



BSI Standards Publication

Information technology — Open Connectivity Foundation (OCF) Specification

Part 9: Core optional specification

National foreword

This British Standard is the UK implementation of ISO/IEC 30118-9:2021.

The UK participation in its preparation was entrusted to Technical Committee ICT/1, Information systems co-ordination.

A list of organizations represented on this committee can be obtained on request to its committee manager.

Contractual and legal considerations

This publication has been prepared in good faith, however no representation, warranty, assurance or undertaking (express or implied) is or will be made, and no responsibility or liability is or will be accepted by BSI in relation to the adequacy, accuracy, completeness or reasonableness of this publication. All and any such responsibility and liability is expressly disclaimed to the full extent permitted by the law.

This publication is provided as is, and is to be used at the recipient's own risk.

The recipient is advised to consider seeking professional guidance with respect to its use of this publication.

This publication is not intended to constitute a contract. Users are responsible for its correct application.

© The British Standards Institution 2022
Published by BSI Standards Limited 2022

ISBN 978 0 539 16191 5

ICS 35.200

Compliance with a British Standard cannot confer immunity from legal obligations.

This British Standard was published under the authority of the Standards Policy and Strategy Committee on 28 February 2022.

Amendments/corrigenda issued since publication

Date	Text affected
------	---------------

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

**Part 9:
Core optional specification**

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

Partie 9: Spécification facultative du cœur



Reference number
ISO/IEC 30118-9:2021(E)

AVA Documents
Document Sharing Hub
© ISO/IEC 2021



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.....	2
3.1 Terms and definitions	2
4 Document conventions and organization.....	3
4.1 Conventions	3
4.2 Notation	3
4.3 Data types	4
5 Functional interactions.....	4
5.1 Introduction	4
5.2 Onboarding, provisioning and configuration	4
5.3 Device management	6
5.4 Scenes	18
5.5 Rules	22
5.6 Icons	30
5.7 Alerts	31
Annex A (normative) Resource Type definitions.....	33
A.1 List of Resource Type definitions.....	33
A.2 Device Configuration	33
A.3 Platform Configuration.....	38
A.4 Icon.....	42
A.5 Maintenance	45
A.6 Network Monitoring.....	48
A.7 Scene List.....	52
A.8 Scene Collection	56
A.9 Scene Member	62
A.10 Alert.....	66
A.11 Alert Collection	69
A.12 software update.....	75
A.13 OCF Rule	79
A.14 OCF Rule Input Collection	84
A.15 OCF Rule Expression	88
A.16 OCF Rule Action Collection	91
A.17 OCF Rule Action.....	96

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 Core Optional 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*.

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 9: Core optional specification

1 Scope

The OCF Core specifications are divided into a series of documents:

- Core specification: The Core specification document specifies the Framework, i.e., the OCF core architecture, interfaces, protocols and services to enable OCF profiles implementation for Internet of Things (IoT) usages and ecosystems. This document is mandatory for all Devices to implement.
- Core optional specification (this document): The Core optional specification document specifies the Framework, i.e., the OCF core architecture, interfaces, protocols and services to enable OCF profiles implementation for Internet of Things (IoT) usages and ecosystems that can optionally be implemented by any Device.
- Core extension specification(s): The Core extension specification(s) document(s) specifies optional OCF Core functionality that are significant in scope (e.g., Wi-Fi easy setup, Cloud).

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 DIS 20924, *Information Technology – Internet of Things – Vocabulary*, June 2018
<https://www.iso.org/standard/69470.html>

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>

IETF RFC 3339, *Date and Time on the Internet: Timestamps*, July 2002
<https://www.rfc-editor.org/info/rfc3339>

IETF RFC 5234, *Augmented BNF for Syntax Specifications: ABNF*, January 2008
<https://www.rfc-editor.org/info/rfc5234>

IETF RFC 5424, *The Syslog Protocol*, March 2009
<https://tools.ietf.org/html/rfc5424>

IETF RFC 5646, *Tags for Identifying Languages*, September 2009
<https://www.rfc-editor.org/info/rfc5646>

IANA ifType-MIB Definitions
<https://www.iana.org/assignments/ianaiftype-mib/ianaiftype-mib>

IANA Media Types Assignment, March 2017

<http://www.iana.org/assignments/media-types/media-types.xhtml>

OpenAPI specification, *fka Swagger RESTful API Documentation 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, ISO/IEC 30118-2, 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

Alert

information provided by the Device by means of a specialised Resource Type that provides details with regard to potential problems, issues, or Device status of interest on which action can be taken

3.1.2

Rule

Resource that implements autonomous decision logic according to a condition-action pattern

3.1.3

Rule Action

Resource that is actuated with a defined value when the *Rule Result* (3.1.6) holds "true"

3.1.4

Rule Expression

definition of the *Rule* (3.1.1) logic in terms of the defined *Rule Inputs* (3.1.5), and which evaluates to a boolean *Rule Result* (3.1.6), for which "true" means that the *Rule* (3.1.1) has been triggered

3.1.5

Rule Input

Resources that contain the Properties whose values are evaluated as part of the *Rule Expression* (3.1.4)

3.1.6

Rule Result

Property which reflects the result of the evaluation of the *Rule Expression* (3.1.4)

3.1.7

Scene

static entity that stores a set of defined Property values for a Collection of Resources

Note 1 to entry: A *Scene* (3.1.3) is a prescribed setting of a set of Resources with each having a predetermined value for the Property that has to change.

3.1.8

Scene Collection

Collection that contains an enumeration of possible *Scene Values* (3.1.10) and the current *Scene Value* (3.1.10)

Note 1 to entry: The member values of the *Scene Collection* (3.1.8) are *Scene Members* (3.1.9).

3.1.9**Scene Member**

Resource that contains mappings of *Scene Values* (3.1.10) to values of a Property in the Resource

3.1.10**Scene Value**

Scene (3.1.3) enumerator representing the state in which a Resource can be

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)(M).

- These basic features shall be implemented to comply with Core Architecture. 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)(S).

- These features add functionality supported by Core Architecture and should be implemented. Recommended features take advantage of the capabilities Core Architecture, 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 (may or allowed)(O).

- These features are neither required nor recommended by Core Architecture, but if the feature is implemented, it shall meet the specified requirements to be in compliance with these guidelines.

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.

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.

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

Words that are emphasized are printed in *italic*.

In all of the Property and Resource definition tables that are included throughout this document the "Mandatory" column indicates that the item detailed is mandatory to implement; the mandating of inclusion of the item in a Resource Payload associated with a CRUDN action is dependent on the applicable schema for that action.

4.3 Data types

Resources are defined using data types derived from JSON values as defined in clause 4.3 in ISO/IEC 30118-1.

5 Functional interactions

5.1 Introduction

The functional interactions between a Client and a Server are described in 5.2 through 5.7 respectively. The functional interactions use CRUDN messages (clause 8 in ISO/IEC 30118-1) and include Discovery, Notification, and Device management. These functions require support of core defined Resources as defined in Table 1.

Table 1 – List of optional Core Resources

Pre-defined URI	Resource Name	Resource Type	Related Functional Interaction	Mandatory
(none)	Configuration	"oic.wk.con"	Device management	No
(none)	Configuration	"oic.wk.con.p"	Device management	No
"/oic/mnt"	Maintenance	"oic.wk.mnt"	Device management	No
(none)	Network monitoring	"oic.wk.nmon"	Device management	No
(none)	Software update	"oic.wk.softwareupdate"	Device management	No
(none)	Icon	"oic.r.icon"	Icons	No
(none)	Scene List	"oic.wk.scenelist"	Scenes	No
(none)	Scene Collection	"oic.wk.scenecollection"	Scenes	No
(none)	Scene Member	"oic.wk.scenemember"	Scenes	No
(none)	Alerts	"oic.r.alert"	Alerts	No
(none)	Alerts Collection	"oic.r.alertcollection"	Alerts	No

5.2 Onboarding, provisioning and configuration

Onboarding and provisioning are fully defined by ISO/IEC 30118-2.

Should a Device support Client update of configurable information it shall do so via exposing an "oic.wk.con" Core Resource (Table 2) in "/oic/res".

Table 2 – Configuration Resource

Example URI	Resource Type Title	Resource Type ID ("rt" value)	OCF Interfaces	Description	Related Functional Interaction
"/example/oic/con"	Device configuration	"oic.wk.con"	"oic.if.rw"	The Resource Type through which configurable information specific to the Device is exposed.	Configuration

				The Resource Properties exposed in "oic.wk.con" are listed in Table 3.	
"/example/oic/con"	Platform configuration	"oic.wk.con.p"	"oic.if.rw"	The optional Resource Type through which configurable information specific to the Platform is exposed. The Properties exposed in "oic.wk.con.p" are listed in Table 4.	Configuration

Table 3 defines the "oic.wk.con" Resource Type. Complete details are provided in annex A.2.

Table 3 – "oic.wk.con" Resource Type definition

Property title	Property name	Value type	Value rule	Unit	Access mode	Mandatory	Description
(Device) Name	"n" (Common Property of "/example/oic/con")	"string"	N/A	N/A	R, W	Yes	Human friendly name configurable by the end user (e.g. Bob's thermostat). The "n" Common Property of the oic.wk.con Core Resource and the "n" Common Property of the "/oic/d" Core Resource shall have the same Value. When the "n" Common Property Value of the oic.wk.con Core Resource is modified, it shall be reflected to the "n" Common Property of "/oic/d" Core Resource.
Location	"loc"	array of float (has two elements, the first is latitude, the second is longitude)	N/A	Degrees	R, W	No	Provides location information where available.
Location Name	"locn"	"string"	N/A	N/A	R, W	no	Human friendly name for location For example, "Living Room".
Currency	"c"	"string"	N/A	N/A	R,W	no	Indicates the currency that is used for any monetary transactions
Region	"r"	"string"	N/A	N/A	R,W	no	Free form text Indicating the current region in which the Device is located geographically.
Localized Names	"ln"	"array"	N/A	N/A	R,W	no	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 IETF 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.
Default Language	"dl"	"language-tag"	N/A	N/A	R,W	no	The default language supported by the Device, specified as an IETF RFC 5646 language tag. By default, clients can treat any string Property as being in this language unless the Property specifies otherwise.

Table 4 defines the "oic.wk.con.p" Resource Type. Complete details are provided in annex A.3.

Table 4 – "oic.wk.con.p" Resource Type definition

Property title	Property name	Value type	Value rule	Unit	Access mode	Mandatory	Description
Platform Names	"mnpn"	"array"	N/A	N/A	R,W	No	Friendly name of the Platform. This Property is an array of objects where each object has a "language" field (containing an IETF RFC 5646 language tag) and a "value" field containing the platform friendly name in the indicated language. For example, [{"language": "en", "value": "Dave's Laptop"}]

5.3 Device management

5.3.1 Overview

Device management includes the following functions:

- Diagnostics and maintenance
- Network monitoring

5.3.2 Diagnostics and maintenance Resource Type

The Diagnostics and Maintenance Resource Type is intended to enable the resolution of issues encountered with the Devices while operating in the field. If diagnostics and maintenance is supported by a Device, the Core Resource "/oic/mnt" shall be supported as described in Table 5.

Table 5 – Optional diagnostics and maintenance Device management Core Resources

Pre-defined URI	Resource Type Title	Resource Type ID ("rt" value)	OCF Interfaces	Description	Related Functional Interaction
"/oic/mnt"	Maintenance	"oic.wk.mnt"	"oic.if.rw"	The Resource through which the Device is maintained and can be used for diagnostic purposes. The Properties exposed by "/oic/mnt" are listed in Table 6.	Device management

Table 6 defines the "oic.wk.mnt" Resource Type. At least one of the Factory_Reset, Reboot, or last error Properties shall be implemented. Complete details are provided in annex A.5.

Table 6 – "oic.wk.mnt" Resource Type definition

Property title	Property name	Value type	Value rule	Unit	Access mode	Mandatory	Description
Factory_Reset	"fr"	"boolean"	N/A	N/A	R, W	No	When writing to this Property: false – No action (Default*) true – Start Factory Reset

							When reading this Property, a value of true indicates a pending factory reset. Once the factory reset has been completed, the Device shall set the value back to false. This Property is functionally equivalent to a transition to a state of Hard Reset as defined in ISO/IEC 30118-2, clause 8.1
Reboot	"rb"	"boolean"	N/A	N/A	R, W	No	When writing to this Property: false – No action (Default) true – Start Reboot After Reboot, this value shall be changed back to the default value (i.e., false)
Last error	"err"	"integer"	HTTP error code	N/A	R	No	Last occurred error code, shall be cleared to 503 (service unavailable), when doing a Factory Reset or Reboot. All HTTP errors outside the 100, 200 or 300 range shall be stored.

NOTE Default indicates the value of this Property as soon as the Device is rebooted or factory reset.

5.3.3 Core behaviours on Device maintenance state changes

5.3.3.1 Overview

As defined in ISO/IEC 30118-2 a Device has a state machine through which it transitions during its operational lifetime.

ISO/IEC 30118-2 details actions on such state transitions for the Resources defined therein. This clause defines the actions to be taken on such state transitions for the Resources and functionality defined within this document.

The state transitions to be considered are:

- RFNOP to Soft Reset
- RFNOP to Hard Reset
- RFNOP to RFPRO
- RFPRO to RFNOP

Table 7 provides a summary of the actions to be taken in each case for functions defined in the ISO/IEC 30118-2 and this document, other extensions to these documents may define further behaviours.

Table 7 – Actions on Device state change

	Soft reset	Hard reset	RFNOP -> RFPRO	RFPRO -> RFNOP
SVR	As per ISO/IEC 30118-2 clause 8.5	As per ISO/IEC 30118-2 clause 8.1	As per ISO/IEC 30118-2 clause 8.3	As per ISO/IEC 30118-2 clause 8.4
Mandatory Core Resources	No change	Reset to defined defaults, see clause 5.3.3.3	No change	No change

Optional Core Resources	No change	Reset to defined defaults, see clause 5.3.3.3.4	No change	No change
Vertical Resources	No change	Reset to defined defaults; see clause 5.3.3.3	No change	No change
Created Resources	No change	Deleted	No change	No change
Observe Transactions	No change	Canceled; see clause 5.3.3.2	No change	Re-evaluate ACL; see clause 5.3.3.2
OCF Cloud	No change	See clause Error! Reference source not found.	No change	No change

5.3.3.2 Handling of Observe transactions

On a transition to hard reset all active Observe transactions shall be cancelled by the Server by sending a "Service Unavailable" response on each active Observe transaction.

On a state transition that allows for modification of the access controls that exist against a Resource (such as from RFPRO to RFNOP) it is possible that the access controls themselves as defined within the ISO/IEC 30118-2 are changed such that the original RETRIEVE operation that established the Observe would not have been allowed. In such instances the Server shall cancel the Observe by sending a "Service Unavailable" response on the Observe transaction.

5.3.3.3 Reset of Resource Properties to defined defaults

5.3.3.3.1 Overview

On a hard or factory reset Resource Properties are reset to default values. These are commonly referred to as *manufacturer defaults* however it is not possible in all instances to revert to such values as they may not be known or be practicable.

The default values to be applied for the mandatory and optional Core Resources, plus any Vertical Resources are defined in clauses 5.3.3.3.2 through 5.3.3.3.4 respectively.

5.3.3.3.2 Defaults for Vertical Resources

Default values for any Vertical Resources exposed by a Device are up to the implementation.

5.3.3.3.3 Defaults for mandatory Core Resources

Table 8 and Table 9 capture default values that shall be set for mandatory Properties of the mandatory Core Resources where those Resources contain Properties that can be changed by a Client. This excludes "/oic/res" as that has no mutable Properties.

Table 8 – Default values for "/oic/d"

Property	Default	Notes
"n"	""	Empty string if "/oic/con" is also exposed, otherwise not mutable.
"di"	See ISO/IEC 30118-2 requirements	.
"icv"	Unchanged	Not mutable
"dmv"	Unchanged	Not mutable
"piid"	See ISO/IEC 30118-2 requirements.	

Table 9 – Default values for "/oic/p"

Property	Default	Notes
"pi"	See ISO/IEC 30118-2 requirements.	
"mnmn"	Unchanged	Not mutable

5.3.3.3.4 Defaults for optional Core Resources

This clause details the actions to be taken with respect to the optional Core Resources.

The icon Resource ("oic.r.icon") has no mutable Properties, so no action is to be taken.

The network monitoring Resource ("oic.wk.nmon") shall have all Properties reset (i.e. behaviour as if an UPDATE with the "reset" Property set to "true" had been received). The "col" Property shall be set to "false".

Any Resources that were added to an instance of a Collection or a specialisation of a Collection (e.g. Scene List ("oic.wk.scenelist") or Scene Collection ("oic.wk.scenecollection")) by a Client shall be deleted.

The Device configuration Resource ("oic.wk.con") shall be modified in accordance with Table 10 for those Properties that are implemented.

Table 10 – Default values for Device configuration Resource

Property	Default	Notes
"loc"	[0.0,0.0]	
"locn"	""	Empty string
"c"	""	Empty string
"r"	""	Empty string
"ln"	[{}]	One item array with an empty object.
"dl"	Defined by the manufacturer	Recommend the primary language tag for the region into which the Device is marketed (e.g. "en" for primarily English speaking countries).

The platform configuration Resource ("oic.wk.con.p") shall be modified in accordance with Table 11

Table 11 – Default values for Platform configuration Resource

Property	Default	Notes
"mnpn"	[{}]	One item array with an empty object

5.3.4 Network monitoring Resource Type

Network monitoring is used for monitoring the current network state of the Device.

The network monitoring Resource Type is "oic.wk.nmon" and is described in Table 12. The Resource Type may occur multiple times if more than 1 network interface is implemented. The Common Property "n" may be used to distinguish the different network interfaces, like distinguish the 2.4 and 5G Wi-Fi network interfaces.

Table 12 – Optional monitoring Device management Core Resources

Example URI	Resource Type Title	Resource Type ID ("rt" value)	OCF Interfaces	Description	Related Functional Interaction
/example/oc/nmon	Network Monitoring	"oic.wk.nmon"	"oic.if.rw", "oic.if.baseline"	The Resource through which the Device is monitored. The Resource exposes Properties relevant to aspects that may be monitored. The Resource Properties exposed by Resource Type "oic.wk.nmon" are listed in Table 13	Device management

Table 13 defines "oic.wk.nmon" Resource Type. Complete details are provided in annex A.6.

Table 13 – "oic.wk.nmon" Resource Type definition

Property title	Property name	Value type	Value rule	Unit	Access mode	Mandatory	Description
Network indicator	"ianaifType"	"integer"	The integer value of the ianaifType	N/A	R	Yes	The network type this Resource is collecting information from as defined by IANA ifType-MIB Definitions.
reset	"reset"	"boolean"	True, all collected values should be reset. The server should reset the value automatically to false after the reset occurred.	N/A	RW	Yes	Reset of the collected values
Collecting status indication	"col"	"boolean"	True: collecting data. False: not collecting data	N/A	RW	Yes	Boolean to start/stop collecting data.
Transmission bytes	"tx"	"integer"	N/A	kilo bytes	R	No	Amount of transmitted kilo bytes from the collection
Reception bytes	"rx"	"integer"	N/A	kilo bytes	R	No	Amount of received kilo bytes from the collection.
Maximum message size tx	"mmstx"	"integer"	bytes	bytes	R	No	Maximum transmitted message, e.g. Max(tx) in the collection period
Maximum message size rx	"mmsrx"	"integer"	bytes	bytes	R	No	Maximum received message, e.g. Max(rx) in the collection period
Average message size -tx	"amstx"	"integer"	bytes	bytes	R	No	Average transmitted message size, e.g. AVG(tx) in the collection period.

Average message size -rx	"amsrx"	"integer"	bytes	byte s	R	No	Average received message size e.g. AVT(rx) in the collection period.
--------------------------	---------	-----------	-------	-----------	---	----	---

Examples of typical used values for ianaifType are 71 (ieee80211) for Wi-Fi and 6 (ethernetCsmacd) for Ethernet.

A Device should start collecting network monitoring data when receiving an UPDATE operation with the parameter "col" = true. A Device should stop collecting network data when receiving an UPDATE operation with parameter "col" = false. The collected network data should be reset when an UPDATE operation with parameter "reset" = true is received, if the parameter "reset" is false then the values should not be reset. Figure 1 illustrates the interactions with the network monitoring Resource.

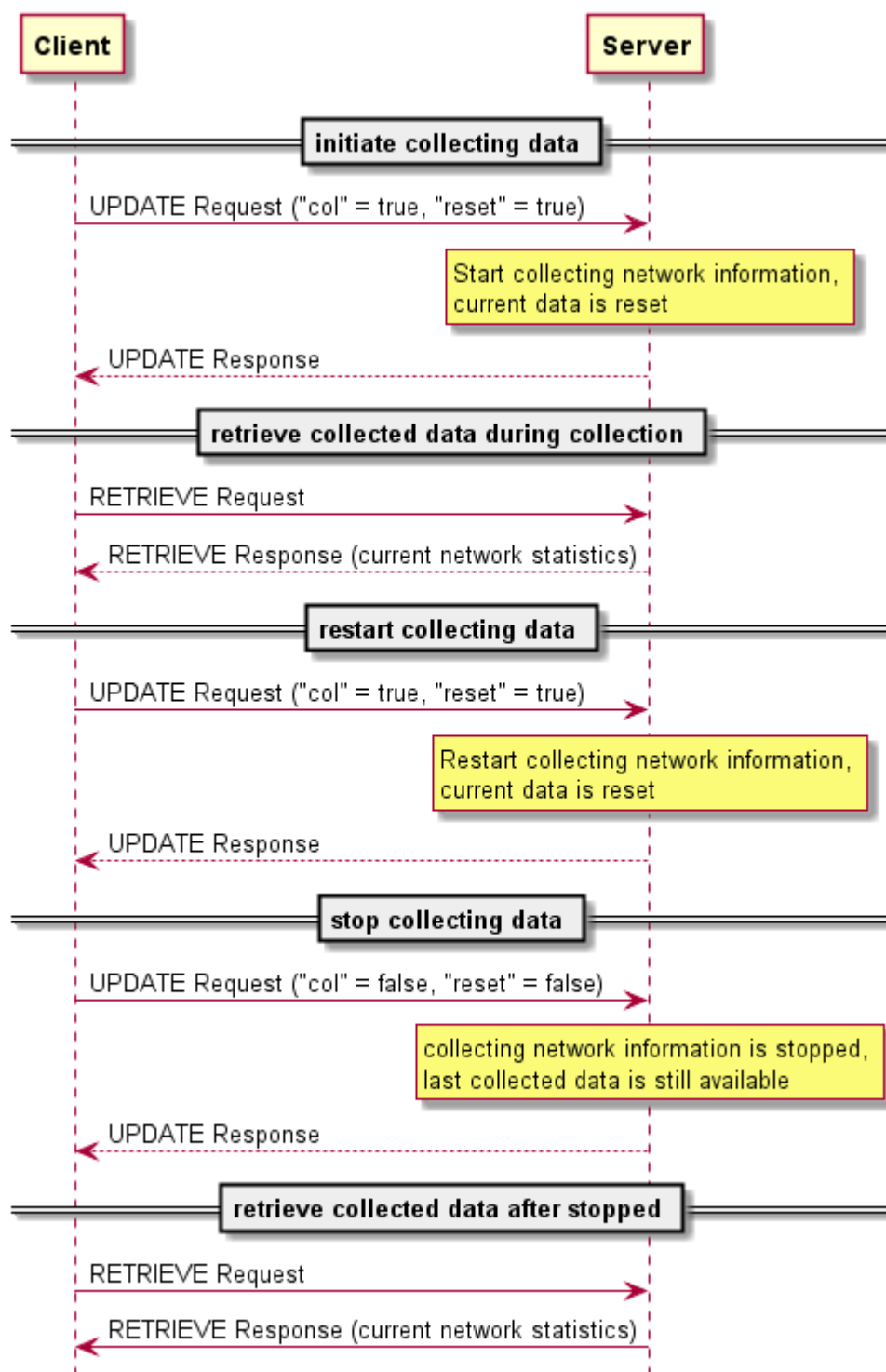


Figure 1 – Interactions with the network monitoring Resource

The state transition diagram for collecting or not collecting network information is described by Figure 2.

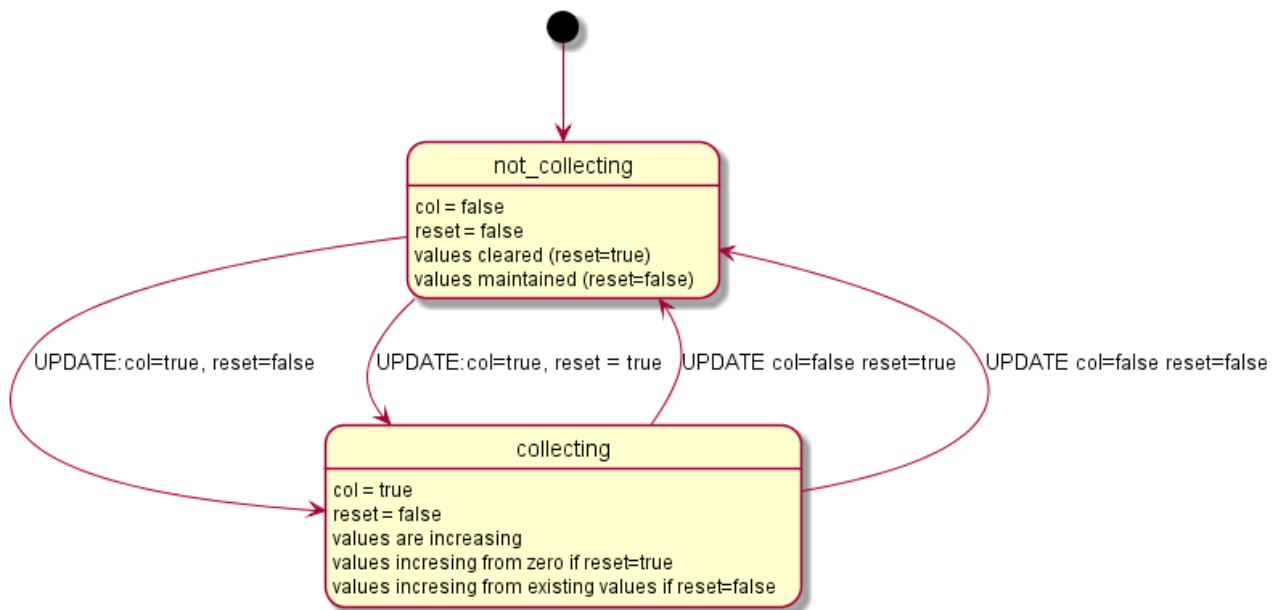


Figure 2 – State transition diagram for network information collection

5.3.5 Software update Resource Type

The software update Resource is used to control software updates of the Device.

In ISO/IEC 30118-2 there is already a manual triggered software update mechanism available. The triggering of the Client (manual) software update is achieved via the security Resource Type "oic.r.pstat" by using the appropriate bits in the "tm" Property. The software update triggering results in updates of the "cm" Property in the "oic.r.pstat" Resource Type (see ISO/IEC 30118-2 clause 13.8). The software update Resource adds additional features to the security specified mechanism, like:

- Specify the source to obtain the software package.
- Time scheduled software update actions.
- Status information, especially more info about various error situations.

If the Device implements the software update Resource, it is required to implement the software update behaviour to actually update the software of the Device as indicated by the "oic.r.pstat" "cm" bits as defined in ISO/IEC 30118-2 clause 13.8. Also the security defined software update process shall use the data that is set on the software update Resource like the "purl" Property.

The software update Resource Type is "oic.r.softwareupdate" and is described in Table 14.

Table 14 – Optional software update Resources

Example URI	Resource Type Title	Resource Type ID ("rt" value)	OCF Interfaces	Description	Related Functional Interaction
"/example/oic/swupd"	Software Update	"oic.r.softwareupdate"	"oic.if.rw", "oic.if.baseline"	The Resource exposes Properties to control and monitor the software update mechanism. The Properties exposed by Resource Type "oic.r.softwareupdate" are listed in Table 15.	Device management

Table 15 defines the Properties of the "oic.r.softwareupdate" Resource Type. Complete details are provided in annex A.12.

Table 15 – "oic.r.softwareupdate" Resource Type definition

Property title	Property name	Value type	Value rule	Unit	Access mode	Mandatory	Description
New version	"nv"	"string"	N/A	N/A	R	No	New available software version.
Package url	"purl"	"string"	URL	N/A	RW	Yes	Source of the software package, might be an HTTPS or a CoAPs URL.
Action	"swupdateaction"	"string"	enum (see Table 17)	N/A	RW	Yes	Scheduled action to do a software update.
State	"swupdatestate"	"string"	enum (see Table 16)	N/A	R	Yes	State of the software update.
Result	"swupdateresult"	"integer"	N/A	N/A	R	Yes	Result of the software update. List of error codes are as defined in Table 18.
Lastupdate	"lastupdate"	"string"	date-time	N/A	R	No	Time of the last software update according to IETF RFC 3339. Initial set on date of manufacturing.
Signage	"signed"	"string"	enum	N/A	R	No	Signage method of the software package, currently the only allowed value is "vendor".
Updatetime	"updatetime"	"string"	date-time	N/A	RW	Yes	Scheduled time, according to IETF RFC 3339, to do action which is specified in the "swupdateaction" Property.

The values of the "swupdatestate" Property are described in Table 16.

Table 16 – State definitions and state transitions of the software update Resource

Description	Value of Property "swupdatestate"	equivalent "cm" bit values in "pstat".	Transition allowed from state
Idle, waiting for updates	"idle"	Bit 64 = 0 Bit 128 = 0 Bit 256 = 0	"nsa", "svv", "sva", "upgrading"
New software available (after checking for new software being available on the url indicated by "purl"). This step does not download the new software	"nsa"	Bit 64 = 0 Bit 128 = 0 Bit 256 = 1	"idle", "svv", "sva", "upgrading"
Software version validation (during downloading and checking the software integrity)	"svv"	Bit 64 = 0 Bit 128 = 0 Bit 256 = 1	"idle", "nsa", "sva", "upgrading"
Software version available (The software is downloaded and deemed to be valid)	"sva"	Bit 64 = 1 Bit 128 = 0 Bit 256 = 1	"idle", "nsa", "svv", "upgrading"
Upgrading	"upgrading"	Bit 64 = 1 Bit 128 = 1 Bit 256 = 1	"idle", "nsa", "svv", "sva"

The typical state transitions are described Figure 3. The state transitions can be triggered manually or by a timed action. The manual state triggers (i.e., "tm" Property of "oic.r.pstat") are described in ISO/IEC 30118-2 clause 13.8. The timed state triggers are managed using the "swupdateaction" and "updatetime" Properties of the software update Resource to trigger software update actions at some future date and time. The action names for scheduled actions are listed in Table 17. When the "updatetime" for the timed action is in the past then the update shall not take place, it is implementation dependent if the UPDATE with an "updatetime" value in the past will give an error on the UPDATE operation.

Table 17 – Value definitions for the Property "swupdateaction"

Description	Value of Property "swupdateaction", for scheduled update actions.	Action taken	Equivalent "pstat" "tm" bits.
Nothing scheduled (not applicable).	"idle"	No action	
Initiate software availability check.	"isac"	Check on remote end point if a newer software version is available.	"tm" bit 256.
Initiate software version validation.	"isvv"	Downloads and verifies if the software version is valid.	"tm" bit 64.
Initiate secure software update.	"upgrade"	Upgrades the software in the Device. It uses the downloaded and validated software package. If no validated software package is available on the Device, the Device takes the necessary steps to obtain a validated software package, by downloading and verifying the software from the external source.	"tm" bit 128.

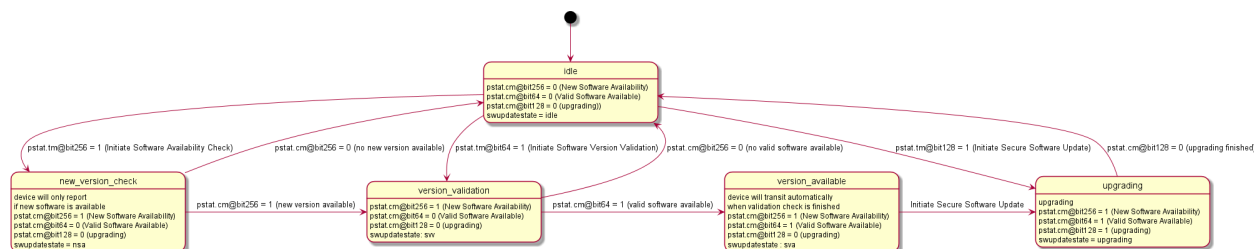


Figure 3 – Typical state transition diagram for software update

The "purl" Property indicates the URL to obtain the software package from. This URL shall be a fully qualified URL. If the value is an empty string ("") then the Device will use the built in vendor defined URL (see ISO/IEC 30118-2). If a built in URL is not implemented, setting the "purl" Property value to an empty string will result in an error code value of 6 as defined in Table 18.

Table 18 – List of codes of the "swupdateresult" Property.

Description	code
Idle.	0
Success, everything went well.	1
Not enough RAM.	2
Not enough Flash.	3

Connection lost.	4
Software validation failure.	5
Invalid URL to receive the software package.	6
Unsupported protocol for download URL.	7
Firmware update failed.	8
Software transport error codes. HTTP result codes when accessing the URL to download the software package.	400-600

Figure 4 depicts a typical update scenario. This scenario is using the observability of "pstat", so that the Client is informed on the changes of the "cm" bit value to track the progress.

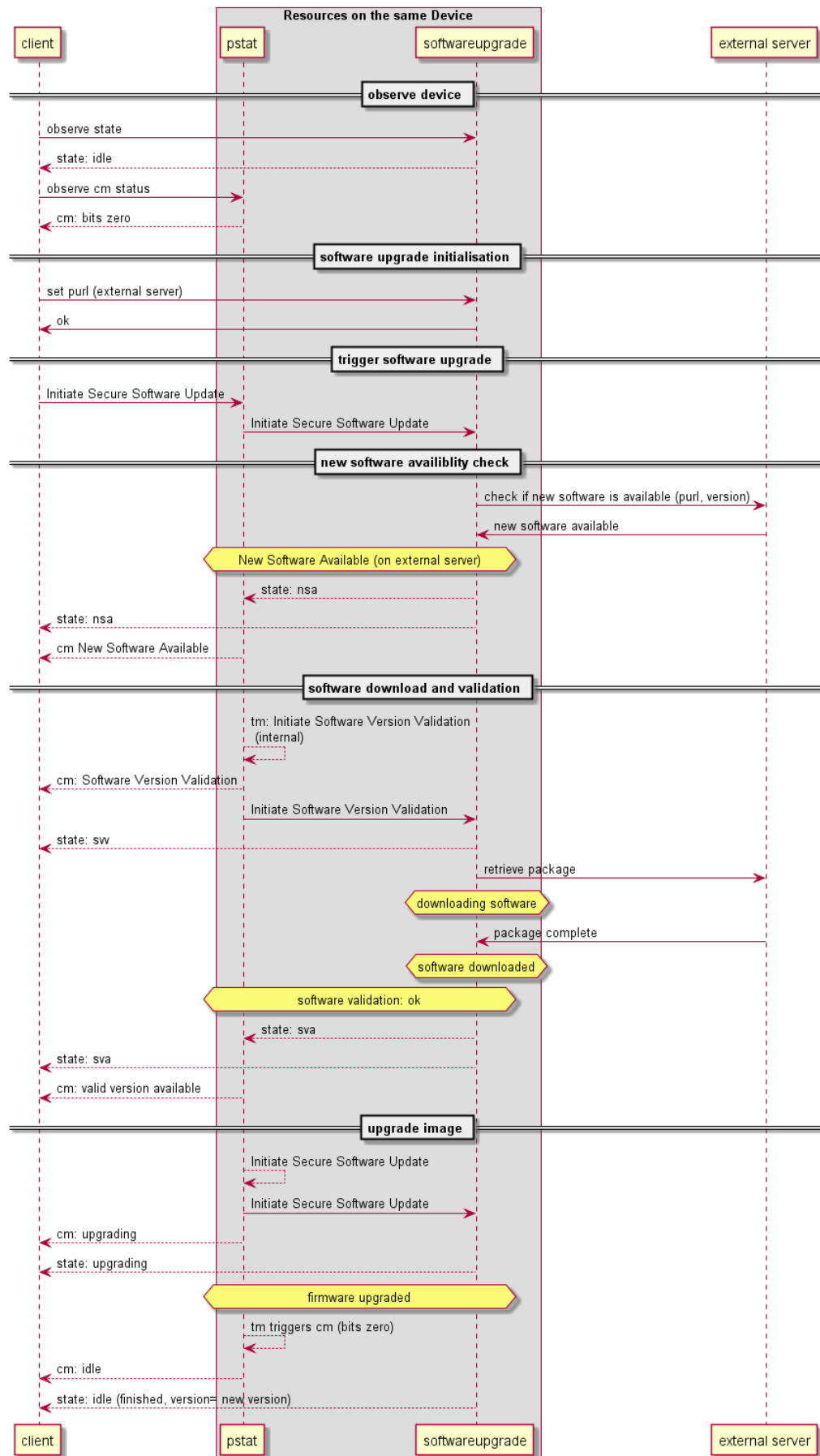


Figure 4 – Typical sequence for non-scheduled upgrading software

5.4 Scenes

5.4.1 Introduction

Scenes are a mechanism for automating certain operations.

A Scene is a static entity that stores a set of defined Property values for a Collection of Resources. Scenes provide a mechanism to store a setting over multiple Resources that may be hosted by multiple separate Servers. Scenes, once set up, can be used by multiple Clients to recall a setup.

Scenes can be grouped and reused, a group of Scenes is also a Scene.

In short, Scenes are bundled user settings.

5.4.2 Scenes Resource model

5.4.2.1 Introduction

Scenes are described by means of Resources. The Scene Resources are hosted by a Server and the top level Resource is listed in "/oic/res". This means that a Client can determine if the Scene functionality is hosted on a Server via Resource discovery as defined in clause 11.2 in ISO/IEC 30118-1. The setup of Scenes is driven by Client interactions. This includes creating new Scenes, and mappings of Server Properties that are part of a Scene.

The Scene functionality is created by multiple Resources and has the structure depicted in Figure 5. The sceneList and sceneCollection Resources are overloaded Collection Resources. The sceneCollection Resource contains a list of Scenes. This list contains zero or more Scenes. The sceneMember Resource contains the mapping between a Scene and what needs to happen according to that Scene on an indicated Resource.



Figure 5 – Generic Scene Resource structure

5.4.2.2 Scene creation

A Client desiring to interact with Scenes needs to first determine if the Server supports the Scene feature; the sceneMembers of a Scene that are Resources of end Device being updated by the Scene change do not have to be co-located on the Server supporting the Scene feature. This can be done by checking if "/oic/res" contains the "rt" of the sceneList Resource. This is depicted in first steps of Figure 6. The sceneCollection Resource is created by the Server using some out of bound mechanism, Client creation of Scenes is not supported at this time. This will entail defining the Scene with an applicable list of Scene Values and the mappings for each Resource being part of the Scene. The mapping for each Resource being part of the sceneCollection Resource is described by a Resource called sceneMember. The sceneMember Resource contains the link to a Resource and the mapping between the Scene listed in the "sceneValues" Property and the actual Property value of the Resource indicated by the Link.

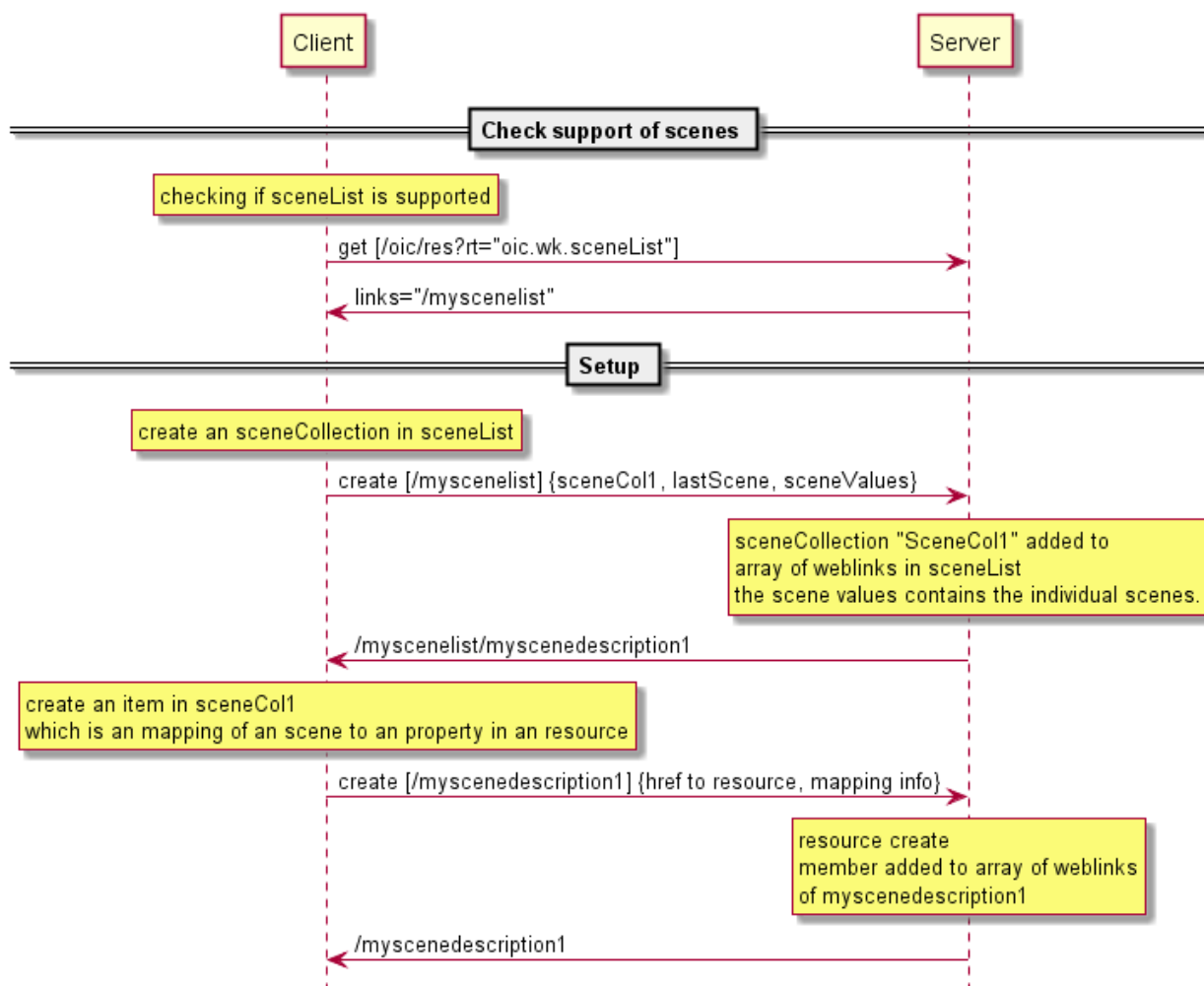


Figure 6 – Interactions to check Scene support and setup of specific Scenes

5.4.2.3 Interacting with Scenes

All capable Clients can interact with Scenes. The allowed Scene Values and the last applied Scene Value can be retrieved from the Server hosting the Scene. The Scene Value shall be changed by issuing an UPDATE operation with a payload that sets the "lastScene" Property to one of the listed allowed Scene Values. These steps are depicted in Figure 7. Note that the "lastScene" Property value does not imply that the current state of all Resources that are part of the Scene will be at the mapped value. This is due to that the setting the Scene Values are not modelled as actual states of the system. This means that another Client can change just one Resource being part of the Scene without having feedback that the state of the Scene is changed.

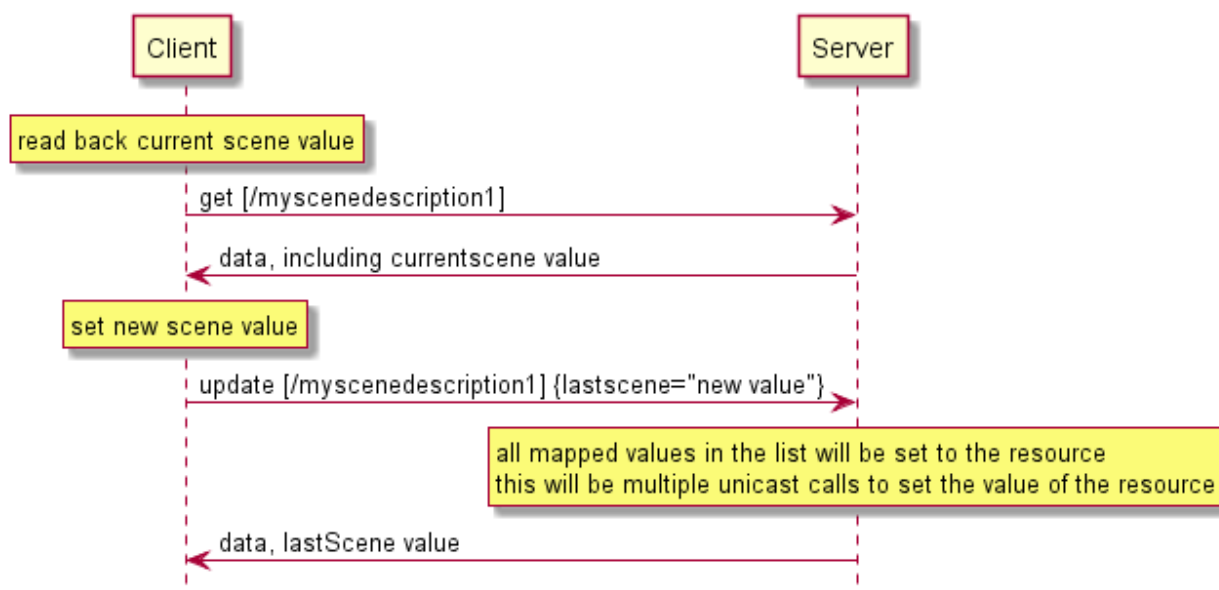


Figure 7 – Client interactions on a specific Scene

As described previously, a Scene can reference one or more Resources (i.e., sceneMembers) that are present on one or more Servers. The Scene Members are re-evaluated each time a Scene change takes place. This evaluation is triggered by a Client that is either embedded as part of the Server hosting the Scene, or separate to the Server having knowledge of the Scene via a RETRIEVE operation, Observing the referenced Resources using the mechanism described in clause 11.3.2 in ISO/IEC 30118-1. The embedded Client located in the same Device with the Server is a general Client but interacts only with Scene functionalities. During the evaluation the mappings for the new Scene Value will be applied to the Servers which contain sceneMembers from the Scene that is being updated. This behaviour is depicted in Figure 8.

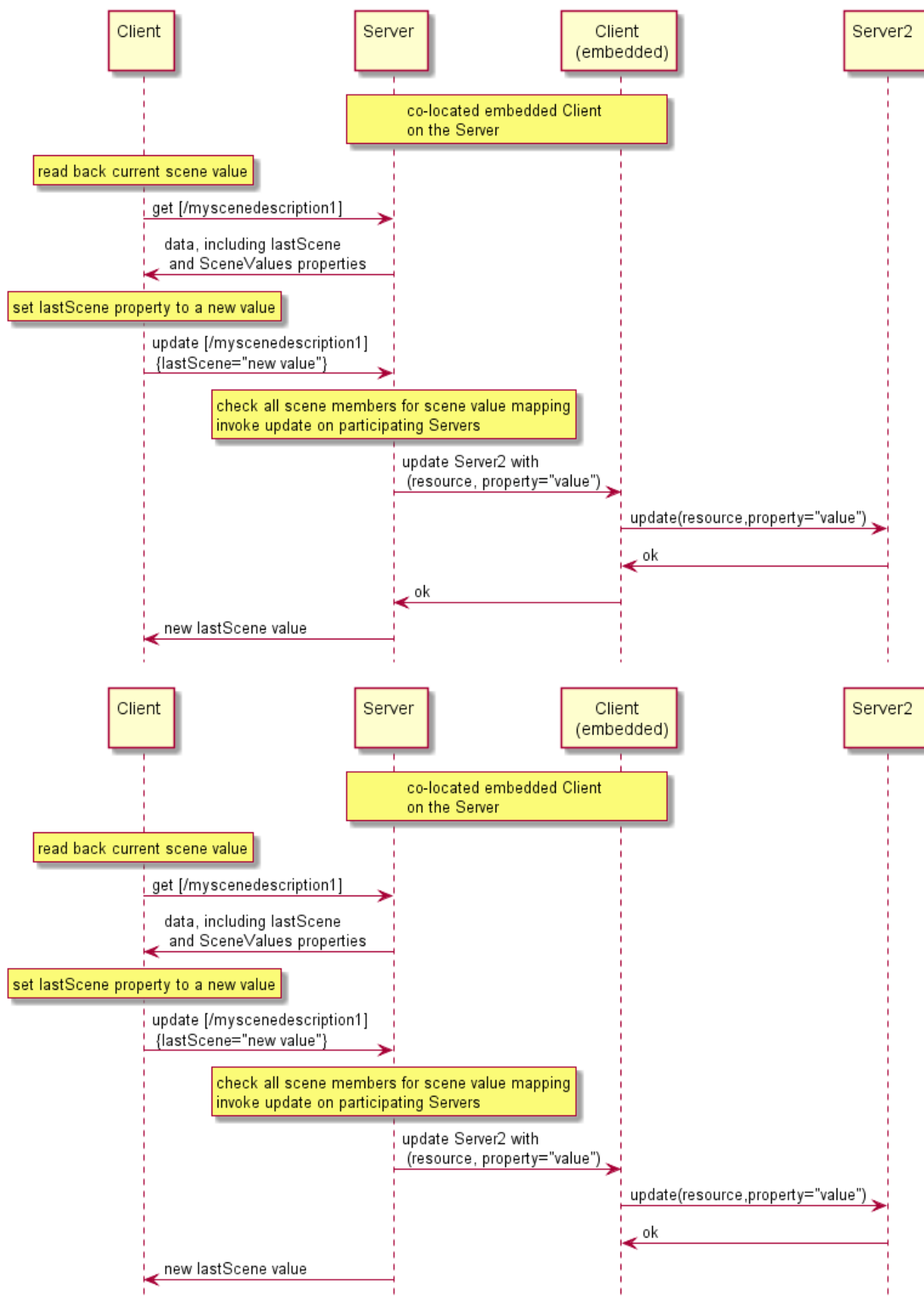


Figure 8 – Interaction overview due to a Scene change

5.4.2.4 Summary of Resource Types defined for Scene functionality

Table 19 summarizes the list of Resource Types that are part of Scenes. Complete details are provided in annex A.7, annex A.8, and annex A.9.

Table 19 – list of Resource Types for Scenes

Friendly Name (informative)	Resource Type (rt)	Short Description	Clause
sceneList	"oic.wk.scenelist"	Top Level Collection containing sceneCollections	A.7
sceneCollection	"oic.wk.sceneCollection"	Description of zero or more scenes	A.8
sceneMember	"oic.wk.sceneMember"	Description of mappings for each specific Resource part of the sceneCollection	A.9

5.4.3 Security considerations

Creation of Scenes on a Server that is capable of this functionality is dependent on the ACLs applied to the Resources and the Client having the appropriate permissions. Interaction between a Client (embedded or separate) and a Server that hosts the Resource that is referenced as a Scene Member is contingent on the Client having appropriate permissions to access the Resource on the host Server.

See ISO/IEC 30118-2 for details on the use of ACLs and also the mechanisms around Device Authentication that are necessary to ensure that the correct permissions exist for the Client to access the Scene Member Resource(s) on the Server.

5.5 Rules

5.5.1 Overview

Rules are Resources that implement autonomous decision logic according to a condition-action pattern. The Rule is evaluated based on the Property values of selected Resource instances. Rule Actions are triggered when a Rule Expression evaluates to "true" and consist of defined UPDATE operations that act upon Scene Collections by updating the "lastScene" Property to a defined value.

A Rule has the following components:

- A Collection of Links to the Resources (i.e., Rule Inputs) that contain the Properties whose values are evaluated as part of the Rule Expression.
- One Rule Expression that defines the Rule logic in terms of the defined Rule Inputs, and which evaluates to a boolean Rule Result, for which "true" means that the Rule has been triggered.
- A Collection of Links to one or more Rule Actions, which are processed when the Rule Result is evaluated to "true"; the Rule Action provides a specific value for the "lastScene" Property that is updated in the Linked Scene Collection.

Figure 9 shows how these components are organized with respect to the Rule Resource.

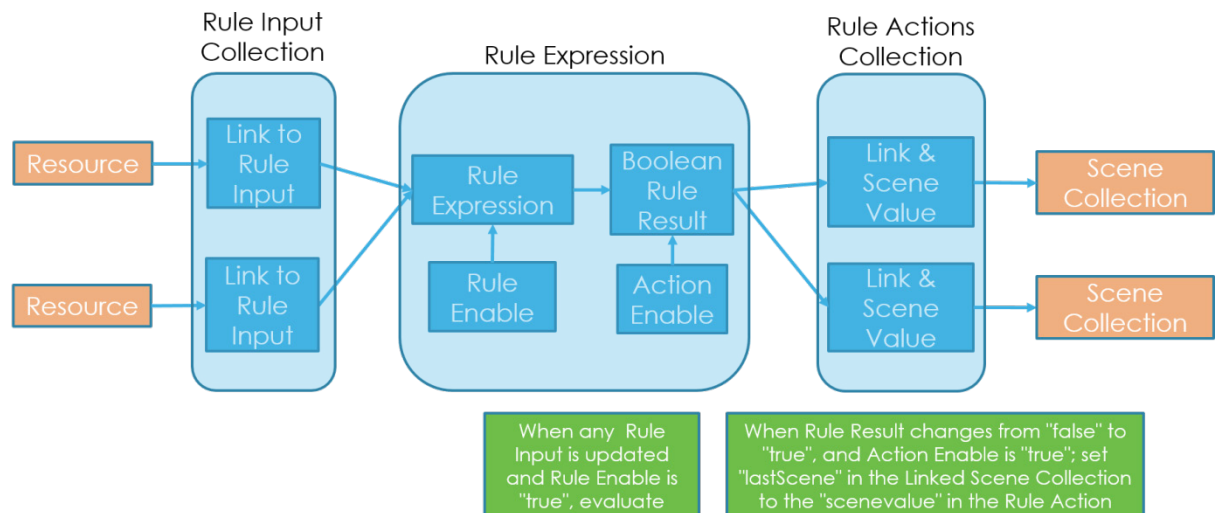


Figure 9 – Components of a Rule

5.5.2 Rule structure

5.5.2.1 Introduction and Rule Resource

A Rule is a Resource with a Resource Type of "oic.r.rule" that is a Collection. A Rule instance shall contain the following:

- A single Link to a Rule Input Collection (see clause 5.5.2.2).
- A single Link to a Rule Expression Resource (see clause 5.5.2.3) which contains:
 - One Rule Expression Property.
 - One boolean Rule Enable Property, which controls whether the Rule is or is not active.
 - One boolean Action Enable Property, which controls whether the Rule actions are or are not triggered when the Rule Result evaluates to "true"
 - One boolean Rule Result Property, which reflects the result of the evaluation of the Rule Expression.
- A single Link to a Rule Action Collection (see clause 5.5.2.4).

A summary for the Rule, Rule Input Collection, Rule Expression, Rule Action, and Rule Action Collection Resource Types is provided in Table 20.

Table 20 – Optional Rule Resources

Example URI	Resource Type Title	Resource Type ID ("rt" value)	OCF Interfaces	Description	Related Functional Interaction
/example/ruleURI	Rule	"oic.r.rule"	"oic.if.ll", "oic.if.base line"	The Resource through which the Device exposes Rules. The Properties exposed by "oic.r.rule" are listed in Table 21.	Rules
/example/ruleinputcollectionURI	Rule Input Collection	"oic.r.rule.inputcollection"	"oic.if.ll", "oic.if.base line"	A specialisation of a Collection that contains Links to the locally hosted Resources that provide input to the Rule Expression. The Properties exposed by "oic.r.rule.inputcollection" are listed in Table 22	Rules
/example/ruleexpressionURI	Rule Expression	"oic.r.rule.expression"	"oic.if.rw", "oic.if.baseline"	The Resource that contains the Rule Expression and Rule Result.	Rules

				The Properties exposed by "oic.r.rule.expression" are listed in Table 24.	
"/example/ruleactionURI"	Rule Action	"oic.r.rule.action"	"oic.if.rw","oic.if.baseline"	The Resource that contains the action to be taken on evaluation of the Rule Result to true. The Properties exposed by "oic.r.rule.action" are listed in Table 26.	Rules
"/example/ruleactioncollectionURI"	Rule Action Collection	"oic.r.rule.actioncollection"	"oic.if.ll","oic.if.baseline"	A specialisation of a Collection that contains only instances of "oic.r.rule.ruleaction". The Properties exposed by "oic.r.rule.actioncollection" are listed in Table 25.	Rules

The Rule ("oic.r.rule") Resource is described in Table 21. Complete details are provided in annex A.13.

Table 21 – "oic.r.rule" Resource Type definition

Property title	Property name	Value type	Value rule	Unit	Access mode	Mandatory	Description
Links	"links"	"array"	See Table 13 in ISO/IEC 30118-1		R	Yes	See Table 13 in ISO/IEC 30118-1.
Resource Type	"rt"	"array"	["oic.r.rule"]		R	Yes	See Table 4 in ISO/IEC 30118-1
Resource Types	"rts"	"array"	Resource Types that may be linked from the Rule		R	Yes	See Table 11 in [Bookmark to Core Spec]

5.5.2.2 Rule input

Rule Inputs are Links in a Collection ("oic.r.rule.inputcollection") that is itself Linked from the Rule. Each Link in the Collection corresponds to a different input variable in the Rule Expression. Each Link therefore corresponds to a Resource defined by the Rule. For example, a Rule that evaluates a temperature input will include a Link to a Resource with a Resource Type of "oic.r.temperature". Resource Types for Rule Inputs shall be identical to the Resources to which they are linked. The Link has an "if" Link Parameter, which shall be a single element array containing the OCF Interface used for the internal observe of the input Resource.

The Rule Input Collection ("oic.r.rule.inputcollection") Resource is described in Table 22. Complete details are provided in annex A.14.

Table 22 – "oic.r.rule.inputcollection" Resource Type definition

Property title	Property name	Value type	Value rule	Unit	Access mode	Mandatory	Description
Links	"links"	"array"	See Table 13 in ISO/IEC 30118-1		R	Yes	See Table 13 in ISO/IEC 30118-1.
Resource Type	"rt"	"array"	["oic.r.rule.inputcollection"]		R	Yes	See Table 4 in ISO/IEC 30118-1.
Resource Types	"rts"	"array"	Resource Types that may be linked from the Rule Input Collection		R	Yes	See Table 11 in ISO/IEC 30118-1.

Rule Inputs shall be indicated by the Link relation type "ruleinput" in the Link (i.e., "rel" Parameter), thus semantically describing the relationship between the "href" and the "anchor" Parameters contained in the Link. The "href" Parameter of the Link shall correspond to the input Resource and shall be a relative URI to a Resource that is hosted on the same Device as the Rule. The "anchor" Parameter of the Link corresponds to the variable name in the Rule Expression, the variable name (and thus the content of "anchor") is defined by the Rule Expression and shall be unique within the context of the Rule Expression. For example a Rule Expression with a variable of "mytemperature:temperature" has an associated Link within the Rule Input Collection with the "anchor" set to "mytemperature". For example:

```
{
  "anchor": "mytemperature",
  "href": "/mylocaltemperaturesensor",
  "rel": ["ruleinput"],
  "rt": ["oic.r.temperature"],
  "if": ["oic.if.s"]
}
```

5.5.2.3 Rule expression

The Rule Expression is a Resource ("oic.r.rule.expression") that contains a "rule" Property, which is defined as a string that contains a logical expression over the Rule Inputs, and which evaluates to a boolean value. This value is exposed in the "ruleresult" Property, which shall have a default value of "false". The expression shall conform to the ABNF syntax defined in clause 5.5.5.

Rule Inputs within the "rule" Property are specified by including the "anchor" Link Parameter from the associated Rule Input, including the desired Property Name from the linked Resource. Figure 10 shows an example for a Rule Input with an "anchor" named "mytemperature" following on from the example Rule Input shown in clause 5.5.2.2, and the Property Name of "temperature", thus the name "mytemperature:temperature" is used in the Rule Expression to refer to this Rule Input. Specifically, that is the value of the "temperature" Property of the Resource at the "href" of "/mylocaltemperaturesensor".

```
mytemperature:temperature >= "25"
```

Figure 10 – Example "rule" Property with single Rule Input

There are no restrictions on the number of Rule Inputs that may be part of a Rule Expression. Consider an additional example where the Rule Result evaluates to "true" if the temperature is greater than or equal to "25" and a door is open ("openState" Property of an instance of "oic.r.door"). Thus, given a Rule Input for "mydoor" as shown in Figure 11 we can construct the Rule Expression shown in Figure 12.

```
{
  "anchor": "mydoor",
  "href": "/mylocaldoor",
  "rel": ["ruleinput"],
  "rt": ["oic.r.door"],
  "if": ["oic.if.a"]
}
```

Figure 11 – Example Link to Rule Input Resource for "mydoor"

```
mytemperature:temperature >= "25" and mydoor:openState contains
"Open"
```

Figure 12 – Example "rule" Property with more than one Rule Input

The Rule Expression also contains two Properties that allow the enabling of the Rule and the actuating of any Rule Actions to be controlled by a Client.

The "ruleenable" Property controls whether the "ruleresult" Property is updated upon processing of the Rule Expression. If the "ruleenable" Property is set to "true", then the "ruleresult" Property shall be set according to evaluation of the Rule Expression each time any Rule Input changes, in effect the Rule observes the Rule Inputs. An initial evaluation of the Rule Expression shall occur when "ruleenable" is set to "true"; subsequent re-evaluation shall only take place when any of the Rule Input values change. If the "ruleenable" Property is set to "false", the Rule Expression shall not be re-evaluated irrespective of the state of the Rule Inputs. If the "ruleenable" Property is set to "false", the Server shall not change the values of any of the other Properties in the instance of Rule Expression; this simply means that the Rule is no longer re-evaluated whenever a Rule Input changes. The "ruleenable" Property shall have a default value of "false". A Server shall only expose a "ruleenable" Property set to "true" (or allow it to be set to "true" by a Client, such an attempt shall be rejected with an appropriate failure reason, e.g. "bad request") if there exists a Rule Input Link for each variable within the Rule Expression.

The "actionenable" Property controls whether the Rule Actions (see clause 5.5.2.4) are processed when the "ruleresult" Property is set to "true". If the "actionenable" Property is set to "true", all Rule Actions shall be processed when the "ruleresult" Property is set to "true" as a result of Rule Expression evaluation. No action shall be taken if the result of Rule Expression evaluation is "false", that is, the "ruleresult" Property changes from "true" to "false". If the "actionenable" Property is set to "false", the Rule Actions shall not be processed. The "actionenable" Property shall have a default value of "false".

Setting the "ruleenable" and "actionenable" Properties places the Rule into one of four modes that are summarized in Table 23.

Table 23 – Summary of "ruleenable" and "actionenable" Property behaviours

"ruleenable" Value	"actionenable" Value	Rule Expression Processed	Rule Action Processed	Notes
false	false	No	No	These are the default initial settings for a Rule
false	true	No	Yes if the "ruleresult" Property is set to "true" by a Client	The "ruleresult" Property may be updated from "false" to "true" in order to manually trigger the processing of any Rule Actions.
true	false	Yes	No	Rule Inputs may be updated and the "ruleresult" Property may be observed to test the logic and processing of the Rule Expression
true	true	Yes	Yes	The Rule Expression is processed and the "ruleresult" Property is updated when Rule Inputs are updated. Rule Actions are processed when the "ruleresult" Property value is set to "true"

As noted in Table 4; the "ruleresult" may be set to "true" by a Client. A Client shall only be able to set "ruleresult" in the mode that is noted, that is when "ruleenable" is "false" and "actionenable" is "true".

The Properties of the Rule Expression Resource are summarized in Table 24. Complete details are provided in annex A.15. An example flow showing use of the Rule Enable and Action Enable Properties is shown in Figure 13.