INTERNATIONAL STANDARD

ISO/IEC 30118-3

First edition
2018-11

Information technology — Open Connectivity Foundation (OCF) Specification —

Part 3: Bridging specification

Technologies de l'information — Spécification de la Fondation pour la connectivité ouverte (Fondation OCF) —
Partie 3: Spécification de pontage

Reference number
ISO/IEC 30118-3:2018(E)

ISO copyright office
CP 401 • Ch. de Blandonnet 8
CH-1214 Vernier, Geneva
Phone: +41 22 749 01 11
Fax: +41 22 749 09 47
Email: copyright@iso.org
Website: www.iso.org

Published in Switzerland

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. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

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).

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).

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.

This document was prepared by the Open Connectivity Foundation (OCF) (as the OCF Bridging Specification, Version 1.0.0) 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*.

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

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.

CONTENTS

Figures

Tables

1 Scope

This document specifies a framework for translation between OCF devices and other ecosystems, and specifies the behaviour of a translator that exposes AllJoyn producer applications to OCF clients, and exposes OCF servers to AllJoyn consumer applications. Translation of specific AllJoyn interfaces to or from specific OCF resource types is left to other specifications. Translation of protocols other than AllJoyn is left to a future version of this specification. This document provides generic requirements that apply unless overridden by a more specific document.

2 Normative references

The following documents, in whole or in part, are normatively referenced in this document and are indispensable for its application. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

AllJoyn About Interface Specification, *About Feature Interface Definitions*, Version 14.12
https://allseenalliance.org/framework/documentation/learn/core/about-announcement/interface

AllJoyn Configuration Interface Specification, *Configuration Interface Definition*, Version 14.12
https://allseenalliance.org/framework/documentation/learn/core/configuration/interface

D-Bus Specification, *D-Bus Specification*
https://dbus.freedesktop.org/doc/dbus-specification.html

IEEE 754, *IEEE Standard for Floating-Point Arithmetic*, August 2008

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

IETF RFC 4648, *The Base16, Base32, and Base64 Data Encodings*, October 2006
https://www.rfc-editor.org/info/rfc4648

IETF RFC 6973, *Privacy Considerations for Internet Protocols*, July 2013
https://www.rfc-editor.org/info/rfc6973

IETF RFC 7049, *Concise Binary Object Representation (CBOR)*, October 2013
https://www.rfc-editor.org/info/rfc7049

IETF RFC 7159, *The JavaScript Object Notation (JSON) Data Interchange Format*, March 2014
https://www.rfc-editor.org/info/rfc7159

JSON Schema Core, *JSON Schema: core definitions and terminology*, January 2013
http://json-schema.org/latest/json-schema-core.html

JSON Schema Validation, *JSON Schema: interactive and non interactive validation*, January 2013
http://json-schema.org/latest/json-schema-validation.html

JSON Hyper-Schema, *JSON Hyper-Schema: A Vocabulary for Hypermedia Annotation of JSON*, October 2016
http://json-schema.org/latest/json-schema-hypermedia.html

OCF 1.0 Core Specification, *Open Connectivity Foundation Core Specification*, Version 1.0

OCF Security Specification, *Open Connectivity Foundation Security Specification*, Version 1.0

OCF ASA Mapping, *OCF Resource to ASA Interface Mapping*, v0.3 candidate, July 2016
https://workspace.openconnectivity.org/apps/org/workgroup/smarthome_tg/download.php/6287/OCF_Resource_to_ASA_Interface_Mapping_v.0.3_candidate.docx

OIC 1.1 Core Specification, *Open Interconnect Consortium Core Specification*, Version 1.1

RAML Specification, *Restful API modelling language,* Version 0.8.
https://github.com/raml-org/raml-spec/blob/master/versions/raml-08/raml-08.md

OCF Resource Type Definitions, *API Definition Language for OCF Resource Type Definitions,* Release OCF-v1.0.0
https://github.com/openconnectivityfoundation/bridging

3 Terms, definitions, symbols and abbreviations

3.1 Terms and definitions

3.1.1
OCF Bridge Device
An OCF Device that can represent devices that exist on the network but communicate using a Bridged Protocol rather than OCF protocols.

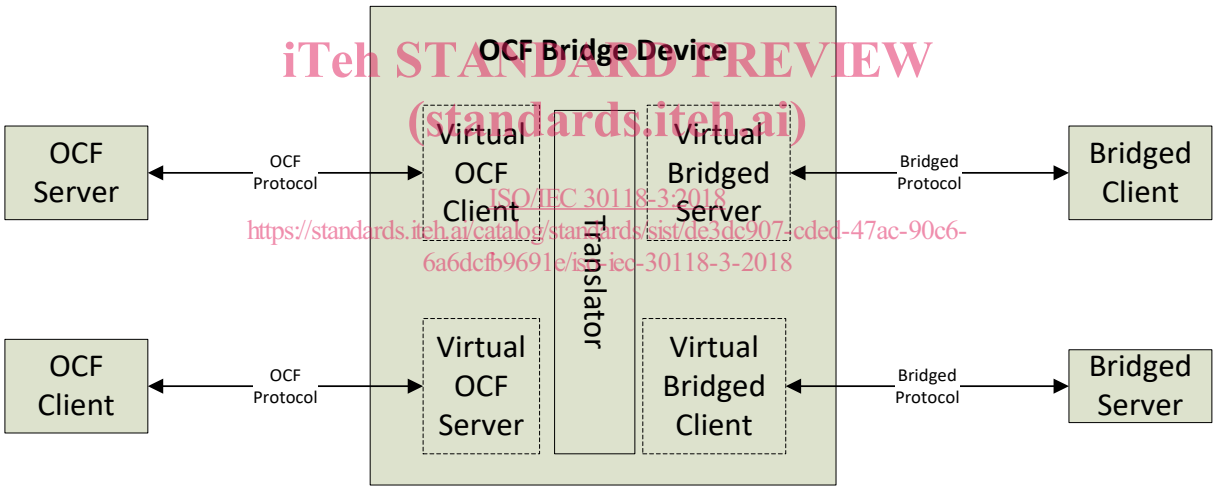

Figure 1. OCF Bridge Device Components

3.1.2
Bridged Protocol
another protocol (e.g., AllJoyn) that is being translated to or from OCF protocols

3.1.3
Translator
an OCF Bridge Device component that is responsible for translating to or from a specific Bridged Protocol. More than one translator can exist on the same OCF Bridge Device, for different Bridged Protocols.

3.1.4
OCF Client
a logical entity that accesses an OCF Resource on an OCF Server, which might be a Virtual OCF Server exposed by the OCF Bridge Device.

3.1.5
Bridged Client
a logical entity that accesses data via a Bridged Protocol. For example, an AllJoyn Consumer application is a Bridged Client.

3.1.6
Virtual OCF Client
a logical representation of a Bridged Client, which an OCF Bridge Device exposes to OCF Servers.

3.1.7
Virtual Bridged Client
a logical representation of an OCF Client, which an OCF Bridge Device exposes to Bridged Servers.

3.1.8
OCF Device
a logical entity that assumes one or more OCF roles (OCF Client, OCF Server). More than one OCF Device can exist on the same physical platform.

3.1.9
Virtual OCF Server
a logical representation of a Bridged Server, which an OCF Bridge Device exposes to OCF Clients.

3.1.10
Bridged Server
a logical entity that provides data via a Bridged Protocol. For example, an AllJoyn Producer is a Bridged Server. More than one Bridged Server can exist on the same physical platform.

3.1.11
Virtual Bridged Server
a logical representation of an OCF Server, which an OCF Bridge Device exposes to Bridged Clients.

3.1.12
OCF Resource
represents an artifact modelled and exposed by the OCF Framework

3.1.13
Virtual OCF Resource
a logical representation of a Bridged Resource, which an OCF Bridge Device exposes to OCF Clients.

3.1.14
Bridged Resource
represents an artifact modelled and exposed by a Bridged Protocol. For example, an AllJoyn object is a Bridged Resource.

3.1.15
OCF Resource Property
a significant aspect or notion including metadata that is exposed through the OCF Resource

3.1.16
OCF Resource Type
an OCF Resource Property that represents the data type definition for the OCF Resource

3.1.17
Bridged Resource Type

a schema used with a Bridged Protocol. For example, AllJoyn Interfaces are Bridged Resource Types.

3.1.18
OCF Server
a logical entity with the role of providing resource state information and allowing remote control of its resources.

3.1.19
Onboarding Tool
defined by the OCF Security Specification as: A logical entity within a specific IoT network that establishes ownership for a specific device and helps bring the device into operational state within that network.

3.1.20
Bridged Device
a Bridged Client or Bridged Server.

3.1.21
Virtual OCF Device
a Virtual OCF Client or Virtual OCF Server.

3.2 Symbols and abbreviations

3.2.1
CRUDN
Create Read Update Delete Notify
indicating which operations are possible on the resource

3.2.2
CSV
Comma Separated Value List
construction to have more fields in 1 string separated by commas. If a value contains a comma, then the comma can be escaped by adding "\" in front of the comma.

3.2.3
OCF
Open Connectivity Foundation
organization that created these specifications

3.2.4
RAML
RESTful API Modeling Language
Simple and succinct way of describing practically RESTful APIs (see the OIC 1.1 Core Specification, *Open Interconnect Consortium Core Specification*, Version 1.1

RAML Specification)

3.3 Conventions

In this specification several 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.

4 Document conventions and organization

For the purposes of this document, the terms and definitions given in the OCF 1.0 Core Specification apply.

4.1 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 specification. 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 specification and should be implemented. Recommended features take advantage of the capabilities of this specification, 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, 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 specification, they should not be implemented except for backward compatibility. The occurrence of a deprecated feature during operation of an implementation compliant with the current specification 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 specification.

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

Words that are emphasized are printed in *italic*.

4.2 Data types

Data types are defined in the OCF 1.0 Core Specification.

4.3 Document structure

Section 5 discusses operational scenarios. Section 6 covers generic requirements for any OCF Bridge, and section 7 covers the specific requirements for a Bridge that translates to/from AllJoyn. These are covered separately to ease the task of defining translation to other protocols in the future.

5 Operational Scenarios

The overall goals are to:

1. make Bridged Servers appear to OCF clients as if they were native OCF servers, and

2. make OCF servers appear to Bridged Clients as if they were native non-OCF servers.

5.1 "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. Translators 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 translator 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 translator. 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 translator. 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*.

For AllJoyn, deep translation is specified in OCF ASA Mapping, and on-the-fly translation is covered in section 7.2 of this document.

5.2 Use of introspection

Whenever possible, the translation code should make use of metadata available that indicates what the sender and recipient of the message in question are expecting. For example, devices that are AllJoyn Certified are required to carry the introspection data for each object and interface they expose. The OIC 1.1 Core Specification makes no such requirement, but the OCF 1.0 Core Specification does. When the metadata is available, translators should convert the incoming payload to exactly the format expected by the recipient and should use information when translating replies to form a more useful message.

For example, for an AllJoyn translator, the expected interaction list is presented on the list below:

Message Type	Sender	Receiver	Metadata
Request	AllJoyn 16.10	OIC 1.1	Not available
Request	AllJoyn 16.10	OCF 1.0	Available
Request	OIC 1.1 or OCF 1.0	AllJoyn 16.10	Available
Response	AllJoyn 16.10	OIC 1.1 or OCF 1.0	Available
Response	OIC 1.1	AllJoyn 16.10	Not available
Response	OCF 1.0	AllJoyn 16.10	Available

5.3 Stability and loss of data

Round-tripping through the translation process specified in this document is not expected to reproduce the same original message. The process is, however, designed not to lose data or precision in messages, though it should be noted that both OCF and AllJoyn payload formats allow for future extensions not considered in this document.

However, a third round of translation should produce the same identical message as was previously produced, provided the same information is available. That is, in the above chain, payloads 2 and 4 as well as 3 and 5 should be identical.

6 OCF Bridge Device

This section describes the functionality of an OCF Bridge Device; such a device is illustrated in Figure 2.

An OCF Bridge Device is a device that represents one or more Bridged Devices as Virtual OCF Devices on the network and/or represents one or more OCF Devices as Virtual 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 Virtual Bridged Device is how the device is encapsulated in an OCF Bridge Device.

An OCF Bridge Device shall be indicated on the OCF network 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 2.

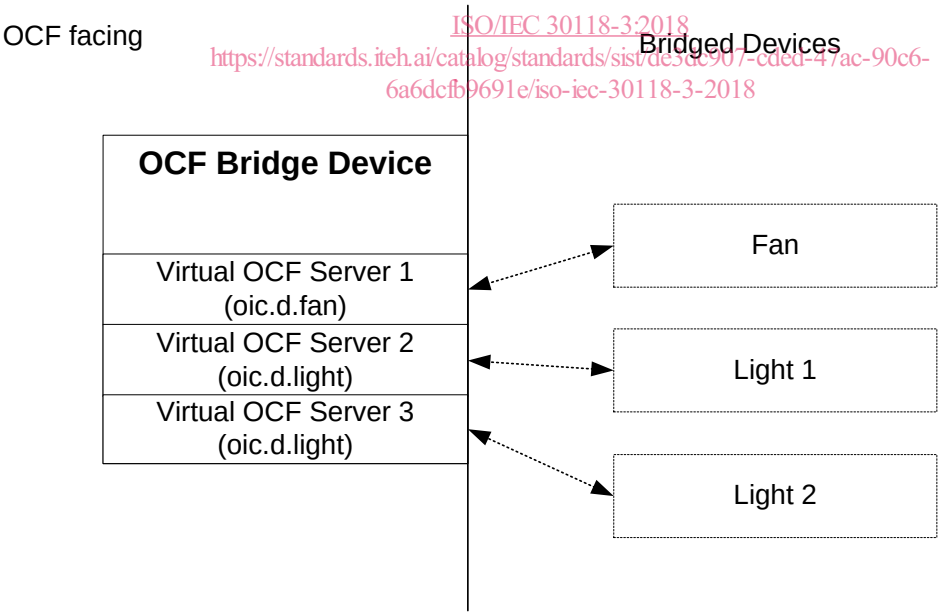

Figure 2: Schematic overview of an OCF Bridge Device bridging non-OCF devices

It is expected that the OCF Bridge Device creates a set of devices during the start-up of the OCF Bridge Device. The exposed set of Virtual OCF Devices can change as Bridged Devices are added or removed from the bridge. The adding and removing of Bridged Devices is implementation