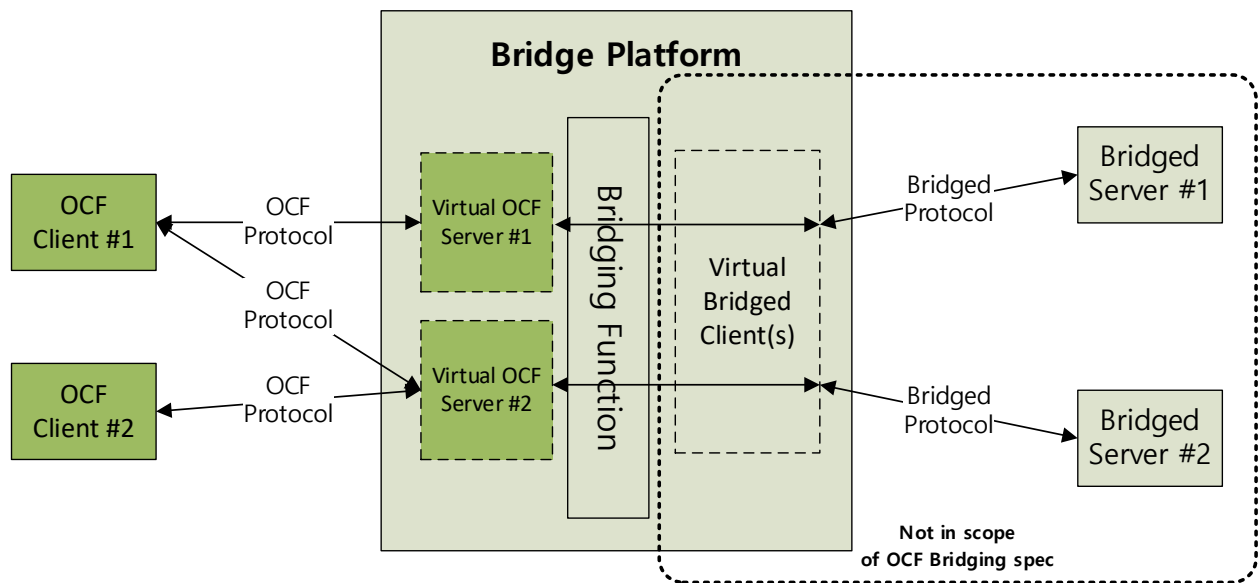


Figure 4 – Asymmetric server bridge

In Figure 4 each Bridged Server is exposed as a Virtual OCF Server to OCF side. These Virtual OCF Servers are same as normal OCF Servers except that they have additional rt value ("oic.d.virtual") for "/oic/d". The details of the Virtual Bridged Client are not in scope of this document.

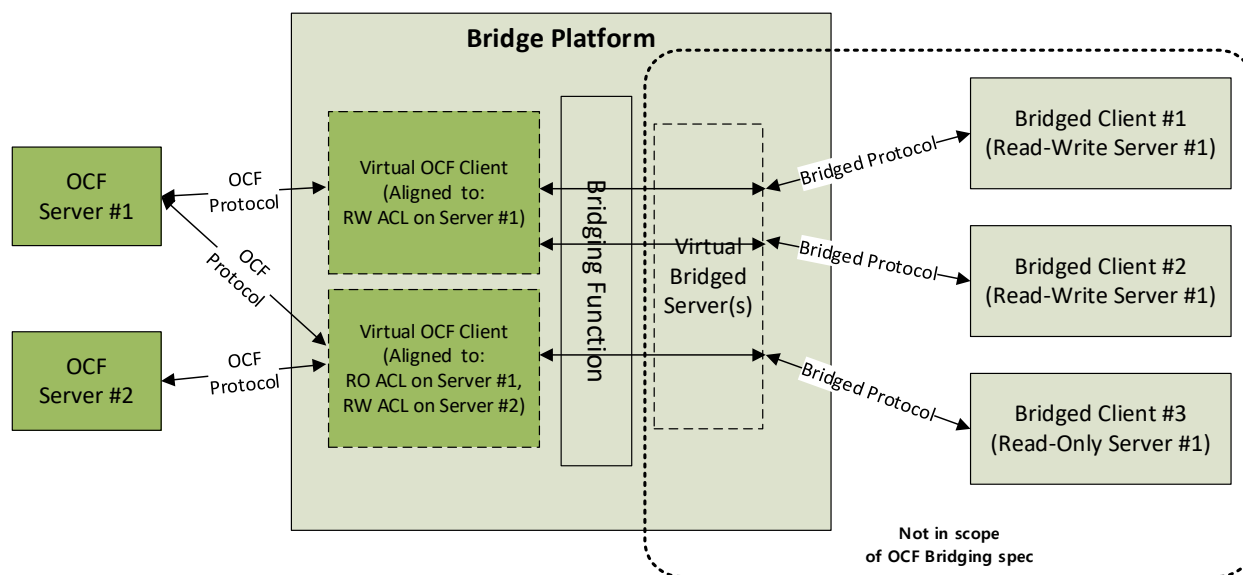


Figure 5 – Asymmetric client bridge

Figure 5 shows that each access to the OCF Server is modelled as a Virtual OCF Client. Those accesses can be aggregated if their target OCF servers and access permissions are same, therefore a Virtual OCF Client can tackle multiple Bridged Clients.

5.4 General requirements

5.4.1 Requirements common to all bridge platforms

A VOD shall have a Device Type that contains "oic.d.virtual". This allows Bridge Platforms to determine if a device is already being translated when multiple Bridge Platforms are present or Clients to determine if corresponding Server is a VOD or not.

Each Bridged Device shall be exposed as a separate Virtual OCF Server or Client, with its own OCF Endpoint, and set of mandatory Resources (as defined in ISO/IEC 30118-1 and ISO/IEC 30118-2).

Discovery of a VOD is the same as for an ordinary OCF Device; that is the VOD shall respond to multicast discovery requests. This allows platform-specific, device-specific, and resource-specific fields to all be preserved across translation.

The Bridge Introspection Device Data (IDD) provides information for the Resources exposed by the Bridge only. Each VOD shall expose an instance of "oic.wk.introspection" which provides a URL to an IDD for the specific VOD.

5.4.2 Requirements specific to symmetric bridge platforms

In addition to the requirements mentioned in 5.4.1, Symmetric Bridging shall satisfy following requirements.

The Bridge Platform shall check the protocol-independent UUID ("piid" in OCF) of each device and shall not advertise back into a Bridged Protocol a device originally seen via that Bridged Protocol. The Bridge Platform shall stop translating any Bridged Protocol device exposed in OCF via another Bridge Platform if the Bridge Platform sees the device via the Bridged Protocol. Similarly, the Bridge Platform shall not advertise an OCF Device back into OCF, and the Bridge Platform shall stop translating any OCF device exposed in the Bridged Protocol via another Bridge Platform if the Bridge Platform sees the device via OCF. These require that the Bridge Platform can determine when a device is already being translated. A VOD shall be indicated on the OCF Security Domain with a Device Type of "oic.d.virtual".

The Bridge Platform shall detect duplicate VODs (with the same protocol-independent UUID) present in a network and shall not create more than one corresponding virtual device as it translates those duplicate devices into another network.

5.5 VOD list

For maintenance purposes, the Bridge maintains a list of VODs. This list includes Virtual OCF Servers and Virtual OCF Clients created by the Bridge Platform and subsequently on-boarded, as specified in ISO/IEC 30118-2. A single instance of the Resource Type that defines the VOD list (see clause 7.2) shall be exposed by the Bridge. Please refer to ISO/IEC 30118-2 for detailed operational requirements for the VOD list.

5.6 Resource discovery

A Bridge Platform shall detect devices that arrive and leave the Bridged network or the OCF Security Domain. Where there is no pre-existing mechanism to reliably detect the arrival and departure of devices on a network, a Bridge Platform shall periodically poll the network to detect the arrival and departure of devices, for example using COAP multicast discovery (a multicast RETRIEVE of "/oic/res") in the case of the OCF Security Domain. Bridge Platform implementations are encouraged to use a poll interval of 30 seconds plus or minus a random delay of a few seconds.

A Bridge Platform and any exposed VODs shall each respond to network discovery commands. The response to a RETRIEVE on "/oic/res" shall only include the devices that match the RETRIEVE request.

For example, if a Bridge exposes VODs for the fan and lights shown in Figure 3, and an OCF Client performs a discovery request with a content format of "application/vnd.ocf+cbor", there will be four discrete responses, one for the Bridge, one for the virtual fan Device, and two for the virtual light Devices. Note that what is returned is not in the JSON format (see JSON Schema Core) but in a suitable encoding as defined in ISO/IEC 30118-1.

Response from the Bridge:

```
[
  {
    "anchor": "ocf://e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",
    "href": "/oic/res",
    "rel": "self",
    "rt": ["oic.wk.res"],
    "if": ["oic.if.ll", "oic.if.baseline"],
    "p": {"bm": 3},
    "eps": [{"ep": "coap://[2001:db8:a::b1d4]:5555"},
            {"ep": "coaps://[2001:db8:a::b1d4]:1111"}]
  },
  {
    "anchor": "ocf://e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",
    "href": "/oic/d",
    "rt": ["oic.wk.d", "oic.d.bridge"],
    "if": ["oic.if.r", "oic.if.baseline"],
    "p": {"bm": 3},
    "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:1111"}]
  },
  {
    "anchor": "ocf://e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",
    "href": "/oic/p",
    "rt": ["oic.wk.p"],
    "if": ["oic.if.r", "oic.if.baseline"],
    "p": {"bm": 3},
    "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:1111"}]
  },
  {
    "anchor": "ocf://e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",
    "href": "/oic/sec/doxm",
    "rt": ["oic.r.doxm"],
    "if": ["oic.if.baseline"],
  }
]
```

```

    "p": {"bm": 1},
    "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:11111"}]
  },
  {
    "anchor": "ocf://e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",
    "href": "/oic/sec/pstat",
    "rt": ["oic.r.pstat"],
    "if": ["oic.if.baseline"],
    "p": {"bm": 1},
    "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:11111"}]
  },
  {
    "anchor": "ocf://e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",
    "href": "/oic/sec/cred",
    "rt": ["oic.r.cred"],
    "if": ["oic.if.baseline"],
    "p": {"bm": 1},
    "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:11111"}]
  },
  {
    "anchor": "ocf://e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",
    "href": "/oic/sec/acl2",
    "rt": ["oic.r.acl2"],
    "if": ["oic.if.baseline"],
    "p": {"bm": 1},
    "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:11111"}]
  },
  {
    "anchor": "ocf://e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",
    "href": "/myIntrospection",
    "rt": ["oic.wk.introspection"],
    "if": ["oic.if.r", "oic.if.baseline"],
    "p": {"bm": 3},
    "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:11111"}]
  },
  {
    "anchor": "ocf://e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",
    "href": "/myVodlist",
    "rt": ["oic.r.vodlist"],
    "if": ["oic.if.r", "oic.if.baseline"],
    "p": {"bm": 3},
    "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:11111"}]
  }
]

```

Response from the Fan VOD:

```

[
  {
    "anchor": "ocf://88b7c7f0-4b51-4e0a-9faa-cfb439fd7f49",
    "href": "/oic/res",
    "rt": ["oic.wk.res"],
    "if": ["oic.if.ll", "oic.if.baseline"],
    "p": {"bm": 3},
    "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:22222"}]
  },
  {
    "anchor": "ocf://88b7c7f0-4b51-4e0a-9faa-cfb439fd7f49",
    "href": "/oic/d",
    "rt": ["oic.wk.d", "oic.d.fan", "oic.d.virtual"],
    "if": ["oic.if.r", "oic.if.baseline"],
    "p": {"bm": 3},
    "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:22222"}]
  },
  {
    "anchor": "ocf://88b7c7f0-4b51-4e0a-9faa-cfb439fd7f49",
    "href": "/oic/p",
    "rt": ["oic.wk.p"],
    "if": ["oic.if.r", "oic.if.baseline"],
    "p": {"bm": 3},

```

```

    "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:22222"}]
  },
  {
    "anchor": "ocf://88b7c7f0-4b51-4e0a-9faa-cfb439fd7f49",
    "href": "/myFan",
    "rt": ["oic.r.switch.binary"],
    "if": ["oic.if.a", "oic.if.baseline"],
    "p": {"bm": 3},
    "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:22222"}]
  },
  {
    "anchor": "ocf://88b7c7f0-4b51-4e0a-9faa-cfb439fd7f49",
    "href": "/oic/sec/doxm",
    "rt": ["oic.r.doxm"],
    "if": ["oic.if.baseline"],
    "p": {"bm": 1},
    "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:22222"}]
  },
  {
    "anchor": "ocf://88b7c7f0-4b51-4e0a-9faa-cfb439fd7f49",
    "href": "/oic/sec/pstat",
    "rt": ["oic.r.pstat"],
    "if": ["oic.if.baseline"],
    "p": {"bm": 1},
    "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:22222"}]
  },
  {
    "anchor": "ocf://88b7c7f0-4b51-4e0a-9faa-cfb439fd7f49",
    "href": "/oic/sec/cred",
    "rt": ["oic.r.cred"],
    "if": ["oic.if.baseline"],
    "p": {"bm": 1},
    "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:22222"}]
  },
  {
    "anchor": "ocf://88b7c7f0-4b51-4e0a-9faa-cfb439fd7f49",
    "href": "/oic/sec/acl2",
    "rt": ["oic.r.acl2"],
    "if": ["oic.if.baseline"],
    "p": {"bm": 1},
    "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:22222"}]
  },
  {
    "anchor": "ocf://88b7c7f0-4b51-4e0a-9faa-cfb439fd7f49",
    "href": "/myFanIntrospection",
    "rt": ["oic.wk.introspection"],
    "if": ["oic.if.r", "oic.if.baseline"],
    "p": {"bm": 3},
    "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:22222"}]
  }
]

```

Response from the first Light VOD:

```

[
  {
    "anchor": "ocf://dc70373c-1e8d-4fb3-962e-017eaa863989",
    "href": "/oic/res",
    "rt": ["oic.wk.res"],
    "if": ["oic.if.ll", "oic.if.baseline"],
    "p": {"bm": 3},
    "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:33333"}]
  },
  {
    "anchor": "ocf://dc70373c-1e8d-4fb3-962e-017eaa863989",
    "href": "/oic/d",
    "rt": ["oic.wk.d", "oic.d.light", "oic.d.virtual"],
    "if": ["oic.if.r", "oic.if.baseline"],
    "p": {"bm": 3},
    "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:33333"}]
  }
]

```

```

},
{
  "anchor": "ocf://dc70373c-1e8d-4fb3-962e-017eaa863989",
  "href": "/oic/p",
  "rt": ["oic.wk.p"],
  "if": ["oic.if.r", "oic.if.baseline"],
  "p": {"bm": 3},
  "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:3333"}]
},
{
  "anchor": "ocf://dc70373c-1e8d-4fb3-962e-017eaa863989",
  "href": "/myLight",
  "rt": ["oic.r.switch.binary"],
  "if": ["oic.if.a", "oic.if.baseline"],
  "p": {"bm": 3},
  "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:3333"}]
},
{
  "anchor": "ocf://dc70373c-1e8d-4fb3-962e-017eaa863989",
  "href": "/oic/sec/doxm",
  "rt": ["oic.r.doxm"],
  "if": ["oic.if.baseline"],
  "p": {"bm": 1},
  "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:3333"}]
},
{
  "anchor": "ocf://dc70373c-1e8d-4fb3-962e-017eaa863989",
  "href": "/oic/sec/pstat",
  "rt": ["oic.r.pstat"],
  "if": ["oic.if.baseline"],
  "p": {"bm": 1},
  "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:3333"}]
},
{
  "anchor": "ocf://dc70373c-1e8d-4fb3-962e-017eaa863989",
  "href": "/oic/sec/cred",
  "rt": ["oic.r.cred"],
  "if": ["oic.if.baseline"],
  "p": {"bm": 1},
  "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:3333"}]
},
{
  "anchor": "ocf://dc70373c-1e8d-4fb3-962e-017eaa863989",
  "href": "/oic/sec/acl2",
  "rt": ["oic.r.acl2"],
  "if": ["oic.if.baseline"],
  "p": {"bm": 1},
  "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:3333"}]
},
{
  "anchor": "ocf://dc70373c-1e8d-4fb3-962e-017eaa863989",
  "href": "/myLightIntrospection",
  "rt": ["oic.wk.introspection"],
  "if": ["oic.if.r", "oic.if.baseline"],
  "p": {"bm": 3},
  "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:3333"}]
}
]

```

Response from the second Light VOD:

```

[
  {
    "anchor": "ocf://8202138e-aa22-452c-b512-9ebad02bef7c",
    "href": "/oic/res",
    "rt": ["oic.wk.res"],
    "if": ["oic.if.ll", "oic.if.baseline"],
    "p": {"bm": 3},
    "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:4444"}]
  },

```

```

{
  "anchor": "ocf://8202138e-aa22-452c-b512-9ebad02bef7c",
  "href": "/oic/d",
  "rt": ["oic.wk.d", "oic.d.light", "oic.d.virtual"],
  "if": ["oic.if.r", "oic.if.baseline"],
  "p": {"bm": 3},
  "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:44444"}]
},
{
  "anchor": "ocf://8202138e-aa22-452c-b512-9ebad02bef7c",
  "href": "/oic/p",
  "rt": ["oic.wk.p"],
  "if": ["oic.if.r", "oic.if.baseline"],
  "p": {"bm": 3},
  "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:44444"}]
},
{
  "anchor": "ocf://8202138e-aa22-452c-b512-9ebad02bef7c",
  "href": "/myLight",
  "rt": ["oic.r.switch.binary"],
  "if": ["oic.if.a", "oic.if.baseline"],
  "p": {"bm": 3},
  "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:44444"}]
},
{
  "anchor": "ocf://8202138e-aa22-452c-b512-9ebad02bef7c",
  "href": "/oic/sec/doxm",
  "rt": ["oic.r.doxm"],
  "if": ["oic.if.baseline"],
  "p": {"bm": 1},
  "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:44444"}]
},
{
  "anchor": "ocf://8202138e-aa22-452c-b512-9ebad02bef7c",
  "href": "/oic/sec/pstat",
  "rt": ["oic.r.pstat"],
  "if": ["oic.if.baseline"],
  "p": {"bm": 1},
  "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:44444"}]
},
{
  "anchor": "ocf://8202138e-aa22-452c-b512-9ebad02bef7c",
  "href": "/oic/sec/cred",
  "rt": ["oic.r.cred"],
  "if": ["oic.if.baseline"],
  "p": {"bm": 1},
  "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:44444"}]
},
{
  "anchor": "ocf://8202138e-aa22-452c-b512-9ebad02bef7c",
  "href": "/oic/sec/acl2",
  "rt": ["oic.r.acl2"],
  "if": ["oic.if.baseline"],
  "p": {"bm": 1},
  "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:44444"}]
},
{
  "anchor": "ocf://8202138e-aa22-452c-b512-9ebad02bef7c",
  "href": "/myLightIntrospection",
  "rt": ["oic.wk.introspection"],
  "if": ["oic.if.r", "oic.if.baseline"],
  "p": {"bm": 3},
  "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:44444"}]
}
]

```

Figure 6 – /oic/res example responses

5.7 "Deep translation" vs. "on-the-fly"

When translating a service between a Bridged Protocol (e.g., AllJoyn) and OCF protocols, there are two possible types of translation. Bridge Platforms are expected to dedicate most of their logic to "deep translation" types of communication, in which data models used with the Bridged Protocol are mapped to the equivalent OCF Resource Types and vice-versa, in such a way that a compliant OCF Client or Bridged Client would be able to interact with the service without realising that a translation was made.

"Deep translation" is out of the scope of this document, as the procedure far exceeds mapping of types. For example, clients on one side of a Bridge Platform may decide to represent an intensity as an 8-bit value between 0 and 255, whereas the devices on the other may have chosen to represent that as a floating-point number between 0.0 and 1.0. It's also possible that the procedure may require storing state in the Bridge Platform. Either way, the programming of such translation will require dedicated effort and study of the mechanisms on both sides.

The other type of translation, the "on-the-fly" or "one-to-one" translation, requires no prior knowledge of the device-specific schema in question on the part of the Bridge Platform. The burden is, instead, on one of the other participants in the communication, usually the client application. That stems from the fact that "on-the-fly" translation always produces Bridged Resource Types and OCF Resource Types as vendor extensions.

5.8 Security

Please refer to ISO/IEC 30118-2 for security specific requirements as they pertain to a Bridge Platform. These security requirements include both universal requirements applicable to all Bridged Protocols, and additional security requirements specific to each Bridged Protocol.

6 Device type definitions

The required Resource Types are listed in Table 1.

Table 1 – Device type definitions

Device Name (informative)	Device Type ("rt") (Normative)	Required Resource name	Required Resource Type
Bridge	oic.d.bridge	Secure Mode	oic.r.securemode
Virtual Device	oic.d.virtual	Device	oic.wk.d

7 Resource type definitions

7.1 List of resource types

Table 2 lists the Resource Types defined in this document.

Table 2 – Alphabetical list of resource types

Friendly Name (informative)	Resource Type (rt)	Clause
VOD List	oic.r.vodlist	10.4

7.2 VOD list

7.2.1 Introduction

This Resource describes the VODs that have been onboarded on the Bridge Platform.

7.2.2 Example URI

/VODListResURI

7.2.3 Resource type

The Resource Type is defined as: "oic.r.vodlist".

7.2.4 OpenAPI 2.0 definition

```
{
  "swagger": "2.0",
  "info": {
    "title": "VOD List",
    "version": "2019-05-16",
    "license": {
      "name": "OCF Data Model License",
      "url":
"https://github.com/openconnectivityfoundation/core/blob/e28a9e0a92e17042ba3e83661e4c0fbce8bdc4ba/LICEN
SE.md",
      "x-copyright": "Copyright 2019 Open Connectivity Foundation, Inc. All rights reserved."
    },
    "termsOfService": "https://openconnectivityfoundation.github.io/core/DISCLAIMER.md"
  },
  "schemes": ["http"],
  "consumes": ["application/json"],
  "produces": ["application/json"],
  "paths": {
    "/VODListResURI" : {
      "get": {
        "description": "This Resource describes the VODs that have been onboarded on the Bridge
Platform.\n",
        "parameters": [
          {"$ref": "#/parameters/interface"}
        ],
        "responses": {
          "200": {
            "description": "Example response payload",
            "x-example": {
              "rt": ["oic.r.vodlist"],
              "vods": [
                {
                  "n": "Smoke sensor",
                  "di": "54919CA5-4101-4AE4-595B-353C51AA1234",
                  "econame": "Z-Wave"
                },
                {
                  "n": "Thermostat",
                  "di": "54919CA5-4101-4AE4-595B-353C51AA5678",
                  "econame": "Zigbee"
                }
              ]
            }
          },
          "schema": { "$ref": "#/definitions/vodlist" }
        }
      }
    }
  },
  "parameters": {
    "interface": {
      "in": "query",
      "name": "if",

```

```

        "type" : "string",
        "enum" : ["oic.if.r", "oic.if.baseline"]
    },
    },
    "definitions": {
        "vodentry" : {
            "description": "Information for a VOD created by the Bridge",
            "type": "object",
            "properties": {
                "n": {
                    "$ref":
"https://openconnectivityfoundation.github.io/core/schemas/oic.common.properties.core-
schema.json#/definitions/n"
                },
                "di" : {
                    "$ref": "https://openconnectivityfoundation.github.io/core/schemas/oic.types-
schema.json#/definitions/uuid"
                },
                "econame": {
                    "description": "Ecosystem Name of the Bridged Device which is exposed by this VOD",
                    "type": "string",
                    "enum": [ "BLE", "oneM2M", "UPlus", "Zigbee", "Z-Wave" ],
                    "readOnly": true
                }
            }
        },
        "required": ["n", "di", "econame"]
    },
    "vodlist": {
        "type": "object",
        "properties": {
            "n": {
                "$ref":
"https://openconnectivityfoundation.github.io/core/schemas/oic.common.properties.core-
schema.json#/definitions/n"
            },
            "rt" : {
                "description": "Resource Type",
                "items": {
                    "maxLength": 64,
                    "type": "string",
                    "enum": ["oic.r.vodlist"]
                },
                "minItems": 1,
                "uniqueItems": true,
                "readOnly": true,
                "type": "array"
            },
            "id": {
                "$ref":
"https://openconnectivityfoundation.github.io/core/schemas/oic.common.properties.core-
schema.json#/definitions/id"
            },
            "if" : {
                "description": "The OCF Interface set supported by this Resource",
                "items": {
                    "enum": [
                        "oic.if.baseline",
                        "oic.if.r"
                    ],
                    "type": "string"
                },
                "minItems": 2,
                "uniqueItems": true,
                "readOnly": true,
                "type": "array"
            },
            "vods": {
                "description": "Array of information per VOD created by the Bridge",
                "type": "array",
                "minItems": 0,
                "uniqueItems": true,
                "readOnly": true,
                "items": {
                    "$ref": "#/definitions/vodentry"
                }
            }
        }
    },
    },

```

```

    "required": ["vods"]
  }
}
}

```

7.2.5 Property definition

Table 3 defines the Properties that are part of the "oic.r.vodlist" Resource Type.

Table 3 – The Property definitions of the Resource with type "rt" = "oic.r.vodlist"

Property name	Value type	Mandatory	Access mode	Description
if	array: see schema	No	Read Only	The OCF Interface set supported by this Resource
vods	array: see schema	Yes	Read Only	Array of information per VOD created by the Bridge
id	multiple types: see schema	No	Read Write	
n	multiple types: see schema	No	Read Write	
rt	array: see schema	No	Read Only	Resource Type
econame	string	Yes	Read Only	Ecosystem Name of the Bridged Device which is exposed by this VOD
n	multiple types: see schema	Yes	Read Write	
di	multiple types: see schema	Yes	Read Write	

7.2.6 CRUDN behaviour

Table 4 defines the CRUDN operations that are supported on the "oic.r.vodlist" Resource Type.

Table 4 – The CRUDN operations of the Resource with type "rt" = "oic.r.vodlist"

Create	Read	Update	Delete	Notify
	get			observe

