OIC RESOURCE TYPE SPECIFICATION V1.0.0

Open Interconnect Consortium (OIC)
admin@openinterconnect.org

Legal Disclaimer

NOTHING CONTAINED IN THIS DOCUMENT SHALL BE DEEMED AS GRANTING YOU ANY KIND OF LICENSE IN ITS CONTENT, EITHER EXPRESSLY OR IMPLIEDLY, OR TO ANY INTELLECTUAL PROPERTY OWNED OR CONTROLLED BY ANY OF THE AUTHORS OR DEVELOPERS OF THIS DOCUMENT. THE INFORMATION CONTAINED HEREIN IS PROVIDED ON AN "AS IS" BASIS, AND TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, THE AUTHORS AND DEVELOPERS OF THIS SPECIFICATION HEREBY DISCLAIM ALL OTHER WARRANTIES AND CONDITIONS, EITHER EXPRESS OR IMPLIED, STATUTORY OR AT COMMON LAW, INCLUDING, BUT NOT LIMITED TO, IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. OPEN INTERCONNECT CONSORTIUM, INC. FURTHER DISCLAIMS ANY AND ALL WARRANTIES OF NON-INFRINGEMENT, ACCURACY OR LACK OF VIRUSES.

CONTENTS

1 Scope 13

2 Normative references 13

3 Terms, definitions, symbols and abbreviations 13

3.1 Terms and definitions 13

3.3 Symbols and abbreviations 14

3.4 Conventions 14

4 Document conventions and organization 14

4.1 Notation 14

4.2 Data types 15

5 Baseline Model Constructs 15

5.1 URI 15

5.2 Interfaces 16

5.3 RAML definition 16

5.4 Property definition 17

5.4.1 Common Properties 17

5.4.2 Resource Properties 17

5.4.3 Basic Resource Schema 18

5.4.4 CRUDN Operation Response Codes 20

5.4.5 ‘id’ property 21

5.5 Example Resource Definitions 21

5.6 Observable Resources 24

5.7 Composite resources 24

6 Resource Type definitions 25

6.1 Air Flow 28

6.1.1 Introduction 28

6.1.2 Example URI 28

6.1.3 Resource Type 28

6.1.4 RAML Definition 28

6.1.5 Property Definition 31

6.1.6 CRUDN behavior 32

6.2 Air Flow Control 32

6.2.1 Introduction 32

6.2.2 Example URI 32

6.2.3 Resource Type 32

6.2.4 RAML Definition 32

6.2.5 Property Definition 36

6.2.6 CRUDN behavior 36

6.3 Battery 36

6.3.1 Introduction 36

6.3.2 Example URI 36

6.3.3 Resource Type 36

6.3.4 RAML Definition 36

6.3.5 Property Definition 37

6.3.6 CRUDN behavior 37

6.4 Binary Switch 37

6.4.1 Introduction 37

6.4.2 Example URI 37

6.4.3 Resource Type 37

6.4.4 RAML Definition 37

6.4.5 Property Definition 39

6.4.6 CRUDN behavior 39

6.5 Brightness 39

6.5.1 Introduction 39

6.5.2 Example URI 39

6.5.3 Resource Type 40

6.5.4 RAML Definition 40

6.5.5 Property Definition 42

6.5.6 CRUDN behavior 42

6.6 Colour Chroma 42

6.6.1 Introduction 42

6.6.2 Example URI 42

6.6.3 Resource Type 42

6.6.4 RAML Definition 42

6.6.5 Property Definition 45

6.6.6 CRUDN behavior 45

6.7 Colour RGB 45

6.7.1 Introduction 45

6.7.2 Example URI 45

6.7.3 Resource Type 45

6.7.4 RAML Definition 45

6.7.5 Property Definition 47

6.7.6 CRUDN behavior 48

6.8 Dimming 48

6.8.1 Introduction 48

6.8.2 Example URI 48

6.8.3 Resource Type 48

6.8.4 RAML Definition 48

6.8.5 Property Definition 51

6.8.6 CRUDN behavior 51

6.9 Door 51

6.9.1 Introduction 51

6.9.2 Example URI 52

6.9.3 Resource Type 52

6.9.4 RAML Definition ... 52
6.9.5 Property Definition ... 54
6.9.6 CRUDN behavior ... 54
6.10 Energy Consumption ... 54
6.10.1 Introduction ... 54
6.10.2 Example URI ... 54
6.10.3 Resource Type ... 54
6.10.4 RAML Definition ... 54
6.10.5 Property Definition ... 55
6.10.6 CRUDN behavior ... 56
6.11 Energy Usage ... 56
6.11.1 Introduction ... 56
6.11.2 Example URI ... 56
6.11.3 Resource Type ... 56
6.11.4 RAML Definition ... 56
6.11.5 CRUDN behavior ... 57
6.12 Humidity ... 57
6.12.1 Introduction ... 57
6.12.2 Example URI ... 57
6.12.3 Resource Type ... 57
6.12.4 RAML Definition ... 57
6.12.5 Property Definition ... 59
6.12.6 CRUDN behavior ... 60
6.13 Ice Maker ... 60
6.13.1 Introduction ... 60
6.13.2 Example URI ... 60
6.13.3 Resource Type ... 60
6.13.4 RAML Definition ... 60
6.13.5 Property Definition ... 63
6.13.6 CRUDN behavior ... 63
6.14 Lock ... 63
6.14.1 Introduction ... 63
6.14.2 Example URI ... 63
6.14.3 Resource Type ... 63
6.14.4 RAML Definition ... 63
6.14.5 Property Definition ... 66
6.14.6 CRUDN behavior ... 66
6.15 Lock Code ... 66
6.15.1 Introduction ... 66
6.15.2 Example URI ... 66
6.15.3 Resource Type ... 66
6.15.4 RAML Definition ... 66
6.15.5 Property Definition ... 68
6.15.6 CRUDN behavior ... 68

6.16 Mode ... 69
6.16.1 Introduction ... 69
6.16.2 Example URI ... 69
6.16.3 Resource Type ... 69
6.16.4 RAML Definition ... 69
6.16.5 Property Definition ... 72
6.16.6 CRUDN behavior ... 72
6.17 Open Level ... 72
6.17.1 Introduction ... 72
6.17.2 Example URI ... 72
6.17.3 Resource Type ... 72
6.17.4 RAML Definition ... 72
6.17.5 Property Definition ... 75
6.17.6 CRUDN behavior ... 76
6.18 Operational State ... 76
6.18.1 Introduction ... 76
6.18.2 Example URI ... 76
6.18.3 Resource Type ... 76
6.18.4 RAML Definition ... 76
6.18.5 Property Definition ... 80
6.18.6 CRUDN behavior ... 80
6.19 Ramp Time ... 80
6.19.1 Introduction ... 80
6.19.2 Example URI ... 80
6.19.3 Resource Type ... 80
6.19.4 RAML Definition ... 80
6.19.5 Property Definition ... 83
6.19.6 CRUDN behavior ... 83
6.20 Refrigeration ... 84
6.20.1 Introduction ... 84
6.20.2 Example URI ... 84
6.20.3 Resource Type ... 84
6.20.4 RAML Definition ... 84
6.20.5 Property Definition ... 86
6.20.6 CRUDN behavior ... 87
6.21 Temperature ... 87
6.21.1 Introduction ... 87
6.21.2 Example URI ... 87
6.21.3 Resource Type ... 87
6.21.4 RAML Definition ... 87
6.21.5 Property Definition ... 91
6.21.6 CRUDN behavior ... 91
6.22 Time Period ... 91
6.22.1 Introduction ... 91

6.22.2 Example URI 91
6.22.3 Resource Type 91
6.22.4 RAML Definition 91
6.22.5 Property Definition 94
6.22.6 CRUDN behavior 94
6.23 Activity Count 94
6.23.1 Introduction 94
6.23.2 Example URI 94
6.23.3 Resource Type 94
6.23.4 RAML Definition 94
6.23.5 Property Definition 96
6.23.6 CRUDN behavior 96
6.24 Atmospheric Pressure Sensor 96
6.24.1 Introduction 96
6.24.2 Example URI 96
6.24.3 Resource Type 96
6.24.4 RAML Definition 96
6.24.5 Property Definition 97
6.24.6 CRUDN behavior 97
6.25 Audio Controls 97
6.25.1 Introduction 97
6.25.2 Example URI 98
6.25.3 Resource Type 98
6.25.4 RAML Definition 98
6.25.5 Property Definition 100
6.25.6 CRUDN behavior 100
6.26 Auto Focus 100
6.26.1 Introduction 100
6.26.2 Example URI 100
6.26.3 Resource Type 100
6.26.4 RAML Definition 100
6.26.5 Property Definition 102
6.26.6 CRUDN behavior 102
6.27 Automatic Document Feeder 102
6.27.1 Introduction 102
6.27.2 Example URI 103
6.27.3 Resource Type 103
6.27.4 RAML Definition 103
6.27.5 Property Definition 104
6.27.6 CRUDN behavior 104
6.28 Button Switch 104
6.28.1 Introduction 104
6.28.2 Example URI 104
6.28.3 Resource Type 104

6.28.4 RAML Definition 104
6.28.5 Property Definition 105
6.28.6 CRUDN behavior 105
6.29 Carbon Dioxide Sensor 105
6.29.1 Introduction 105
6.29.2 Example URI 105
6.29.3 Resource Type 105
6.29.4 RAML Definition 105
6.29.5 Property Definition 106
6.29.6 CRUDN behavior 106
6.30 Carbon Monoxide Sensor 107
6.30.1 Introduction 107
6.30.2 Example URI 107
6.30.3 Resource Type 107
6.30.4 RAML Definition 107
6.30.5 Property Definition 108
6.30.6 CRUDN behavior 108
6.31 Auto White Balance 108
6.31.1 Introduction 108
6.31.2 Example URI 108
6.31.3 Resource Type 108
6.31.4 RAML Definition 108
6.31.5 Property Definition 110
6.31.6 CRUDN behavior 110
6.32 Colour Saturation 110
6.32.1 Introduction 110
6.32.2 Example URI 110
6.32.3 Resource Type 110
6.32.4 RAML Definition 110
6.32.5 Property Definition 112
6.32.6 CRUDN behavior 113
6.33 Contact Sensor 113
6.33.1 Introduction 113
6.33.2 Example URI 113
6.33.3 Resource Type 113
6.33.4 RAML Definition 113
6.33.5 Property Definition 114
6.33.6 CRUDN behavior 114
6.34 Demand Response Load Control (DRLC) 114
6.34.1 Introduction 114
6.34.2 Example URI 114
6.34.3 Resource Type 114
6.34.4 RAML Definition 114
6.34.5 Property Definition 117

6.34.6 CRUDN behavior 117
6.35 Energy Overload/Circuit Breaker 118
6.35.1 Introduction 118
6.35.2 Example URI 118
6.35.3 Resource Type 118
6.35.4 RAML Definition 118
6.35.5 Property Definition 119
6.35.6 CRUDN behavior 119
6.36 Generic Sensor 119
6.36.1 Introduction 119
6.36.2 Example URI 119
6.36.3 Resource Type 119
6.36.4 RAML Definition 119
6.36.5 Property Definition 120
6.36.6 CRUDN behavior 120
6.37 Glass Break Sensor 120
6.37.1 Introduction 120
6.37.2 Example URI 120
6.37.3 Resource Type 120
6.37.4 RAML Definition 120
6.37.5 Property Definition 121
6.37.6 CRUDN behavior 121
6.38 Heart Rate Zone 121
6.38.1 Introduction 121
6.38.2 Example URI 121
6.38.3 Resource Type 122
6.38.4 RAML Definition 122
6.38.5 Property Definition 123
6.38.6 CRUDN behavior 123
6.39 Illuminance Sensor 123
6.39.1 Introduction 123
6.39.2 Example URI 123
6.39.3 Resource Type 123
6.39.4 RAML Definition 123
6.39.5 Property Definition 124
6.39.6 CRUDN behavior 124
6.40 Magnetic Field Direction Sensor 124
6.40.1 Introduction 124
6.40.2 Example URI 124
6.40.3 Resource Type 124
6.40.4 RAML Definition 124
6.40.5 Property Definition 125
6.40.6 CRUDN behavior 125
6.41 Media 125

6.41.1 Introduction 125
6.41.2 Example URI 125
6.41.3 Resource Type 125
6.41.4 RAML Definition 126
6.41.5 Property Definition 127
6.41.6 CRUDN behavior 127
6.42 Media Source List 127
6.42.1 Introduction 127
6.42.2 Example URI 128
6.42.3 Resource Type 128
6.42.4 RAML Definition 128
6.42.5 Property Definition 130
6.42.6 CRUDN behavior 131
6.42.7 Referenced JSON schemas 131
6.43 Motion Sensor 131
6.43.1 Introduction 131
6.43.2 Example URI 131
6.43.3 Resource Type 131
6.43.4 RAML Definition 131
6.43.5 Property Definition 132
6.43.6 CRUDN behavior 132
6.44 Night Mode 132
6.44.1 Introduction 132
6.44.2 Example URI 132
6.44.3 Resource Type 133
6.44.4 RAML Definition 133
6.44.5 Property Definition 135
6.44.6 CRUDN behavior 135
6.45 Presence Sensor 135
6.45.1 Introduction 135
6.45.2 Example URI 135
6.45.3 Resource Type 135
6.45.4 RAML Definition 135
6.45.5 Property Definition 136
6.45.6 CRUDN behavior 136
6.46 Pan Tilt Zoom Movement 136
6.46.1 Introduction 136
6.46.2 Example URI 136
6.46.3 Resource Type 136
6.46.4 RAML Definition 136
6.46.5 Property Definition 140
6.46.6 CRUDN behavior 140
6.47 Signal Strength 140
6.47.1 Introduction 140

6.47.2 Example URI 140
6.47.3 Resource Type 140
6.47.4 RAML Definition 140
6.47.5 Property Definition 141
6.47.6 CRUDN behavior 141
6.48 Speech Synthesis-TTS 142
6.48.1 Introduction 142
6.48.2 Example URI 142
6.48.3 Resource Type 142
6.48.4 RAML Definition 142
6.48.5 Property Definition 145
6.48.6 CRUDN behavior 145
6.49 Touch Sensor 145
6.49.1 Introduction 145
6.49.2 Example URI 145
6.49.3 Resource Type 145
6.49.4 RAML Definition 145
6.49.5 Property Definition 146
6.49.6 CRUDN behavior 146
6.50 UV Radiation 146
6.50.1 Introduction 146
6.50.2 Example URI 146
6.50.3 Resource Type 146
6.50.4 RAML Definition 146
6.50.5 Property Definition 147
6.50.6 CRUDN behavior 147
6.51 Water Sensor 147
6.51.1 Introduction 147
6.51.2 Example URI 147
6.51.3 Resource Type 147
6.51.4 RAML Definition 148
6.51.5 Property Definition 148
6.51.6 CRUDN behavior 148

Figures

No table of figures entries found.

Tables

Table 5-1 Conversion between OIC CRUDN and RAML definitions 16

Table 5-3 Property definitions of a Resource in the JSON schema 18

Table 5-4 JSON Schema for OIC Core Specification defined properties............................... 19

Table 5-5 JSON Schema for basic Resource Type 20

Table 5-6 Return codes behaviour in RAML... 20

Table 5-7 RAML example of an Resource representing an Actuator..................................... 21

Table 5-8 RAML example of an Resource specifying a Sensor 23

Table 5-9 RAML example of Composite Resource 24

Table 6-1 Alphabetical list of resource types 25

1 Scope

The OIC Resource Type Specification specifies the Resources that have been defined by OIC that may be exposed by an OIC Device.

Application Profile Device specifications specify device types appropriate to the Profile; such specifications use Resource Type definitions from this document.

This specification is built on top of the Core Specification. The Core Specification specifies the OIC core architecture, interfaces protocols and services to enable the implementation of OIC profiles for IoT usages and ecosystems. The Core specification defines the OIC core architecture with the main architectural components of network connectivity, discovery, data transmission, device & service management and ID & security. The core architecture is scalable to support simple devices (constrained device) and more capable devices (smart device).

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.

OIC Core Specification, *Open Interconnect Consortium Core Specification*, Version 1.0.

JSON SCHEMA, *JSON Schema: Core Definitions and Terminology*, Version 4.0, http://json-schema.org/latest/json-schema-core.html.

RAML, *Restful API modelling language*, Version 0.8. http://raml.org/spec.html.

ISO 8601:2004, *Data elements and interchange formats – information interchange – Representation of dates and times*.

IETF draft-ietf-core-interfaces-02, *CoRE Interfaces*, November 9, 2014. http://www.ietf.org/id/draft-ietf-core-interfaces-02.txt.

CIE CIE159:2004, *A colour appearance model for colour management systems: CIECAM02*, January 19, 2004.

http://www.cie.co.at/index.php/Publications/index.php?i_ca_id=435

3 Terms, definitions, symbols and abbreviations

3.1 Terms and definitions

3.1.1
Actuator
OIC Resource with support of the update operation.

3.1.2
Sensor
OIC Resource without support of the update operation.

3.2

3.2.1

TBD

To Be Determined

3.2.2

Resource

Term used as OIC Resource.

3.3 Symbols and abbreviations

3.3.1

CRUDN

Create Read Update Delete Notify

This is an acronym indicating which operations are possible on the resource.

3.3.2

CSV

Comma Separated Value List

Comma Separated Value List is a 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.

3.3.3

OIC

Open Interconnect Consortium

OIC is the standards organization which created and owns this specification.

3.3.4

RAML

RESTful API modelling language

RAML is a simple and succinct way of describing practically-**RESTful APIs**. See **RAML**.

3.3.5

REST

Representational State Transfer

REST is an architecture style for designing networked applications and relies on a stateless, client-server, cacheable communications protocol.

3.4 Conventions

In this specification 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.

4 Document conventions and organization

This document lists all the Resources currently specified by OIC. The Resources are used by Application Profile device definitions. The Resources mentioned in this document can be used by any OIC conforming device in any collection or device representation.

For the purposes of this document, the terms and definitions given in OIC 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 OIC 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).

> These features add functionality supported by OIC Core Architecture and should be implemented. Recommended features take advantage of the capabilities OIC 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 (or allowed).

> These features are neither required nor recommended by OIC 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 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.

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*.

4.2 Data types

See OIC Core Specification.

5 Baseline Model Constructs

5.1 URI

The resource URIs mentioned in this document are non-normative, they may be vendor defined, but are mentioned here for completeness of a CRUDN (REST) definition.

An Instance of a Resource is indicated by the URI. When more than one instance of the same Resource is used in an OIC device, different URIs for the different Resource instances shall be used.

5.2 Interfaces

The OIC Core Specification specifies that all resources have associated with them at least one interface; this interface is advertised during resource discovery. In addition the Core defines a number of interfaces that can be applied to a resource.

The default interface associated with all resources identified in this specification shall be: oic.if.a (Actuator). Where a resource supports the use of additional or alternative interfaces this will be noted in the resource specific specification text.

5.2.1.1 Retrieve Behaviour

On reception of a valid Retrieve request an OIC Server hosting the resource that is the target of the request generates a response depending on the interface included in the request.

5.2.1.2 Update Behaviour

On reception of a valid Update request an OIC Server hosting the resource that is the target of the request shall generate a response depending on the interface included in the request.

An Update request that includes properties designated in the schema as ReadOnly shall be processed by the OIC Server as if the ReadOnly properties were not included (that is, they are ignored).

An Update request is allowed to omit optional properties.

The properties in the Update request shall be returned in the response.

5.3 RAML definition

The RAML definitions used in this document are normative.

The RAML definitions are used to describe the payloads of the CRUDN operations on the specified Resource. The CRUDN operations are defined in the OIC Core Specification. The Core also specifies additional properties in the payloads of the CRUDN operations. A vendor can’t use the RAML definitions in this document directly to create an implementation, additional properties defined in the Core specification needs to be added to create a compliant implementation. The CRUDN operations in this document are defined only for success path scenarios, failure cases and responses thereon align with the core resource model definitions for CRUDN. This specification defines a set of ‘200 class’ responses indicating a successful operation; specifics on the use of these responses are defined in Table 5-6 Return codes behaviour in RAML Table 5-6. Note that the actual values of success and error conditions are defined in the OIC Core Specification.

The RAML definitions map the OIC CRUDN behaviour to the RAML as defined in Table 5-1.

Table 5-1 Conversion between OIC CRUDN and RAML definitions[1]

Resource	**Create**	**Read**	**Update**	**Delete**	**Notify**
/example	put or post	get	put or post	delete	

Notify is not part of an RAML definition but is defined in the Core specification. The semantics of a Notify are the same as the CRUDN Read value.

[1] Please refer to OIC Core Specification Table 30 for detailed semantics around the appropriate use of CoAP request methods

5.4 Property definition

5.4.1 Common Properties

The OIC Core Specification specifies a number of properties that may be defined for OIC resources. The properties identified in Table 5-2 Common Properties for OIC Resources shall be specified for all resources defined in this specification. These properties are exposed within the well-known Core defined /oic/res resource through which the OIC Server and its available resources are discovered. If a client requires that these properties be included in a resource representation that is provided in response to a RETRIEVE operation then the client shall select the Core defined default interface by specifying this in a queryParameter.

Further, should an OIC Client apply the Core defined batch interface (oic.if.b) to a RETRIEVE operation then the returned set of resource representations shall consist of the resource type name ‘rt’ and ‘if’ Common Properties in addition to the resource properties as defined in Table 5-3 Property definitions of a Resource in the JSON schema.

Table 5-2 Common Properties for OIC Resources

Property Name	Friendly Alias **Name**	**Property Value**	**Value Type**	**Value Rules**	**Access Modes**	**Mandatory**	**Description**
if	interface	Supported interface(s)	String	Enum of supported interfaces	Readonly	Yes	Core defined; interface(s) supported by the Resource
rt	ResType	See Core Spec	string	See Core Spec	Readonly	Yes	Core defined; Resource type. The resource types are defined in this document.
p	Policy	See Core Spec	string	See Core Spec	Readonly	No	Core defined Indicators for whether the resource is discoverable and/or observable.

5.4.2 Resource Properties

The properties against which the CRUDN operations are defined with JSON schemas (see JSON SCHEMA).

A basic Resource is formulated around one single value denoting a physical property.

Such a Resource is specified with the properties as defined Table 5-3. Mandatory in the table means that the property shall be defined as part of the overall resource schema; actual inclusion of the property as part of a returned or generated payload is dependent upon the schema that applies to the operation being invoked.

Table 5-3 Property definitions of a Resource in the JSON schema

Property Name	Friendly Alias **Name**	**Property Value**	**Value Type**	**Value Rules**	**Access Modes**	**Mandatory**	**Description**
n	Name	Implementation dependent	string	None	Read/Write	no	Core defined; friendly name of the Resource
id	identifier	Implementation dependent	string	None	Readonly	yes	Unique identifier of the Resource (over all resources in the OIC device)
Default is <value>,may change dependent on the resource	Default is <value>, may change dependent on the resource	Dependent on the resource	Dependent on the resource	Dependent on the resource	Dependent on the resource	yes	The current value of the resource
range	Range	[Min,Max]	string	Linear range	Readonly	no	Range of input values, specified as a CSV.
x_<vendor>		Implementation dependent	Implementation dependent	Implementation dependent	Implementation dependent	no	Vendor extension of the schema. Shall always start with "x_". Shall always be optional.

For resources, which by their nature have more than one physical parameter, the value property can be replaced with multiple properties specifying the different physical parameters. The type of the value shall be indicated in the RAML definition of the Resource and should be suitable for the conveyed value. The description in Table 5-3 indicates if a property is referenced from the OIC Core Specification.

5.4.3 Basic Resource Schema

All resource types defined herein are represented as previously noted by JSON Schemas. The RAML definitions of the resource types embed the resource type specific schema elements.

The complete resource type definition is made up, as indicated in Table 5-3 Property definitions of a Resource in the JSON schema of properties from the OIC Core Specification, basic properties defined by this specification and resource type specific properties. The following figures show the complete JSON schema for the basic Resource defined in this section; this illustrates how the schema is built up from underlying definitions:

Table 5-4 JSON Schema for OIC Core Specification defined properties

```
{
  "id": "http://openinterconnect.org/schemas/oic.core#",
  "$schema": "http://json-schema.org/schema#",
{

  "id": "http://openinterconnect.org/schemas/oic.core#",

  "$schema": "http://json-schema.org/schema#",

  "title": "Core",

  "$ref": "#/definitions/oic.core",

  "definitions": {

    "oic.core": {

      "type": "object",

      "properties": {

        "n": {

          "type": "string",

          "description": "Friendly name of the resource"

        }

      }

    }

  }

}
```

Table 5-5 JSON Schema for basic Resource Type

```
{
  "id": "http://openinterconnect.org/schemas/oic.r.baseResource#",
  "$schema": "http://json-schema.org/schema#",
  "title": "Base Resource",
  "definitions": {
    "oic.r.baseResource": {
      "type": "object",
      "properties": {
        "id": {
               "type": "string",
               " description": "ReadOnly, Instance ID of this specific resource"
        },
        "value": { "type": "string" },
        "range": { "type": "string" }
      }
    }
  },
  "type": "object",
  "allOf": [
    {"$ref": "oic.core.json#/definitions/oic.core"},
    {"$ref": "#/definitions/oic.r.baseResource"}
  ]
}
```

5.4.4 CRUDN Operation Response Codes

A Resource can be created or updated depending on the resource definition and the allowed CRUDN operations. The operation can have different response codes with different meanings. This is explained in Table 5-6.

Table 5-6 Return codes behaviour in RAML

Response Code	Meaning
200	Payload of the response will confirm the change. The RAML definition will contain a schema to define the payload.
201	Payload is the URL of the resource that was created by the server as a result of a Create operation. The RAML definition will contain schema to define the payload.
204	Ok, everything went well, no payload provided. The RAML definition does not contain a schema. The RAML definition may even omit this value, since it is regarded as default behaviour of an OIC Server.

403	Case 1: In the case of a Retrieve on a resource with the use of a queryParameter selecting specific property values; should the server not support the values provided then this response shall be returned. The response payload will include the allowed values for the queryParameter. Case 2: The server could not create or update the resource due to a problem with the provided payload. For an update, unless otherwise stated in the resource definition, the response payload will include the same schema defined for a 200; indicating the current resource property value(s).

5.4.5 ‘id’ property

The id property is a unique (across the scope of the host OIC Server) instance identifier for a specific instance of the resource. The encoding of this identifier is device and implementation dependent.

5.5 Example Resource Definitions

Table 5-7 RAML example of an Resource representing an Actuator

```
#%RAML 0.8
title: OICExampleActuator
version: v1.0

/ActuatorExample:
  description: |
    ResourceActuatorExample description.
    If the ActuatorExample is implemented as the example in the RAML the next values apply:
    The name of the Resource is “ResourceExample Name”
    The resource type is “oic.r.ActuatorExample”
    The interface (if) is denoting an actuator by having the value oic.if.a.
    The policy property p is indicating no discoverable (bit0=0)and not observable (bit1=0) e.g.0
    The unique identification is “actuator_example_id”
    The value of the ActuatorExample is modeled as integer
    The range of the value of ActuatorExample is between 0 and 100
  get:
    description: |
      retrieves the example resource.
    responses:
      200:
        body:
          application/json:
            schema: |
              {
               "id": "http://openinterconnect.org/schemas/oic.r.baseResource#",
               "$schema": "http://json-schema.org/schema#",
               "title": "AcutatorExample",
               "definitions": {
                 "oic.r.baseResource": {
                   "type": "object",
                   "properties": {
                     "value": { "type": "string" },
                     "range": { "type": "string" }
                     }
                   }
               },
               "type": "object",
               "allOf": [
                 {"$ref": "oic.core.json#/definitions/oic.core"},
                 {"$ref": "#/definitions/oic.r.baseResource"}
               ],
               "required": ["n", "id","value"]
             }
            example: |
              {
                "n":     "ActuatorExample Name",
                "id":    "actuator_example_id",
                "value": "0",
                "range": "0,100"
              }
  post:
    description: |
      sets the actuator value
      example only updates the value of the resource
      it does not change the resource name, although it is allowed to do so.
    body:
      applicaton/json:
        schema: |
             {
               "id": "http://openinterconnect.org/schemas/oic.r.baseResource#",
               "$schema": "http://json-schema.org/schema#",
               "title": "AcutatorExample",
               "definitions": {
                 "oic.r.baseResource": {
                   "type": "object",
                   "properties": {
                     "value": { "type": "string" },
                     "range": { "type": "string" }
                     }
                   }
               },
               "type": "object",
               "allOf": [
                 {"$ref": "oic.core.json#/definitions/oic.core"},
                 {"$ref": "#/definitions/oic.r.baseResource"}
               ],
               "required": ["id","value"]
             }
        example: |
            {
              "id":    "actuator_example_id",
              "value" : 5
            }
    responses:
      200:
        body:
          application/json:
            schema: |
              {
               "id": "http://openinterconnect.org/schemas/oic.r.baseResource#",
               "$schema": "http://json-schema.org/schema#",
               "title": "AcutatorExample",
               "definitions": {
                 "oic.r.baseResource": {
                   "type": "object",
                   "properties": {
                     "value": { "type": "string" },
                     "range": { "type": "string" }
                     }
                   }
               },
               "type": "object",
               "allOf": [
                 {"$ref": "oic.core.json#/definitions/oic.core"},
                 {"$ref": "#/definitions/oic.r.baseResource"}
               ],
               "required": ["id","value"]
              }
            example: |
              {
                 "id":    "actuator_example_id",
                 "value": 5
              }


      204:
```

Table 5-8 RAML example of an Resource specifying a Sensor

```
#%RAML 0.8
title: OICExampleSensor
version: v1.0


/SensorExample:
  description: |
    SensorExample description.
    If the SensorExample is implemented as the example in the RAML the next values apply:
    The name of the Resource is “ResourceExample_Name”
    The resource type is “oic.r.SensorExample”
    The interface (if) is denoting an sensor by having the value oic.if.s.
    The policy property p is indicating discoverable (bit0=0)and observable (bit1=1) e.g. 3
    The unique identification is “sensor_example_id”
    The value of the ResourceSensorExample is modeled as integer
    Since the value is only, the optional range property is not specified
  get:
    description: |
      retrieves the example resource.
    responses:
      200:
        body:
          application/json:
            schema: |
              {
               "id": "http://openinterconnect.org/schemas/oic.r.baseResource#",
               "$schema": "http://json-schema.org/schema#",
               "title": "SensorExample",
               "definitions": {
                 "oic.r.baseResource": {
                   "type": "object",
                   "properties": {
                     "value": { "type": "string" },
                     "range": { "type": "string" }
                     }
                   }
               },
               "type": "object",
               "allOf": [
                 {"$ref": "oic.core.json#/definitions/oic.core"},
                 {"$ref": "#/definitions/oic.r.baseResource"}
               ],
               "required": ["n","id","value"]
             }
            example: |
              {
                "n":     "SensorExample_Name",
                "id":    "sensor_example_id",
                "value": "3"
              }
```

5.6 Observable Resources

The Core defines a mechanism by which Resources can advertise themselves as ‘Observable’ to an OIC Client. The use of this Core defined Policy property is entirely implementation dependent.

5.7 Composite resources

Composite resources are resources that comprises of one or more single or other composite resources, an example of which is shown in Table 5-9 RAML example of Composite Resource. The Composite resource can be viewed upon as a new single resource due to that the composite has a new identifier in the system. The composite resource mechanism is a powerful concept since it uses existing resources in new combination to express more contexts to a resource without specifying new single unit resource types.

Composite resources are defined by linking the referenced existing resource values in to a set.

The linking is done by using an array of web-links; refer to the OIC Core Specification section 7.1.6.2 for more details. Note that the example listed below contains a partial schema of this definition as it is for descriptive purpose only. The property name of the array is “resources”. The relationship type shall be “contains”, denoting that the composite contains other resources that make up the composite resource.

The access to the listed resources can be achieved in a single operation by using the Specification defined oic.if.ll interface.

Table 5-9 RAML example of Composite Resource

```
#%RAML 0.8
title: OICExampleCompositeResource
version: v1.0


/CompositeExample:
  description: |
    CompositeExample description.
    If the CompoisteExample is implemented as the example in the RAML the next values apply:
    The name of the Resource is “CompositeExample Name”
    The resource type is “CompositeExample”
    The interface (if) can denote sensor or actuator
    The unique identification is “composite_example_id”
    The value of the ActuatorExample is modeled as 2 references to other implemented resources
    In the example oic.r.SensorExample and oic.r.ActuatorExample are used.
  get:
    description: |
      retrieves the composite example resource.
    responses:
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.baseResource#",
                "$schema": "http://json-schema.org/schema#",
                "title": "SensorExample",
                "definitions": {
                  "oic.r.baseResource": {
                    "type": "object",
                    "properties": {
                       "resources": {
                          "type": "array",
                          "items": {
                             "href": {"type": "string"},
                             "rel":  {"type": "string"},
                             "rt":   {"type": "string"},
                             "if":   {"type": "string"}
                          }
                        }
                      }
                    }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "#/definitions/oic.r.baseResource"}
                ],
                "required": ["n","id","resources"]
              }
            example: |
              {
                "n":     "CompositeExample Name",
                "id":    "composite_example_id",
                "resources": [
                    {
                      "href": "/my_1st_reference",
                      "rel":  "contains",
                      "rt":   "oic.r.ActuatorExample",
                      "if":   "oic.if.a"
                    },
                    {
                      "href": "/my_2nd_reference",
                      "rel":  "contains",
                      "rt":   "oic.r.SensorExample".
                      "if":   "oic.if.s"
                    }
                  ]
              }
```

6 Resource Type definitions

Table 6-1 Alphabetical list of resource types

Friendly Name	Resource Type (rt)	Section

(informative)		
Activity Count	oic.r.sensor.activity.count	6.23
Atmospheric Pressure	oic.r.sensor.atmosphericPressure	6.24
Air Flow	oic.r.airFlow	6.1
Air Flow Control	oic.r.airFlowControl	6.2
Audio Controls	oic.r.audio	6.25
Auto Focus	oic.r.autofocus	6.26
Automatic Document Feeder	oic.r.automaticDocumentFeeder	6.27
Auto White Balance	oic.r.colour.autowhitebalance	6.31
Battery	oic.r.energy.battery	6.3
Binary switch	oic.r.switch.binary	6.4
Brightness	oic.r.light.brightness	6.5
Button Switch	oic.r.button	6.28
Carbon Dioxide Sensor	oic.sensor.carbonDioxide	6.29
Carbon Monoxide Sensor	oic.r.sensor.carbonMonoxide	6.30
Colour Chroma	oic.r.colour.chroma	6.6
Colour RGB	oic.r.colour.rgb	6.7
Colour Saturation	oic.r.colour.saturation	6.32
Contact Sensor	oic.r.sensor.contact	6.33
Demand Response Load Control (DRLC)	oic.r.energy.drlc	6.34
Dimming	oic.r.light.dimming	6.8
Door	oic.r.door	6.9
Energy Consumption	oic.r.energy.consumption	6.10

Energy Overload/Circuit Breaker	oic.r.energy.overload	6.35
Energy Usage	oic.r.energy.usage	6.11
Generic Sensor	oic.r.sensor	6.36
Glass Break Sensor	oic.r.sensor.glassBreak	6.37
Heart Rate Zone Sensor	oic.r.sensor.heart.zone	6.38
Humidity	oic.r.humidity	6.12
Icemaker	oic.r.iceMaker	6.13
Illuminance Sensor	oic.r.sensor.illuminance	6.39
Lock	oic.r.lock.status	6.14
Lock Code	oic.r.lock.code	6.15
Magnetic Field Direction	oic.r.sensor.magneticFieldDirection	6.40
Media	oic.r.media	6.41
Media Source	oic.r.media.source	6.42
Mode	oic.r.mode	6.16
Motion Sensor	oic.r.sensor.motion	6.43
Night Mode	oic.r.nightMode	6.44
Open Level	oic.r.openLevel	6.17
Operational State	oic.r.operational.state	6.18
Pan Tilt Zoom Movement	oic.r.ptz	6.46
Presence Sensor	oic.r.sensor.presence	6.45
Ramp Time	oic.r.light.rampTime	6.19
Refrigeration	oic.r.refrigeration	6.20
Signal Strength	oic.r.signalStrength	6.47

Speech Synthesis	oic.r.speech.tts	6.48
Temperature	oic.r.temperature	6.21
Time Period	oic.r.time.period	6.22
Touch Sensor	oic.r.sensor.touch	6.49
UV Radiation	oic.r.sensor.radiation.uv	6.50
Water Sensor	oic.r.sensor.water	6.51

All resource types in this document are prefixed with “oic.r” denoting that it is an OIC defined resource type.

6.1 Air Flow

6.1.1 Introduction

This resource describes the properties associated with air flow. The direction is the directionality of the air flow if applicable. Direction values are dependent on the capabilities of the unit. The speed is an integer representing the current speed level for the unit. The range is the min,max values for the speed level.

6.1.2 Example URI

/AirFlowResURI

6.1.3 Resource Type

The resource type (rt) is defined as: oic.r.airFlow.

6.1.4 RAML Definition

```
#%RAML 0.8
title: OICAirFlow
version: v1.0-20150805
traits:
 - interface :
     queryParameters:
       if:
         enum: ["oic.if.a"]

/AirFlowResURI:
  description: |
    This resource describes the properties associated with air flow.
    The direction is the directionality of the air flow if applicable.
    Direction values are dependent on the capabilities of the unit.
    The speed is an integer representing the current speed level for the unit.
    The range is the min,max values for the speed level.

  is : ['interface']
  get:
    description: |
      Retrieves the current air flow values.

    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.airFlow#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Air Flow",
                "definitions": {
                  "oic.r.airFlow": {
                    "type": "object",
                    "properties": {
                      "direction": {
                        "type": "string",
                        "description": "Directionality of the air flow"
                      },
                      "speed": {
                        "type": "integer",
                        "description": "Current speed level"
                      },
                      "range": {
                        "type": "string",
                        "description": "ReadOnly, Min,max values for the speed level"
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.airFlow"}
                ],
                "required": ["speed"]
              }

            example: |
              {
                "rt":         "oic.r.airFlow",
                "id":         "unique_example_id",
                "direction":  "left",
                "speed":      5,
                "range":      "1,7"
              }

  post:
    description: |
      Sets the current air flow values.
      Only direction and speed may be set by an update operation.

    body:
      application/json:
        schema: |
          {
            "id": "http://openinterconnect.org/schemas/oic.r.airFlow#",
            "$schema": "http://json-schema.org/draft-04/schema#",
            "title": "Air Flow",
            "definitions": {
              "oic.r.airFlow": {
                "type": "object",
                "properties": {
                  "direction": {
                    "type": "string",
                    "description": "Directionality of the air flow"
                  },
                  "speed": {
                    "type": "integer",
                    "description": "Current speed level"
                  },
                  "range": {
                    "type": "string",
                    "description": "ReadOnly, Min,max values for the speed level"
                  }
                }
              }
            },
            "type": "object",
            "allOf": [
              {"$ref": "oic.core.json#/definitions/oic.core"},
              {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
              {"$ref": "#/definitions/oic.r.airFlow"}
            ],
            "required": ["speed"]
          }

        example: |
          {
            "id":         "unique_example_id",
            "direction":  "right",
            "speed":      3
          }

    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.airFlow#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Air Flow",
                "definitions": {
                  "oic.r.airFlow": {
                    "type": "object",
                    "properties": {
                      "direction": {
                        "type": "string",
                        "description": "Directionality of the air flow"
                      },
                      "speed": {
                        "type": "integer",
                        "description": "Current speed level"
                      },
                      "range": {
                        "type": "string",
                        "description": "ReadOnly, Min,max values for the speed level"
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.airFlow"}
                ],
                "required": ["speed"]
              }

            example: |
              {
                "id":         "unique_example_id",
                "direction":  "right",
                "speed":      3
              }

      403:
        description: |
          This response is generated by the OIC Server when the client sends:
            An update with an invalid property value for direction.
            An update with an out of range property value for speed.
          The server responds with the current resource representation.

        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.airFlow#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Air Flow",
                "definitions": {
                  "oic.r.airFlow": {
                    "type": "object",
                    "properties": {
                      "direction": {
                        "type": "string",
                        "description": "Directionality of the air flow"
                      },
                      "speed": {
                        "type": "integer",
                        "description": "Current speed level"
                      },
                      "range": {
                        "type": "string",
                        "description": "ReadOnly, Min,max values for the speed level"
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.airFlow"}
                ],
                "required": ["speed"]
              }

            example: |
              {
                "id":         "unique_example_id",
                "direction":  "right",
                "speed":      3
              }

```

6.1.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
direction	string		Read Write	Directionality of the air flow
speed	integer	yes	Read Write	Current speed level

range	string		Read Only	Min,Max Values For The Speed Level

6.1.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/AirFlowResURI		get	post		

6.2 Air Flow Control

6.2.1 Introduction

This resource describes the attributes associated with control of air flow. For example as modelled by a Thermostat (fan), Room A/C or other device. The resource is a composite resource being made up as a collection of: AirFlow Resource BinarySwitch Resource

6.2.2 Example URI

/AirFlowControlResURI

6.2.3 Resource Type

The resource type (rt) is defined as: oic.r.airFlowControl.

6.2.4 RAML Definition

```
#%RAML 0.8
title: OICAirFlowControl
version: v1.0-20150805
traits:
 - interface-b :
     queryParameters:
       if:
         enum: ["oic.if.b"]
 - interface-all :
     queryParameters:
       if:
         enum: ["oic.if.ll", "oic.if.b"]

/AirFlowControlResURI:
  description: |
    This resource describes the attributes associated with control of air flow.
    For example as modelled by a Thermostat (fan), Room A/C or other device.
    The resource is a composite resource being made up as a collection of:
      AirFlow Resource
      BinarySwitch Resource

  is : ['interface-all']
  get:
    description: |
      Retrieves the current air flow control values.

    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.airFlowControl#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Air Flow Control",
                "definitions": {
                  "oic.r.airFlowControl": {
                    "type": "object",
                    "properties": {
                      "airFlowControl": {
                        "type": "array",
                        "minItems": 2,
                        "maxItems": 2,
                        "items": {
                          "$ref": "oic.web-link.json#"
                        }
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.airFlowControl"}
                ],
                "required": ["airFlowControl"]
              }

            example: |
              {
                "rt": "oic.r.airFlowControl",
                "id": "unique_example_id",
                "airFlowControl": [
                  {
                    "href": "/BinarySwitchResURI",
                    "rel":  "contains",
                    "rt":   "oic.r.switch.binary",
                    "if":   "oic.if.a"
                  },
                  {
                    "href": "/AirFlowResURI",
                    "rel":  "contains",
                    "rt":   "oic.r.airFlow",
                    "if":   "oic.if.a"
                  }
                ]
              }

  post:
    description: |
      Sets the current air flow control values using the batch interface

    body:
      application/json:
        schema: |
          {
            "id": "http://openinterconnect.org/schemas/oic.r.airFlowControl#",
            "$schema": "http://json-schema.org/draft-04/schema#",
            "title": "Air Flow Control",
            "definitions": {
              "oic.r.airFlowControl": {
                "type": "object",
                "properties": {
                  "airFlowControl": {
                    "type": "array",
                    "items": {
                      "anyOf": [
                        {"$ref": "oic.r.switch.binary.json#"},
                        {"$ref": "oic.r.airFlow.json#"}
                      ]
                    }
                  }
                }
              }
            },
            "type": "object",
            "allOf": [
              {"$ref": "oic.core.json#/definitions/oic.core"},
              {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
              {"$ref": "#/definitions/oic.r.airFlowControl"}
            ],
            "required": ["airFlowControl"]
          }

        example: |
          {
            "id":     "unique_example_id",
            "airFlowControl": [
              {
                "id":    "unique_example_id",
                "value": true
              },
              {
                "id":         "unique_example_id",
                "direction":  "right",
                "speed":      3
              }
            ]
          }

    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.airFlowControl#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Air Flow Control",
                "definitions": {
                  "oic.r.airFlowControl": {
                    "type": "object",
                    "properties": {
                      "airFlowControl": {
                        "type": "array",
                        "items": {
                          "anyOf": [
                            {"$ref": "oic.r.switch.binary.json#"},
                            {"$ref": "oic.r.airFlow.json#"}
                          ]
                        }
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.airFlowControl"}
                ],
                "required": ["airFlowControl"]
              }

            example: |
              {
                "id":     "unique_example_id",
                "airFlowControl": [
                  {
```

```
              "id":    "unique_example_id",
              "value": true
            },
            {
              "id":        "unique_example_id",
              "direction": "right",
              "speed":     3
            }
          ]
        }

403:
  description: |
    This response is generated by the OIC Server when the client sends:
      An update with an invalid property value for direction.
      An update with an out of range property value for speed.
    The server responds with the current resource representation.

  body:
    application/json:
      schema: |
        {
          "id": "http://openinterconnect.org/schemas/oic.r.airFlowControl#",
          "$schema": "http://json-schema.org/draft-04/schema#",
          "title": "Air Flow Control",
          "definitions": {
            "oic.r.airFlowControl": {
              "type": "object",
              "properties": {
                "airFlowControl": {
                  "type": "array",
                  "items": {
                    "anyOf": [
                      {"$ref": "oic.r.switch.binary.json#"},
                      {"$ref": "oic.r.airFlow.json#"}
                    ]
                  }
                }
              }
            }
          },
          "type": "object",
          "allOf": [
            {"$ref": "oic.core.json#/definitions/oic.core"},
            {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
            {"$ref": "#/definitions/oic.r.airFlowControl"}
          ],
          "required": ["airFlowControl"]
        }

      example: |
        {
          "id":     "unique_example_id",
          "airFlowControl": [
            {
              "id":    "unique_example_id",
              "value": true
            },
            {
              "id":         "unique_example_id",
              "direction":  "right",
              "speed":      3
            }
          ]
        }

```

6.2.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
airFlowControl	array	yes		
maxItems				

6.2.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/AirFlowControlResURI		get	post		

6.3 Battery

6.3.1 Introduction

This resource describes a battery function. The charge is an integer showing the current battery charge level. The charge is a percentage in the range 0-100. A value of 0 means fully discharged. A value of 100 means fully charged.

6.3.2 Example URI

/BatteryResURI

6.3.3 Resource Type

The resource type (rt) is defined as: oic.r.energy.battery.

6.3.4 RAML Definition

```
#%RAML 0.8
title: OICBattery
version: v1.0-20150727
traits:
 - interface :
     queryParameters:
       if:
         enum: ["oic.if.s"]

/BatteryResURI:
  description: |
    This resource describes a battery function.
    The charge is an integer showing the current battery charge level.
    The charge is a percentage in the range 0-100.
    A value of 0 means fully discharged.
    A value of 100 means fully charged.

  is : ['interface']
  get:
    description: |
      Retrieves the state of the battery.

    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.energy.battery#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Battery",
                "definitions": {
                  "oic.r.energy.battery": {
                    "type": "object",
                    "properties": {
                      "charge" : {
                        "type": "integer",
                        "description": "ReadOnly, The current charge percentage."
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.energy.battery"}
                ],
                "required": [ "charge" ]
              }

            example: |
              {
                "rt":     "oic.r.energy.battery",
                "id":     "unique_example_id",
                "charge": 50
              }
```

6.3.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
charge	integer	yes	Read Only	The Current Charge Percentage.

6.3.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/BatteryResURI		get			

6.4 Binary Switch

6.4.1 Introduction

This resource describes a binary switch (on/off). The value is a boolean. A value of 'true' means that the switch is on. A value of 'false' means that the switch is off.

6.4.2 Example URI

/BinarySwitchResURI

6.4.3 Resource Type

The resource type (rt) is defined as: oic.r.switch.binary.

6.4.4 RAML Definition

```
#%RAML 0.8
title: OICBinarySwitch
version: v1.0-20150727
traits:
 - interface :
     queryParameters:
       if:
         enum: ["oic.if.a"]


/BinarySwitchResURI:
  description: |
    This resource describes a binary switch (on/off).
    The value is a boolean.
    A value of 'true' means that the switch is on.
    A value of 'false' means that the switch is off.
```

```
is : ['interface']
get:
  responses :
    200:
      body:
        application/json:
          schema: |
            {
              "id": "http://openinterconnect.org/schemas/oic.r.switch.binary#",
              "$schema": "http://json-schema.org/draft-04/schema#",
              "title": "Binary Switch",
              "definitions": {
                "oic.r.switch.binary": {
                  "type": "object",
                  "properties": {
                    "value": {
                      "type": "boolean",
                      "description": "Status of the switch"
                    }
                  }
                }
              },
              "type": "object",
              "allOf": [
                {"$ref": "oic.core.json#/definitions/oic.core"},
                {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                {"$ref": "#/definitions/oic.r.switch.binary"}
              ],
              "required": [ "value" ]
            }

          example: |
            {
              "rt":     "oic.r.switch.binary",
              "id":     "unique_example_id",
              "value":  false
            }

post:
  body:
    application/json:
      schema: |
        {
          "id": "http://openinterconnect.org/schemas/oic.r.switch.binary#",
          "$schema": "http://json-schema.org/draft-04/schema#",
          "title": "Binary Switch",
          "definitions": {
            "oic.r.switch.binary": {
              "type": "object",
              "properties": {
                "value": {
                  "type": "boolean",
                  "description": "Status of the switch"
                }
              }
            }
          },
          "type": "object",
          "allOf": [
            {"$ref": "oic.core.json#/definitions/oic.core"},
            {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
            {"$ref": "#/definitions/oic.r.switch.binary"}
          ],
          "required": [ "value" ]
        }

      example: |
        {
          "id":    "unique_example_id",
          "value": true
        }

  responses :
    200:
      body:
        application/json:
          schema: |
            {
              "id": "http://openinterconnect.org/schemas/oic.r.switch.binary#",
              "$schema": "http://json-schema.org/draft-04/schema#",
              "title": "Binary Switch",
              "definitions": {
                "oic.r.switch.binary": {
                  "type": "object",
                  "properties": {
                    "value": {
                      "type": "boolean",
                      "description": "Status of the switch"
                    }
                  }
                }
              },
              "type": "object",
              "allOf": [
                {"$ref": "oic.core.json#/definitions/oic.core"},
                {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                {"$ref": "#/definitions/oic.r.switch.binary"}
              ],
              "required": [ "value" ]
            }

          example: |
            {
              "id":    "unique_example_id",
              "value": true
            }
```

6.4.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
value	boolean	yes	Read Write	Status of the switch

6.4.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/BinarySwitchResURI		get	post		

6.5 Brightness

6.5.1 Introduction

This resource describes the brightness of a light or lamp. The brightness percentage is an integer showing the current brightness level. A brightness of 0 is the minimum for the resource. A brightness of 100 is the maximum for the resource.

6.5.2 Example URI

/BrightnessResURI

6.5.3 Resource Type

The resource type (rt) is defined as: oic.r.light.brightness.

6.5.4 RAML Definition

```
#%RAML 0.8

title: OICBrightness
version: v1.0-20150727

traits:
 - interface :
     queryParameters:

       if:
         enum: ["oic.if.a"]


/BrightnessResURI:

  description: |
    This resource describes the brightness of a light or lamp.
    The brightness percentage is an integer showing the current brightness level.
    A brightness of 0 is the minimum for the resource.
    A brightness of 100 is the maximum for the resource.


  is : ['interface']

  get:

    description: |
      Retrieves the current brightness level.


    responses :

      200:

        body:
          application/json:

            schema: |

              {
                "id": "http://openinterconnect.org/schemas/oic.r.light.brightness#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Brightness",
                "definitions": {
                  "oic.r.light.brightness": {
                    "type": "object",
                    "properties": {
                      "brightness": {
                        "type": "integer",
                        "description": "Current sensed or set value for Brightness"}
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.light.brightness"}
                ],
                "required": [ "brightness" ]
              }


            example: |

              {
                "rt":         "oic.r.light.brightness",
                "id":         "unique_example_id",
                "brightness": 50
              }

  post:
    description: |
      Sets the desired brightness level.

    body:
      application/json:
        schema: |
          {
            "id": "http://openinterconnect.org/schemas/oic.r.light.brightness#",
            "$schema": "http://json-schema.org/draft-04/schema#",
            "title": "Brightness",
            "definitions": {
              "oic.r.light.brightness": {
                "type": "object",
                "properties": {
                  "brightness": {
                    "type": "integer",
                    "description": "Current sensed or set value for Brightness"}
                }
              }
            },
            "type": "object",
            "allOf": [
              {"$ref": "oic.core.json#/definitions/oic.core"},
              {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
              {"$ref": "#/definitions/oic.r.light.brightness"}
            ],
            "required": [ "brightness" ]
          }

        example: |
          {
            "id":         "unique_example_id",
            "brightness": 10
          }

    responses :
      200:
        description: |
          Indicates that the brightness was changed.
          The new brightness level is provided in the response.

        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.light.brightness#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Brightness",
                "definitions": {
                  "oic.r.light.brightness": {
                    "type": "object",
                    "properties": {
                      "brightness": {
                        "type": "integer",
                        "description": "Current sensed or set value for Brightness"}
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.light.brightness"}
                ],
                "required": [ "brightness" ]
               }

            example: |
               {
                "id":         "unique_example_id",
                "brightness": 10
               }
```

6.5.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
brightness	integer	yes	Read Write	Current sensed or set value for Brightness

6.5.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/BrightnessResURI		get	post		

6.6 Colour Chroma

6.6.1 Introduction

This resource describes the colour using chroma conventions. Properties are hue, saturation and colorspacevalue. Hue and saturation are integer values as defined by the CIECAM02 model definition. Colourspacevalue is a CSV of chromaX, chromaY, colourTemperature (X,Y,T). ChromaX and chromaY are defined by CIE. ColourTemperature is the Mired color temperature.

6.6.2 Example URI

/ColourChromaResURI

6.6.3 Resource Type

The resource type (rt) is defined as: oic.r.colour.chroma.

6.6.4 RAML Definition

```
#%RAML 0.8

title: OICColourChroma
version: v1.0-20150727

traits:
 - interface :
   queryParameters:
    if:
     enum: ["oic.if.a"]


/ColourChromaResURI:
 description: |
  This resource describes the colour using chroma conventions.
  Properties are hue, saturation and colorspacevalue.
  Hue and saturation are integer values as defined by the CIECAM02 model definition.
  Colourspacevalue is a CSV of chromaX, chromaY, colourTemperature (X,Y,T).
  ChromaX and chromaY are defined by CIE.
  ColourTemperature is the Mired color temperature.

 is : ['interface']

 get:
  description: |
   Provides the colour using chroma conventions.

```

```
    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.colour.chroma#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Colour Chroma",
                "definitions": {
                  "oic.r.colour.chroma": {
                    "type": "object",
                    "properties": {
                      "hue":             {
                        "type": "integer",
                        "description": "Hue as defined by the CIECAM02 model definition"
                      },
                      "saturation":      {
                        "type": "integer",
                        "description": "Saturation as defined by the CIECAM02 model definition"
                      },
                      "colourspacevalue": {
                        "type": "string",
                        "description": "CSV of chromaX, chromaY, colourTemperature (X,Y,T)."
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.colour.chroma"}
                ],
                "required": [ "hue", "saturation", "colourspacevalue" ]

              }

            example: |
              {
                "rt":               "oic.r.colour.chroma",
                "id":               "unique_example_id",
                "hue":              13088,
                "saturation":       212,
                "colourspacevalue": "0.51, 0.41, 467"
              }

  post:
    description: |
      Sets current colour chroma values

    body:
      application/json:
        schema: |
          {
            "id": "http://openinterconnect.org/schemas/oic.r.colour.chroma#",
            "$schema": "http://json-schema.org/draft-04/schema#",
            "title": "Colour Chroma",
            "definitions": {
              "oic.r.colour.chroma": {
                "type": "object",
                "properties": {
                  "hue":             {
                    "type": "integer",
                    "description": "Hue as defined by the CIECAM02 model definition"
                  },
                  "saturation":      {
                    "type": "integer",
                    "description": "Saturation as defined by the CIECAM02 model definition"
                  },
                  "colourspacevalue": {
                    "type": "string",
                    "description": "CSV of chromaX, chromaY, colourTemperature (X,Y,T)."
                  }
                }
              }
            },
            "type": "object",
            "allOf": [
              {"$ref": "oic.core.json#/definitions/oic.core"},
              {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
              {"$ref": "#/definitions/oic.r.colour.chroma"}
            ],
            "required": [ "hue", "saturation", "colourspacevalue" ]

          }

        example: |
          {
            "id":               "unique_example_id",
            "hue":              13088,
            "saturation":       212,
            "colourspacevalue": "0.51, 0.41, 467"
          }

    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.colour.chroma#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Colour Chroma",
                "definitions": {
                  "oic.r.colour.chroma": {
                    "type": "object",
                    "properties": {
                      "hue":             {
                        "type": "integer",
                        "description": "Hue as defined by the CIECAM02 model definition"
                      },
                      "saturation":      {
                        "type": "integer",
                        "description": "Saturation as defined by the CIECAM02 model definition"
                      },
                      "colourspacevalue": {
                        "type": "string",
                        "description": "CSV of chromaX, chromaY, colourTemperature (X,Y,T)."
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.colour.chroma"}
                ],
                "required": [ "hue", "saturation", "colourspacevalue" ]

              }

            example: |
              {
                "id":               "unique_example_id",
                "hue":              13088,
                "saturation":       212,
                "colourspacevalue": "0.51, 0.41, 467"
              }
```

6.6.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
hue	integer	yes	Read Write	Hue as defined by the CIECAM02 model definition
saturation	integer	yes	Read Write	Saturation as defined by the CIECAM02 model definition
colourspacevalue	string	yes	Read Write	CSV of chromaX, chromaY, colourTemperature (X,Y,T).

6.6.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/ColourChromaResURI		get	post		

6.7 Colour RGB

6.7.1 Introduction

This resource specifies the actual colour in the RGB space represented as a string. Each colour value is described with a Red, Green, Blue component. These colour values are encoded as comma separated values in the string. The minimum and maximum colour value per component is described by the range value. When the range value is omitted, then the range is [0,255].

6.7.2 Example URI

/ColourRGBResURI

6.7.3 Resource Type

The resource type (rt) is defined as: oic.r.colour.rgb.

6.7.4 RAML Definition

```
#%RAML 0.8

title: OICColourRGB
version: v1.0-20150727

traits:
 - interface :
     queryParameters:

       if:
         enum: ["oic.if.a"]


/ColourRGBResURI:

  description: |
    This resource specifies the actual colour in the RGB space represented as a string.
    Each colour value is described with a Red, Green, Blue component.
    These colour values are encoded as comma separated values in the string.
    The minimum and maximum colour value per component is described by the range value.
    When the range value is omitted, then the range is [0,255].

  is : ['interface']
  get:
    description: |
      Retrieves the current colour in RGB.
      Value is an CSV of integer values in the order R,G,B.

    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.colour.rgb#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Colour RGB",
                "definitions": {
                  "oic.r.colour.rgb": {
                    "type": "object",
                    "properties": {
                      "rgbValue": {
                        "type": "string",
                        "description": "RGB value"
                      },
                      "range":    {
                        "type": "string",
                        "description": "min max value of RGB"
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.colour.rgb"}
                ],
                "required": ["rgbValue"]
              }

            example: |
              {
                "rt":       "oic.r.colour.rgb",
                "id":       "unique_example_id",
                "rgbValue": "255,255,255",
                "range":    "0,255"
              }

  post:
    description: |
      Sets the current colourRGB value

    body:
      application/json:
        schema: |
          {
            "id": "http://openinterconnect.org/schemas/oic.r.colour.rgb#",
            "$schema": "http://json-schema.org/draft-04/schema#",
            "title": "Colour RGB",
            "definitions": {
              "oic.r.colour.rgb": {
                "type": "object",
                "properties": {
                  "rgbValue": {
                    "type": "string",
                    "description": "RGB value"
                  },
                  "range":    {
                    "type": "string",
                    "description": "min max value of RGB"
                  }
                }
              }
            },
            "type": "object",
            "allOf": [
              {"$ref": "oic.core.json#/definitions/oic.core"},
              {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
              {"$ref": "#/definitions/oic.r.colour.rgb"}
            ],
            "required": ["rgbValue"]
          }

        example: |
          {
            "id":       "unique_example_id",
            "rgbValue": "255,0,0"
          }

    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.colour.rgb#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Colour RGB",
                "definitions": {
                  "oic.r.colour.rgb": {
                    "type": "object",
                    "properties": {
                      "rgbValue": {
                        "type": "string",
                        "description": "RGB value"
                      },
                      "range":    {
                        "type": "string",
                        "description": "min max value of RGB"
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.colour.rgb"}
                ],
                "required": ["rgbValue"]
              }

            example: |
              {
                "id":       "unique_example_id",
                "rgbValue": "255,0,0"
              }

```

6.7.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description

rgbValue	string	yes	Read Write	RGB value
range	string		Read Write	min max value of RGB

6.7.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/ColourRGBResURI		get	post		

6.8 Dimming

6.8.1 Introduction

This resource describes a dimming function. The value is an integer showing the current dimming level. The step is the increment between dimmer values. The range is the maximum and minimum values for the dimming value. If the range is omitted [0,100] is assumed. A value of 0 means total dimming; a value of 100 means no dimming.

6.8.2 Example URI

/DimmingResURI

6.8.3 Resource Type

The resource type (rt) is defined as: oic.r.light.dimming.

6.8.4 RAML Definition

```
#%RAML 0.8
title: OICDimming
version: v1.0-20150727
traits:
 - interface :
     queryParameters:
       if:
         enum: ["oic.if.a"]

/DimmingResURI:
  description: |
    This resource describes a dimming function.
    The value is an integer showing the current dimming level.
    The step is the increment between dimmer values.
    The range is the maximum and minimum values for the dimming value.
    If the range is omitted [0,100] is assumed.
    A value of 0 means total dimming; a value of 100 means no dimming.

  is : ['interface']
  get:
    description: |
      Retrieves the current dimming level.

    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.light.dimming#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Dimming",
                "definitions": {
                  "oic.r.light.dimming": {
                    "type": "object",
                    "properties": {
```

```
                    "dimmingSetting":  {
                      "type": "integer",
                      "description": "Current dimming value"
                    },
                    "step":   {
                      "type": "integer",
                      "description": "ReadOnly, step increment for dimming values"
                    },
                    "range":  {
                      "type": "string",
                      "description": "ReadOnly, Min and Max values for the dimming setting"
                    }
                  }
                }
              },
              "type": "object",
              "allOf": [
                {"$ref": "oic.core.json#/definitions/oic.core"},
                {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                {"$ref": "#/definitions/oic.r.light.dimming"}
              ],
              "required": ["dimmingSetting"]
            }

          example: |
            {
              "rt":             "oic.r.light.dimming",
              "id":             "unique_example_id",
              "dimmingSetting": 30,
              "step":           5,
              "range":          "0,100"
            }

post:
  description: |
    Sets the desired dimming level.

  body:
    application/json:
      schema: |
        {
          "id": "http://openinterconnect.org/schemas/oic.r.light.dimming#",
          "$schema": "http://json-schema.org/draft-04/schema#",
          "title": "Dimming",
          "definitions": {
            "oic.r.light.dimming": {
              "type": "object",
              "properties": {
                "dimmingSetting":  {
                  "type": "integer",
                  "description": "Current dimming value"
                },
                "step":   {
                  "type": "integer",
                  "description": "ReadOnly, step increment for dimming values"
                },
                "range":  {
                  "type": "string",
                  "description": "ReadOnly, Min and Max values for the dimming setting"
                }
              }
            }
          },
          "type": "object",
          "allOf": [
            {"$ref": "oic.core.json#/definitions/oic.core"},
            {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
            {"$ref": "#/definitions/oic.r.light.dimming"}
          ],
          "required": ["dimmingSetting"]
        }

      example: |
        {
          "id":            "unique_example_id",
          "dimmingSetting": 40
        }

  responses :
    200:
      description: |
        Indicates that the dimming was changed.
        The new dimming level is provided in the response.

      body:
        application/json:
          schema: |
            {
              "id": "http://openinterconnect.org/schemas/oic.r.light.dimming#",
              "$schema": "http://json-schema.org/draft-04/schema#",
              "title": "Dimming",
              "definitions": {
                "oic.r.light.dimming": {
                  "type": "object",
                  "properties": {
                    "dimmingSetting":  {
                      "type": "integer",
                      "description": "Current dimming value"
                    },
                    "step":   {
                      "type": "integer",
                      "description": "ReadOnly, step increment for dimming values"
                    },
                    "range":  {
                      "type": "string",
                      "description": "ReadOnly, Min and Max values for the dimming setting"
                    }
                  }
                }
              },
              "type": "object",
              "allOf": [
                {"$ref": "oic.core.json#/definitions/oic.core"},
                {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                {"$ref": "#/definitions/oic.r.light.dimming"}
              ],
              "required": ["dimmingSetting"]
            }

          example: |
            {
              "id":            "unique_example_id",
              "dimmingSetting": 40
            }

    403:
      description: |
        This response is generated by the OIC Server when the client sends:
          An update with an out of range property value for dimmingSetting.
        The server responds with the current resource representation.
```

```
body:
  application/json:
    schema: |
      {
        "id": "http://openinterconnect.org/schemas/oic.r.light.dimming#",
        "$schema": "http://json-schema.org/draft-04/schema#",
        "title": "Dimming",
        "definitions": {
          "oic.r.light.dimming": {
            "type": "object",
            "properties": {
              "dimmingSetting":  {
                "type": "integer",
                "description": "Current dimming value"
              },
              "step":   {
                "type": "integer",
                "description": "ReadOnly, step increment for dimming values"
              },
              "range":  {
                "type": "string",
                "description": "ReadOnly, Min and Max values for the dimming setting"
              }
            }
          }
        },
        "type": "object",
        "allOf": [
          {"$ref": "oic.core.json#/definitions/oic.core"},
          {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
          {"$ref": "#/definitions/oic.r.light.dimming"}
        ],
        "required": ["dimmingSetting"]
      }

    example: |
      {
        "id":             "unique_example_id",
        "dimmingSetting": 40
      }
```

6.8.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
dimmingSetting	integer	yes	Read Write	Current dimming value
step	integer		Read Only	Step Increment For Dimming Values
range	string		Read Only	Min And Max Values For The Dimming Setting

6.8.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/DimmingResURI		get	post		

6.9 Door

6.9.1 Introduction

This resource describes a door. A door is modelled by means of openState (Open/Closed). openDuration (ISO 8601 Time). openAlarm (boolean). For openState, the value 'Open' indicates the door is open. The value 'Closed' indicates the door is closed. The type of openDuration is an ISO 8601 Time encoded string. The openAlarm value 'true' indicates that the open alarm is active. The openAlarm value 'false' indicates that open alarm is not active.

6.9.2 Example URI

/DoorResURI

6.9.3 Resource Type

The resource type (rt) is defined as: oic.r.door.

6.9.4 RAML Definition

```
#%RAML 0.8
title: OICDoor
version: v1.0-20150727
traits:
 - interface :
     queryParameters:
       if:
         enum: ["oic.if.a"]

/DoorResURI:
  description: |
    This resource describes a door.
    A door is modelled by means of
      openState (Open/Closed).
      openDuration (ISO 8601 Time).
      openAlarm (boolean).
    For openState, the value 'Open' indicates the door is open.
    The value 'Closed' indicates the door is closed.
    The type of openDuration is an ISO 8601 Time encoded string.
    The openAlarm value 'true' indicates that the open alarm is active.
    The openAlarm value 'false' indicates that open alarm is not active.

  is : ['interface']
  get:
    description: |
      retrieves the state of the Door.

    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.door#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Door",
                "definitions": {
                  "oic.r.door": {
                    "type": "object",
                    "properties": {
                      "openState" : {
                        "enum": ["Open","Closed"],
                        "description": "ReadOnly, The state of the door (open or closed)"
                      },
                      "openDuration": {
                        "type": "string",
                        "description": "ReadOnly, The time duration the door has been open"
                      },
                      "openAlarm": {
                        "type": "boolean",
                        "description": "The state of the door open alarm"
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.door"}
                ],
                "required": ["openState"]
              }

            example: |
              {
                "rt" :          "oic.r.door",
                "id":           "unique_example_id",
                "openState":    "Open",
                "openDuration": "P0Y0M0DT2H25M5S",
                "openAlarm":    true
              }

  post:
    description: |
      Sets the current Door properties.
      The only property that can be set as part of an update operation is
        the openAlarm.
      This can be made active (true) or inactive (false)

    body:
      application/json:
        schema: |
          {
            "id": "http://openinterconnect.org/schemas/oic.r.door#",
            "$schema": "http://json-schema.org/draft-04/schema#",
            "title": "Door",
            "definitions": {
              "oic.r.door": {
                "type": "object",
                "properties": {
                  "openAlarm": {
                    "type": "boolean",
                    "description": "The state of the door open alarm"
                  }
                }
              }
            },
            "type": "object",
            "allOf": [
              {"$ref": "oic.core.json#/definitions/oic.core"},
              {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
              {"$ref": "#/definitions/oic.r.door"}
            ]
          }

        example: |
          {
            "id":         "unique_example_id",
            "openAlarm":  false
          }

    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.door#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Door",
                "definitions": {
                  "oic.r.door": {
                    "type": "object",
                    "properties": {
                      "openAlarm": {
                        "type": "boolean",
                        "description": "The state of the door open alarm"
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.door"}
                ]
              }

            example: |
              {
                "id":         "unique_example_id",
                "openAlarm":  false
              }

```

6.9.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
openState	enum	yes	Read Only	The State Of The Door (Open Or Closed)
openDuration	string		Read Only	The Time Duration The Door Has Been Open
openAlarm	boolean		Read Write	The state of the door open alarm

6.9.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/DoorResURI		get	post		

6.10 Energy Consumption

6.10.1 Introduction

This resource describes the energy consumed by the device since power up. It provides the instantaneous power draw of the device at the time the resource was queried. The power value is in Watts [W]. The energy value is in Watt Hours [Wh].

6.10.2 Example URI

/EnergyConsumptionResURI

6.10.3 Resource Type

The resource type (rt) is defined as: oic.r.energy.consumption.

6.10.4 RAML Definition

```
#%RAML 0.8

title: OICEnergyConsumption
version: v1.0-20150727

traits:
 - interface :
     queryParameters:
       if:
         enum: ["oic.if.s"]

/EnergyConsumptionResURI:
  description: |
    This resource describes the energy consumed by the device since power up.
    It provides the instantaneous power draw of the device at the time the resource was queried.
    The power value is in Watts [W].
    The energy value is in Watt Hours [Wh].

  is : ['interface']
  get:
    description: |
      Provides the current power draw and cumulative energy usage.

    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.energy.consumption#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Energy Consumption",
                "definitions": {
                  "oic.r.energy.consumption": {
                    "type": "object",
                    "properties": {
                      "power":    {
                        "type": "number",
                        "description": "ReadOnly, Instantaneous Power"
                      },
                      "energy":   {
                        "type": "number",
                        "description": "ReadOnly, Energy consumed"}
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.energy.consumption"}
                ],
                "required": ["power", "energy"]
              }

            example: |
              {
                "rt":     "oic.r.energy.consumption",
                "id":     "unique_example_id",
                "power":  2000,
                "energy": 3500
              }

```

6.10.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
power	number	yes	Read Only	Instantaneous Power
energy	number	yes	Read Only	Energy Consumed

6.10.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/EnergyConsumptionResURI		get			

6.11 Energy Usage

6.11.1 Introduction

This resource describes an energy usage query. The values in the start and stop time strings are encoded according to the rules defined in ISO 8601 and equate to time period over which the query applies. The energy consumption is the separately defined OIC resource.

6.11.2 Example URI

/EnergyUsageResURI

6.11.3 Resource Type

The resource type (rt) is defined as: oic.r.energy.usage.

6.11.4 RAML Definition

```
#%RAML 0.8
title: OICEnergyUsage
version: v1.1-20150805
traits:
 - interface :
     queryParameters:
       if:
         enum: ["oic.if.ll", "oic.if.b"]


/EnergyUsageResURI:
  description: |
    This resource describes an energy usage query.
    The values in the start and stop time strings are encoded according
    to the rules defined in ISO 8601 and equate to time period over which
    the query applies.
    The energy consumption is the separately defined OIC resource.

  is : ['interface']
  get:
    description: |
      Retrieves the energy usage information as a composite of consumption over time.

    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.energy.usage#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Energy Usage",
                "definitions": {
                  "oic.r.energy.usage": {
                    "type": "object",
                    "properties": {
                      "resources": {
                        "type": "array",
                        "minItems": 2,
                        "maxItems": 2,
                        "items": {
                          "$ref": "oic.web-link.json#"
                        }
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.energy.usage"}
                ],
                "required": ["resources"]
              }

            example: |
              {
                "rt": "oic.r.energy.usage",
                "id": "unique_example_id",
                "resources": [
                  {
                    "href": "/TimeIntervalResURI",
                    "rel":  "contains",
                    "rt":   "oic.r.time.period",
                    "if":   "oic.if.a"
                  },
                  {
                    "href": "/EnergyConsumptionResURI",
                    "rel":  "contains",
                    "rt":   "oic.r.energy.consumption",
                    "if":   "oic.if.s"
                  }
                ]
              }

```

6.11.5 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/EnergyUsageResURI		get			

6.12 Humidity

6.12.1 Introduction

This resource describes a sensed or desired humidity. The value humidity is an integer describing the percentage measured relative humidity. The value desiredHumidity is an integer showing the desired target relative humidity.

6.12.2 Example URI

/HumidityResURI

6.12.3 Resource Type

The resource type (rt) is defined as: oic.r.humidity.

6.12.4 RAML Definition

```
#%RAML 0.8
title: OICHumidity
version: v1.0-20150727
traits:
 - interface :
     queryParameters:
       if:
         enum: ["oic.if.a", "oic.if.s"]


/HumidityResURI:
```

```
description: |
  This resource describes a sensed or desired humidity.
  The value humidity is an integer describing the percentage measured relative humidity.
  The value desiredHumidity is an integer showing the desired target relative humidity.

is : ['interface']
get:
  description: |
    Retrieves the current (relative) humidity level.

  responses :
    200:
      body:
        application/json:
          schema: |
            {
              "id": "http://openinterconnect.org/schemas/oic.r.humidity#",
              "$schema": "http://json-schema.org/draft-04/schema#",
              "title": "Humidity",
              "definitions": {
                "oic.r.humidity": {
                  "type": "object",
                  "properties": {
                    "humidity":    {
                      "type": "integer",
                      "description": "ReadOnly, Current sensed value for Humidity"
                    },
                    "desiredHumidity":  {
                      "type": "integer",
                      "description": "Desired value for Humidity"
                    }
                  }
                }
              },
              "type": "object",
              "allOf": [
                {"$ref": "oic.core.json#/definitions/oic.core"},
                {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                {"$ref": "#/definitions/oic.r.humidity"}
              ],
              "required": ["humidity"]
            }

          example: |
            {
              "rt":               "oic.r.humidity",
              "id":               "unique_example_id",
              "humidity":         40,
              "desiredHumidity":  40
            }

post:
  description: |
    Sets the desired relative humidity level.

  body:
    application/json:
      schema: |
        {
          "id": "http://openinterconnect.org/schemas/oic.r.humidity#",
          "$schema": "http://json-schema.org/draft-04/schema#",
          "title": "Humidity",
          "definitions": {
            "oic.r.humidity": {
              "type": "object",
              "properties": {
                "desiredHumidity":  {
                  "type": "integer",
                  "description": "Desired value for Humidity"
                }
              }
            }
          },
          "type": "object",
          "allOf": [
            {"$ref": "oic.core.json#/definitions/oic.core"},
            {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
            {"$ref": "#/definitions/oic.r.humidity"}
          ]
        }

      example: |
        {
          "id":               "unique_example_id",
          "desiredHumidity" : 45
        }

  responses :
    200:
      description: |
        Indicates that the relative humidity level was changed.
        The new relative humidity level is provided in the response.

      body:
        application/json:
          schema: |
            {
              "id": "http://openinterconnect.org/schemas/oic.r.humidity#",
              "$schema": "http://json-schema.org/draft-04/schema#",
              "title": "Humidity",
              "definitions": {
                "oic.r.humidity": {
                  "type": "object",
                  "properties": {
                    "desiredHumidity":  {
                      "type": "integer",
                      "description": "Desired value for Humidity"
                    }
                  }
                }
              },
              "type": "object",
              "allOf": [
                {"$ref": "oic.core.json#/definitions/oic.core"},
                {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                {"$ref": "#/definitions/oic.r.humidity"}
              ]
            }

          example: |
            {
              "id":              "unique_example_id",
              "desiredHumidity": 45
            }
```

6.12.5 Property Definition

Property name	Value	Mandatory	Access	Description

	type		mode	
humidity	integer	yes	Read Only	Current Sensed Value For Humidity
desiredHumidity	integer		Read Write	Desired value for Humidity

6.12.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/HumidityResURI		get	post		

6.13 Ice Maker

6.13.1 Introduction

This resource describies an Ice Maker. The status is a string containing a value from the set of possible ice maker statuses. The possible statuses are defined by the enumeration [on, off, full] A status of 'on' means that the Ice Maker is operating. A status of 'off' means that the Ice Maker is not operating. A status of 'full' means that the ice collection bin is full (Ice Maker is operating).

6.13.2 Example URI

/IceMakerResURI

6.13.3 Resource Type

The resource type (rt) is defined as: oic.r.iceMaker.

6.13.4 RAML Definition

```
#%RAML 0.8
title: OICIceMaker
version: v1.0-20150727
traits:
 - interface :
     queryParameters:
       if:
         enum: ["oic.if.a"]

/IceMakerResURI:
  description: |
    This resource describies an Ice Maker.
    The status is a string containing a value from the set of possible ice maker statuses.
    The possible statuses are defined by the enumeration [on, off, full]
    A status of 'on' means that the Ice Maker is operating.
    A status of 'off' means that the Ice Maker is not operating.
    A status of 'full' means that the ice collection bin is full (Ice Maker is operating).

  is : ['interface']
  get:
    description: |
      Retrieves the current Ice Maker status.

    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.iceMaker#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Ice Maker",
                "definitions": {
                  "oic.r.iceMaker": {
                    "type": "object",
                    "properties": {
                      "status": {
                        "enum": ["on","off","full"],
                        "description": "Status of the Ice Maker"
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.iceMaker"}
                ],
                "required": ["status"]
              }

            example: |
              {
                "rt":     "oic.r.iceMaker",
                "id":     "unique_example_id",
                "status": "on"
              }

  post:
    description: |
      Sets the desired Ice Maker status.
      Only valid settings for status in a Post shall be [on,off].

    body:
      application/json:
        schema: |
          {
            "id": "http://openinterconnect.org/schemas/oic.r.iceMaker#",
            "$schema": "http://json-schema.org/draft-04/schema#",
            "title": "Ice Maker",
            "definitions": {
              "oic.r.iceMaker": {
                "type": "object",
                "properties": {
                  "status": {
                    "enum": ["on","off"],
                    "description": "Set the status of the Ice Maker"
                  }
                }
              }
            },
            "type": "object",
            "allOf": [
              {"$ref": "oic.core.json#/definitions/oic.core"},
              {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
              {"$ref": "#/definitions/oic.r.iceMaker"}
            ],
            "required": ["status"]
          }

        example: |
          {
            "id":     "unique_example_id",
            "status": "off"
          }

    responses :
      200:
        description: |
          Indicates that the Ice Maker status was changed.
          The new status  is provided in the response.

        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.iceMaker#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Ice Maker",
                "definitions": {
                  "oic.r.iceMaker": {
                    "type": "object",
                    "properties": {
                      "status": {
                        "enum": ["on","off"],
                        "description": "Set the status of the Ice Maker"
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.iceMaker"}
                ],
                "required": ["status"]
              }

            example: |
              {
                "id":     "unique_example_id",
                "status": "off"
              }

      403:
        description: |
          This response is generated by the OIC Server when the client sends:
            An update with an invalid property value for status.
          The server responds with the current resource representation.

        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.iceMaker#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Ice Maker",
                "definitions": {
                  "oic.r.iceMaker": {
                    "type": "object",
                    "properties": {
                      "status": {
                        "enum": ["on","off"],
                        "description": "Set the status of the Ice Maker"
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.iceMaker"}
                ],
                "required": ["status"]
              }

            example: |
              {
                "id":     "unique_example_id",
                "status": "off"
              }

```

6.13.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
status	enum	yes	Read Write	Status of the Ice Maker

6.13.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/IceMakerResURI		get	post		

6.14 Lock

6.14.1 Introduction

Resource describing a lock. For the type of lockState, the value 'Locked' indicates that the door is Locked. The value 'Unlocked' indicates that the door is Unlocked.

6.14.2 Example URI

/LockStatusResURI

6.14.3 Resource Type

The resource type (rt) is defined as: oic.r.lock.status.

6.14.4 RAML Definition

```
#%RAML 0.8
title: OICLock
version: v1.0-20150727
traits:
 - interface :
     queryParameters:
       if:
         enum: ["oic.if.a"]


/LockStatusResURI:
  description: |
    Resource describing a lock.
    For the type of lockState, the value 'Locked' indicates that the door is Locked.
    The value 'Unlocked' indicates that the door is Unlocked.

  is : ['interface']
  get:
    description: |
      Retrieves the state of the lock.

    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.lock.status#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Lock",
                "definitions": {
                  "oic.r.lock.status": {
                    "type": "object",
                    "properties": {
                      "lockState" : {
                        "type": "string",
                        "enum": ["Locked", "Unlocked"],
                        "description": "State of the lock."
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.lock.status"}
                ],
                "required": ["lockState"]
              }

            example: |
              {
                "rt":         "oic.r.lock.status",
                "id":         "unique_example_id",
                "lockState":  "Locked"
              }

  post:
    description: |
      Sets the current lock state.

    body:
      application/json:
        schema: |
          {
            "id": "http://openinterconnect.org/schemas/oic.r.lock.status#",
            "$schema": "http://json-schema.org/draft-04/schema#",
            "title": "Lock",
            "definitions": {
              "oic.r.lock.status": {
                "type": "object",
                "properties": {
                  "lockState" : {
                    "type": "string",
                    "enum": ["Locked", "Unlocked"],
                    "description": "State of the lock."
                  }
                }
              }
            },
            "type": "object",
            "allOf": [
              {"$ref": "oic.core.json#/definitions/oic.core"},
              {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
              {"$ref": "#/definitions/oic.r.lock.status"}
            ],
            "required": ["lockState"]
          }

        example: |
          {
            "id":         "unique_example_id",
            "lockState":  "Unlocked"
          }

    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.lock.status#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Lock",
                "definitions": {
                  "oic.r.lock.status": {
                    "type": "object",
                    "properties": {
                      "lockState" : {
                        "type": "string",
                        "enum": ["Locked", "Unlocked"],
                        "description": "State of the lock."
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.lock.status"}
                ],
                "required": ["lockState"]
              }

            example: |
              {
                "id":         "unique_example_id",
                "lockState":  "Unlocked"
              }

      403:
        description: |
          This response is generated by the OIC Server when the client sends:
            An update with an invalid property value for lockState.
          The server responds with the current resource representation.

        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.lock.status#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Lock",
                "definitions": {
                  "oic.r.lock.status": {
                    "type": "object",
                    "properties": {
                      "lockState" : {
                        "type": "string",
                        "enum": ["Locked", "Unlocked"],
                        "description": "State of the lock."
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.lock.status"}
                ],
                "required": ["lockState"]
              }

            example: |
              {
                "lockState": "Unlocked"
              }

```

6.14.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
lockState	string	yes	Read Write	State of the lock.

6.14.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/LockStatusResURI		get	post		

6.15 Lock Code

6.15.1 Introduction

Resource describing a lock code. The lockCodeList is an array of possible codes that may be associated with a lock. These are all presented as strings.

6.15.2 Example URI

/LockCodeResURI

6.15.3 Resource Type

The resource type (rt) is defined as: oic.r.lock.code.

6.15.4 RAML Definition

```
#%RAML 0.8
title: OICLockCode
version: v1.0-20150727
traits:
 - interface :
     queryParameters:
       if:
         enum: ["oic.if.a"]

/LockCodeResURI:
  description: |
    Resource describing a lock code.
    The lockCodeList is an array of possible codes that may be associated with a lock.
    These are all presented as strings.

  is : ['interface']
  get:
    description: |
      Retrieves the current lock code values.

    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.lock.code#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Lock Code",
                "definitions": {
                  "oic.r.lock.code": {
                    "type": "object",
                    "properties": {
                      "lockCodeList" : {
                        "type": "array",
                        "items": {
                          "type": "string",
                          "description": "Value for the lock code"
                        }
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.lock.code"}
                ],
                "required": ["lockCodeList"]
              }

            example: |
              {
                "rt":           "oic.r.lock.code",
                "id":           "unique_example_id",
                "lockCodeList": ["012345","112233"]
              }

  post:
    description: |
      Updates the current lock code values.

    body:
      application/json:
        schema: |
          {
            "id": "http://openinterconnect.org/schemas/oic.r.lock.code#",
            "$schema": "http://json-schema.org/draft-04/schema#",
            "title": "Lock Code",
            "definitions": {
              "oic.r.lock.code": {
                "type": "object",
                "properties": {
                  "lockCodeList" : {
                    "type": "array",
                    "items": {
                      "type": "string",
                      "description": "Value for the lock code"
                    }
                  }
                }
              }
            },
            "type": "object",
            "allOf": [
              {"$ref": "oic.core.json#/definitions/oic.core"},
              {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
              {"$ref": "#/definitions/oic.r.lock.code"}
            ],
            "required": ["lockCodeList"]
          }

        example: |
          {
            "id":           "unique_example_id",
            "lockCodeList": ["543210","332211"]
          }

    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.lock.code#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Lock Code",
                "definitions": {
                  "oic.r.lock.code": {
                    "type": "object",
                    "properties": {
                      "lockCodeList" : {
                        "type": "array",
                        "items": {
                          "type": "string",
                          "description": "Value for the lock code"
                        }
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.lock.code"}
                ],
                "required": ["lockCodeList"]
              }

            example: |
              {
                "id":           "unique_example_id",
                "lockCodeList": ["543210","332211"]
              }

```

6.15.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
lockCodeList	array	yes		

6.15.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/LockCodeResURI		get	post		

6.16 Mode

6.16.1 Introduction

This resource describes the modes of operation that a device can provide. The mode can be read or set. The supportedModes is a comma separated list of possible modes the device supports. The modes are a comma separated list of the currently active mode(s).

6.16.2 Example URI

/ModeResURI

6.16.3 Resource Type

The resource type (rt) is defined as: oic.r.mode.

6.16.4 RAML Definition

```
#%RAML 0.8
title: OICMode
version: v1.0-20150727
traits:
 - interface :
     queryParameters:
       if:
         enum: ["oic.if.a"]

/ModeResURI:
  description: |
    This resource describes the modes of operation that a device can provide.
    The mode can be read or set.
    The supportedModes is a comma separated list of possible modes the device supports.
    The modes are a comma separated list of the currently active mode(s).

  is : ['interface']
  get:
    description: |
      Retrieves the current mode.

    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.mode#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Mode",
                "definitions": {
                  "oic.r.mode": {
                    "type": "object",
                    "properties": {
                      "supportedModes": {
                        "type": "string",
                        "description": "ReadOnly, Comma separated list of possible modes the device
supports."
                      },
                      "modes":          {
                        "type": "string",
                        "description": "Comma separated list of the currently active mode(s)"
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.mode"}
                ],
                "required": ["supportedModes", "modes"]
              }

            example: |
              {
                "rt":             "oic.r.mode",
                "id":             "unique_example_id",
                "supportedModes": "home,away,quiet,sleep",
                "modes":          "quiet"
              }

  post:
    description: |
      Sets the desired mode.

    body:
      application/json:
        schema: |
          {
            "id": "http://openinterconnect.org/schemas/oic.r.mode#",
            "$schema": "http://json-schema.org/draft-04/schema#",
            "title": "Mode",
            "definitions": {
              "oic.r.mode": {
                "type": "object",
                "properties": {
                  "modes":          {
                    "type": "string",
                    "description": "Desired mode"
                  }
                }
              }
            },
            "type": "object",
            "allOf": [
              {"$ref": "oic.core.json#/definitions/oic.core"},
              {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
              {"$ref": "#/definitions/oic.r.mode"}
            ],
            "required": ["modes"]
          }

        example: |
          {
            "id":    "unique_example_id",
            "modes": "sleep"
          }

    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.mode#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Mode",
                "definitions": {
                  "oic.r.mode": {
                    "type": "object",
                    "properties": {
                      "modes":          {
                        "type": "string",
                        "description": "Desired mode"
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.mode"}
                ],
                "required": ["modes"]
              }

            example: |
              {
                "id":    "unique_example_id",
                "modes": "sleep"
              }

      403:
        description: |
          This response is generated by the OIC Server when the client sends:
            An update with an value for mode that is not found in supportedModes.
          The server responds with the current resource representation.

        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.mode#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Mode",
                "definitions": {
                  "oic.r.mode": {
                    "type": "object",
                    "properties": {
                      "supportedModes": {
                        "type": "string",
                        "description": "ReadOnly, Comma separated list of possible modes the device
supports."
                      },
                      "modes":          {
                        "type": "string",
                        "description": "Comma separated list of the currently active mode(s)"
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.mode"}
                ],
                "required": ["supportedModes", "modes"]
              }

            example: |
              {
                "id":               "unique_example_id",
                "supportedModes": "home,away,quiet,sleep",
                "modes":            "quiet"
              }

```

6.16.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
supportedModes	string	yes	Read Only	Comma Separated List Of Possible Modes The Device Supports.
modes	string	yes	Read Write	Comma separated list of the currently active mode(s)

6.16.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/ModeResURI		get	post		

6.17 Open Level

6.17.1 Introduction

This resource describes how open or ajar an entity such as a window, door, blind or shutter is. The openLevel can be read (acting as a sensor). The openLevel can also be set (acting as an actuator). The openLevel is device dependent across the range provided. If no range is provided then 0 to 100 is assumed. 0 means closed, 100 means fully open. If a range is provided the lower bound=closed, upper bound=open. The increment is the step between possible values; if not provide 1 is assumed.

6.17.2 Example URI

/OpenLevelResURI

6.17.3 Resource Type

The resource type (rt) is defined as: oic.r.openLevel.

6.17.4 RAML Definition

```
#%RAML 0.8
title: OICOpenLevel
version: v1.0-20150727
traits:
 - interface :
     queryParameters:
       if:
         enum: ["oic.if.a"]


/OpenLevelResURI:
  description: |
    This resource describes how open or ajar an entity such as a window, door, blind or shutter is.
    The openLevel can be read (acting as a sensor).
    The openLevel can also be set (acting as an actuator).
    The openLevel is device dependent across the range provided.
    If no range is provided then 0 to 100 is assumed.
    0 means closed, 100 means fully open.
    If a range is provided the lower bound=closed, upper bound=open.
    The increment is the step between possible values; if not provide 1 is assumed.

  is : ['interface']
  get:
    description: |
      Retrieves the current openLevel.

    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.openLevel#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Open Level",
                "definitions": {
                  "oic.r.openLevel": {
                    "type": "object",
                    "properties": {
                      "openLevel":      {
                        "type": "integer",
                        "description": "How open or ajar the entity is"
                      },
                      "increment":  {
                        "type": "integer",
                        "description": "ReadOnly, The step between possible values"
                      },
                      "range":      {
                        "type": "string",
                        "description": "ReadOnly, Lower bound=closed, Upper bound=open"
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.openLevel"}
                ],
                "required": ["openLevel"]
              }

            example: |
              {
                "rt":         "oic.r.openLevel",
                "id":         "unique_example_id",
                "openLevel":  50,
                "increment":  2,
                "range":      "0,100"
              }

  post:
    description: |
      Sets the desired openLevel.

    body:
      application/json:
        schema: |
          {
            "id": "http://openinterconnect.org/schemas/oic.r.openLevel#",
            "$schema": "http://json-schema.org/draft-04/schema#",
            "title": "Open Level",
            "definitions": {
              "oic.r.openLevel": {
                "type": "object",
                "properties": {
                  "openLevel":      {
```

```
                    "type": "integer",
                    "description": "How open or ajar the entity is"
                  },
                  "increment":  {
                    "type": "integer",
                    "description": "ReadOnly, The step between possible values"
                  },
                  "range":      {
                    "type": "string",
                    "description": "ReadOnly, Lower bound=closed, Upper bound=open"
                  }
                }
              }
            },
            "type": "object",
            "allOf": [
              {"$ref": "oic.core.json#/definitions/oic.core"},
              {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
              {"$ref": "#/definitions/oic.r.openLevel"}
            ],
            "required": ["openLevel"]
          }

        example: |
          {
            "id":         "unique_example_id",
            "openLevel":  0
          }

    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.openLevel#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Open Level",
                "definitions": {
                  "oic.r.openLevel": {
                    "type": "object",
                    "properties": {
                      "openLevel":      {
                        "type": "integer",
                        "description": "How open or ajar the entity is"
                      },
                      "increment":  {
                        "type": "integer",
                        "description": "ReadOnly, The step between possible values"
                      },
                      "range":      {
                        "type": "string",
                        "description": "ReadOnly, Lower bound=closed, Upper bound=open"
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.openLevel"}
                ],
                "required": ["openLevel"]
              }

            example: |
              {
                "id":         "unique_example_id",
                "openLevel":  0
              }

      403:
        description: |
          This response is generated by the OIC Server when the client sends:
            An update with an out of range property value for openLevel.
          The server responds with the current resource representation.

        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.openLevel#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Open Level",
                "definitions": {
                  "oic.r.openLevel": {
                    "type": "object",
                    "properties": {
                      "openLevel":      {
                        "type": "integer",
                        "description": "How open or ajar the entity is"
                      },
                      "increment":  {
                        "type": "integer",
                        "description": "ReadOnly, The step between possible values"
                      },
                      "range":      {
                        "type": "string",
                        "description": "ReadOnly, Lower bound=closed, Upper bound=open"
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.openLevel"}
                ],
                "required": ["openLevel"]
              }

            example: |
              {
                "id":         "unique_example_id",
                "openLevel":  50,
                "increment":  2,
                "range":      "0,100"
              }
```

6.17.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
openLevel	integer	yes	Read Write	How open or ajar the entity is
increment	integer		Read Only	The Step Between Possible Values
range	string		Read Only	Lower Bound=Closed, Upper Bound=Open

6.17.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/OpenLevelResURI		get	post		

6.18 Operational State

6.18.1 Introduction

This resource describes the operational and job states on a device. The states can be read or set, setting indicates a desired state. A device may reject an attempt to set a state that would result in adverse operational characteristics. The machineStates is a comma separated list of the possible operational states. The currentMachineState is the current state of operation of the device. The jobStates is a comma separated list of the possible job states. The currentJobState is the currently active jobState. The runningTime is the ISO8601 encoded elapsed time in the current operational state. The remainingTime is the ISO8601 encoded time till completion of the current operational state. The progressPercentage is the percentage completeness of the current jobState.

6.18.2 Example URI

/OperationalStateResURI

6.18.3 Resource Type

The resource type (rt) is defined as: oic.r.operational.state.

6.18.4 RAML Definition

```
#%RAML 0.8
title: OICOperation
version: v1.0-20150805
traits:
 - interface :
     queryParameters:
       if:
         enum: ["oic.if.a"]

/OperationalStateResURI:
  description: |
    This resource describes the operational and job states on a device.
    The states can be read or set, setting indicates a desired state.
    A device may reject an attempt to set a state that would result
    in adverse operational characteristics.
    The machineStates is a comma separated list of the possible operational states.
    The currentMachineState is the current state of operation of the device.
    The jobStates is a comma separated list of the possible job states.
    The currentJobState is the currently active jobState.
    The runningTime is the ISO8601 encoded elapsed time in the current operational state.
    The remainingTime is the ISO8601 encoded time till completion of the current operational state.
    The progressPercentage is the percentage completeness of the current jobState.

  is : ['interface']
  get:
    description: |
      Retrieves the current operational and job states.

    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.operational.state#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Operational State",
                "definitions": {
                  "oic.r.operational.state": {
                    "type": "object",
                    "properties": {
                      "machineStates":        {
                        "type": "string",
                        "description": "ReadOnly, Comma separated list of the possible operational
states."
                      },
                      "currentMachineState":  {
                        "type": "string",
                        "description": "Current state of operation of the device."
                      },
                      "jobStates":            {
                        "type": "string",
                        "description": "ReadOnly, Comma separate list of the possible job states."
                      },
                      "currentJobState":      {
                        "type": "string",
                        "description": "Currently active jobState"
                      },
                      "runningTime":          {
                        "type": "string",
                        "description": "ReadOnly, Elapsed time in the current operational state"
                      },
                      "remainingTime":        {
                        "type": "string",
                        "description": "ReadOnly, Time till completion of the current operational
state"
                      },
                      "progressPercentage":   {
                        "type": "integer",
                        "description": "ReadOnly, Percentage completeness of the current jobState"
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.operational.state"}
                ],
                "required": ["machineStates", "currentMachineState"]
              }

            example: |
              {
                "rt":                   "oic.r.operational.state",
                "id":                   "unique_example_id",
                "machineStates":        "pause, standby, ready, active",
                "currentMachineState":  "active",
                "jobStates":            "pre-wash, wash, rinse, spin, dry, air-dry, wrinkle-
prevent",
                "currentJobState":      "rinse",
                "runningTime":          "PT15M20S",
                "remainingTime":        "PT10M40S",
                "progressPercentage":    75
              }

  post:
    description: |
      Sets the desired operational or job state.

```

```
body:
  application/json:
    schema: |
      {
        "id": "http://openinterconnect.org/schemas/oic.r.operational.state#",
        "$schema": "http://json-schema.org/draft-04/schema#",
        "title": "Operational State",
        "definitions": {
          "oic.r.operational.state": {
            "type": "object",
            "properties": {
              "currentMachineState":  {
                "type": "string",
                "description": "Current state of operation of the device."
              },
              "currentJobState":      {
                "type": "string",
                "description": "Currently active jobState"
              }
            }
          }
        },
        "type": "object",
        "allOf": [
          {"$ref": "oic.core.json#/definitions/oic.core"},
          {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
          {"$ref": "#/definitions/oic.r.operational.state"}
        ]
      }

    example: |
      {
        "id":                   "unique_example_id",
        "currentMachineState":  "pause",
        "currentJobState":      "wash"
      }

responses :
  200:
    body:
      application/json:
        schema: |
          {
            "id": "http://openinterconnect.org/schemas/oic.r.operational.state#",
            "$schema": "http://json-schema.org/draft-04/schema#",
            "title": "Operational State",
            "definitions": {
              "oic.r.operational.state": {
                "type": "object",
                "properties": {
                  "currentMachineState":  {
                    "type": "string",
                    "description": "Current state of operation of the device."
                  },
                  "currentJobState":      {
                    "type": "string",
                    "description": "Currently active jobState"
                  }
                }
              }
            },
            "type": "object",
            "allOf": [
              {"$ref": "oic.core.json#/definitions/oic.core"},
              {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
              {"$ref": "#/definitions/oic.r.operational.state"}
```

```
                ]
              }

            example: |
              {
                "id":                   "unique_example_id",
                "currentMachineState":  "pause",
                "currentJobState":      "wash"
              }

      403:
        description: |
          This response is generated by the OIC Server when the client sends:
            An update with an value for currentMachineState that is not found in machineStates.
            An update with an value for currentJobState that is not found in jobStates.
          The server responds with the current resource representation.

        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.operational.state#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Operational State",
                "definitions": {
                  "oic.r.operational.state": {
                    "type": "object",
                    "properties": {
                      "machineStates":        {
                        "type": "string",
                        "description": "ReadOnly, Comma separated list of the possible operational 
states."
                      },
                      "currentMachineState":  {
                        "type": "string",
                        "description": "Current state of operation of the device."
                      },
                      "jobStates":            {
                        "type": "string",
                        "description": "ReadOnly, Comma separate list of the possible job states."
                      },
                      "currentJobState":      {
                        "type": "string",
                        "description": "Currently active jobState"
                      },
                      "runningTime":          {
                        "type": "string",
                        "description": "ReadOnly, Elapsed time in the current operational state"
                      },
                      "remainingTime":        {
                        "type": "string",
                        "description": "ReadOnly, Time till completion of the current operational 
state"
                      },
                      "progressPercentage":   {
                        "type": "integer",
                        "description": "ReadOnly, Percentage completeness of the current jobState"
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.operational.state"}
                ],
                "required": ["machineStates", "currentMachineState"]
              }

            example: |
              {
                "id":                     "unique_example_id",
                "machineStates":          "pause, standby, ready, active",
                "currentMachineState":    "active",
                "jobStates":              "pre-wash, wash, rinse, spin, dry, air-dry, wrinkle-
prevent",
                "currentJobState":        "rinse",
                "runningTime":            "PT15M20S",
                "remainingTime":          "PT10M40S",
                "progressPercentage":     75
              }
```

6.18.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
machineStates	string	yes	Read Only	Comma Separated List Of The Possible Operational States.
currentMachineState	string	yes	Read Write	Current state of operation of the device.
jobStates	string		Read Only	Comma Separate List Of The Possible Job States.
currentJobState	string		Read Write	Currently active jobState
runningTime	string		Read Only	Elapsed Time In The Current Operational State
remainingTime	string		Read Only	Time Till Completion Of The Current Operational State
progressPercentage	integer		Read Only	Percentage Completeness Of The Current Jobstate

6.18.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/OperationalStateResURI		get	post		

6.19 Ramp Time

6.19.1 Introduction

This resource that describes the Ramp Time of a dimming function. This specifies the actual speed of changing between 2 dimming values. Time is specified in milliseconds [ms]. If the range value is not specified then the maximum value is 100 ms. The RampTime of 0ms indicates the minimal delay possible by the implementation.

6.19.2 Example URI

/RampTimeResURI

6.19.3 Resource Type

The resource type (rt) is defined as: oic.r.light.rampTime.

6.19.4 RAML Definition

```
#%RAML 0.8

title: OICRampTime
version: v1.0-20150727

traits:
 - interface :
     queryParameters:
       if:
         enum: ["oic.if.a"]

/RampTimeResURI:
  description: |
    This resource that describes the Ramp Time of a dimming function.
    This specifies the actual speed of changing between 2 dimming values.
    Time is specified in milliseconds [ms].
    If the range value is not specified then the maximum value is 100 ms.
    The RampTime of 0ms indicates the minimal delay possible by the implementation.

  is : ['interface']
  get:
    description: |
      Retrieves the current RampTime.

    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.light.rampTime#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Ramp Time",
                "definitions": {
                  "oic.r.light.rampTime": {
                    "type": "object",
                    "properties": {
                      "rampTime": {
                        "type": "integer",
                        "description": "Actual speed of changing between 2 dimming values"
                      },
                      "range": {
                        "type": "string",
                        "description": "ReadOnly, Min and Max of possible values"
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.light.rampTime"}
                ],
                "required": ["rampTime"]
              }

            example: |
              {
                "rt":       "oic.r.light.rampTime",
                "id":       "unique_example_id",
                "rampTime": 0,
                "range":    "0,100"
              }

  post:
    description: |
      Sets the current RampTime.

    body:
```

```
  application/json:
    schema: |
      {
        "id": "http://openinterconnect.org/schemas/oic.r.light.rampTime#",
        "$schema": "http://json-schema.org/draft-04/schema#",
        "title": "Ramp Time",
        "definitions": {
          "oic.r.light.rampTime": {
            "type": "object",
            "properties": {
              "rampTime": {
                "type": "integer",
                "description": "Actual speed of changing between 2 dimming values"
              },
              "range": {
                "type": "string",
                "description": "ReadOnly, Min and Max of possible values"
              }
            }
          }
        },
        "type": "object",
        "allOf": [
          {"$ref": "oic.core.json#/definitions/oic.core"},
          {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
          {"$ref": "#/definitions/oic.r.light.rampTime"}
        ],
        "required": ["rampTime"]
      }

    example: |
      {
        "id":       "unique_example_id",
        "rampTime": 50
      }

responses :
  200:
    body:
      application/json:
        schema: |
          {
            "id": "http://openinterconnect.org/schemas/oic.r.light.rampTime#",
            "$schema": "http://json-schema.org/draft-04/schema#",
            "title": "Ramp Time",
            "definitions": {
              "oic.r.light.rampTime": {
                "type": "object",
                "properties": {
                  "rampTime": {
                    "type": "integer",
                    "description": "Actual speed of changing between 2 dimming values"
                  },
                  "range": {
                    "type": "string",
                    "description": "ReadOnly, Min and Max of possible values"
                  }
                }
              }
            },
            "type": "object",
            "allOf": [
              {"$ref": "oic.core.json#/definitions/oic.core"},
              {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
              {"$ref": "#/definitions/oic.r.light.rampTime"}
            ],
            "required": ["rampTime"]
          }

        example: |
          {
            "id":       "unique_example_id",
            "rampTime": 50
          }

  403:
    description: |
      This response is generated by the OIC Server when the client sends:
        An update with an out of range property value for rampTime.
      The server responds with the current resource representation.

    body:
      application/json:
        schema: |
          {
            "id": "http://openinterconnect.org/schemas/oic.r.light.rampTime#",
            "$schema": "http://json-schema.org/draft-04/schema#",
            "title": "Ramp Time",
            "definitions": {
              "oic.r.light.rampTime": {
                "type": "object",
                "properties": {
                  "rampTime": {
                    "type": "integer",
                    "description": "Actual speed of changing between 2 dimming values"
                  },
                  "range": {
                    "type": "string",
                    "description": "ReadOnly, Min and Max of possible values"
                  }
                }
              }
            },
            "type": "object",
            "allOf": [
              {"$ref": "oic.core.json#/definitions/oic.core"},
              {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
              {"$ref": "#/definitions/oic.r.light.rampTime"}
            ],
            "required": ["rampTime"]
          }

        example: |
          {
            "id":       "unique_example_id",
            "rampTime": 40
          }
```

6.19.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
rampTime	integer	yes	Read Write	Actual speed of changing between 2 dimming values
range	string		Read Only	Min And Max Of Possible Values

6.19.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/RampTimeResURI		get	post		

6.20 Refrigeration

6.20.1 Introduction

This resource describes a refrigeration function. This is not a Refrigerator device. The filter state is a read-only value providing the percentage life time remaining for the water filter. RapidFreeze is a boolean that controls the rapid freeze capability if present. RapidCool is a boolean that controls the rapid cool capability if present. Defrost is a boolean that controls the defrost cycle if present.

6.20.2 Example URI

/RefrigerationResURI

6.20.3 Resource Type

The resource type (rt) is defined as: oic.r.refrigeration.

6.20.4 RAML Definition

```
#%RAML 0.8
title: OICRefrigeration
version: v1.0-20150727
traits:
 - interface :
     queryParameters:
       if:
         enum: ["oic.if.a"]


/RefrigerationResURI:
  description: |
    This resource describes a refrigeration function.
    This is not a Refrigerator device.
    The filter state is a read-only value providing the percentage life time remaining for the
water filter.
    RapidFreeze is a boolean that controls the rapid freeze capability if present.
    RapidCool is a boolean that controls the rapid cool capability if present.
    Defrost is a boolean that controls the defrost cycle if present.

  is : ['interface']
  get:
    description: |
      Retrieves the current Refrigeration function status.

    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.refrigeration#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Refrigeration",
                "definitions": {
                  "oic.r.refrigeration": {
                    "type": "object",
                    "properties": {
                      "filter":       {
                        "type": "integer",
                        "description": "ReadOnly, Percentage life time remaining for the water
filter"
                      },
                      "supportedFunctions": {
                        "type": "string",
                        "description": "ReadOnly, comma separated value set of supported functions"}
                      },
                      "activeFunction": {
                        "type": "string",
                        "description": "Comma separated value set of active functions from the set
of supportedFunctions"}
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.refrigeration"}
                ],
                "required": ["supportedFunctions", "activeFunction"]
              }

            example: |
              {
                "rt":           "oic.r.refrigeration",
                "id":           "unique_example_id",
                "filter":       75,
                "rapidFreeze":  false,
                "rapidCool":    false,
                "defrost":      true
              }

  post:
    description: |
      Activates the desired Refrigeration functions.
      Suported values are rapidFreeze, rapidCool and defrost.

    body:
      application/json:
        schema: |
          {
            "id": "http://openinterconnect.org/schemas/oic.r.refrigeration#",
            "$schema": "http://json-schema.org/draft-04/schema#",
            "title": "Refrigeration",
            "definitions": {
              "oic.r.refrigeration": {
                "type": "object",
                "properties": {
                  "rapidFreeze":  {
                    "type": "boolean",
                    "description": "Indicates whether the unit has a rapid freeze capability
active."
                  },
                  "rapidCool":    {
                    "type": "boolean",
                    "description": "Indicates whether the unit has a rapid cool capability active"
                  },
                  "defrost":      {
                    "type": "boolean",
                    "description": "Indicates whether a defrost cycle is currently active"
                  }
                }
              }
            },
            "type": "object",
            "allOf": [
              {"$ref": "oic.core.json#/definitions/oic.core"},
              {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
              {"$ref": "#/definitions/oic.r.refrigeration"}
            ],
            "required": ["defrost"]
          }

        example: |
          {
            "id":       "unique_example_id",
            "defrost":  false
          }

    responses :
      200:
        description: |
          Indicates that the Refrigeration function was changed.
          The new status  is provided in the response.

        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.refrigeration#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Refrigeration",
                "definitions": {
                  "oic.r.refrigeration": {
                    "type": "object",
                    "properties": {
                      "rapidFreeze":  {
                        "type": "boolean",
                        "description": "Indicates whether the unit has a rapid freeze capability
active."
                      },
                      "rapidCool":    {
                        "type": "boolean",
                        "description": "Indicates whether the unit has a rapid cool capability
active"
                      },
                      "defrost":      {
                        "type": "boolean",
                        "description": "Indicates whether a defrost cycle is currently active"
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.refrigeration"}
                ],
                "required": ["defrost"]
              }

            example: |
              {
                "id":       "unique_example_id",
                "defrost":  false
              }
```

6.20.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
filter	integer		Read	Percentage Life Time Remaining For

			Only	The Water Filter
supportedFunctions	string	yes	Read Only	Comma Separated Value Set Of Supported Functions
activeFunction	string	yes	Read Write	Comma separated value set of active functions from the set of supportedFunctions

6.20.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/RefrigerationResURI		get	post		

6.21 Temperature

6.21.1 Introduction

This resource describes a sensed or actuated Temperature value. The temperature describes the current value measured. If the units attribute is missing the default is Celsius [C]. The range is a comma separated min,max values for this temperature on this device. If no range is provided the default is +/- MAXINT.

6.21.2 Example URI

/TemperatureResURI

6.21.3 Resource Type

The resource type (rt) is defined as: oic.r.temperature.

6.21.4 RAML Definition

```
#%RAML 0.8
title: OICTemperature
version: v1.0-20150727
traits:
 - interface :
     queryParameters:
       if:
         enum: ["oic.if.a", "oic.if.s"]

/TemperatureResURI:
  description: |
    This resource describes a sensed or actuated Temperature value.
    The temperature describes the current value measured.
    If the units attribute is missing the default is Celsius [C].
    The range is a comma separated min,max values for this temperature on this device.
    If no range is provided the default is +/- MAXINT.

  is : ['interface']
  get:
    description: |
      Retrieves the current temperature value.
      A client can specify the units for the requested temperature by use of a query parameter.

    queryParameters:
      units:
        enum: CFK
    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.temperature#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Temperature",
                "definitions": {
                  "oic.r.temperature": {
                    "type": "object",
                    "properties": {
                      "temperature":  {
                        "type": "number",
                        "description": "Current temperature setting or measurement"
                      },
                      "units":  {
                        "enum": ["C","F","K"],
                        "description": "ReadOnly, Units for the temperature value"
                      },
                      "range":  {
                        "type": "string",
                        "description": "ReadOnly, Comma separated min,max values for this
temperature on this device"
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.temperature"}
                ],
                "required": ["temperature"]
              }

            example: |
              {
                "rt":           "oic.r.temperature",
                "id":           "unique_example_id",
                "temperature":  20.0,
                "units":        "C",
                "range":        "0,100"
              }

      403:
        description: |
          This response is generated by the OIC Server when the client sends:
            A retrieve with q queryParameter indicating a unit that the server does not support.
          The server responds with the units enumeration illustrating the error.

        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.temperature#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Temperature",
                "definitions": {
                  "oic.r.temperature": {
                    "type": "object",
                    "properties": {
                      "units":  {
                        "enum": ["C","F","K"],
                        "description": "ReadOnly, Units for the temperature value"
                      },
                      "range":  {
                        "type": "string",
                        "description": "ReadOnly, Comma separated min,max values for this
temperature on this device"
```

```
           }
          }
         }
        },
        "type": "object",
        "allOf": [
         {"$ref": "oic.core.json#/definitions/oic.core"},
         {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
         {"$ref": "#/definitions/oic.r.temperature"}
        ]
       }

      example: |
       {
        "id":     "unique_example_id",
        "units":  "C"
       }

 post:
  description: |
   Sets the desired temperature value.
   If the units are omitted the current value for units known by the server is used.

  body:
   application/json:
    schema: |
     {
      "id": "http://openinterconnect.org/schemas/oic.r.temperature#",
      "$schema": "http://json-schema.org/draft-04/schema#",
      "title": "Temperature",
      "definitions": {
       "oic.r.temperature": {
        "type": "object",
        "properties": {
         "temperature":  {
          "type": "number",
          "description": "Current temperature setting or measurement"
         },
         "units":  {
          "enum": ["C","F","K"],
          "description": "ReadOnly, Units for the temperature value"
         },
         "range":  {
          "type": "string",
          "description": "ReadOnly, Comma separated min,max values for this temperature
on this device"
         }
        }
       }
      },
      "type": "object",
      "allOf": [
       {"$ref": "oic.core.json#/definitions/oic.core"},
       {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
       {"$ref": "#/definitions/oic.r.temperature"}
      ],
      "required": ["temperature"]
     }

    example: |
     {
      "id":           "unique_example_id",
      "temperature":  18
     }
    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.temperature#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Temperature",
                "definitions": {
                  "oic.r.temperature": {
                    "type": "object",
                    "properties": {
                      "temperature":  {
                        "type": "number",
                        "description": "Current temperature setting or measurement"
                      },
                      "units":  {
                        "enum": ["C","F","K"],
                        "description": "ReadOnly, Units for the temperature value"
                      },
                      "range":  {
                        "type": "string",
                        "description": "ReadOnly, Comma separated min,max values for this
temperature on this device"
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.temperature"}
                ],
                "required": ["temperature"]
              }

            example: |
              {
                "id":          "unique_example_id",
                "temperature": 18
              }

      403:
        description: |
          This response is generated by the OIC Server when the client sends:
            An update with an out of range property value for temperature.
          The server responds with the range property illustrating the error.

        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.temperature#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Temperature",
                "definitions": {
                  "oic.r.temperature": {
                    "type": "object",
                    "properties": {
                      "units":  {
                        "enum": ["C","F","K"],
                        "description": "ReadOnly, Units for the temperature value"
                      },
           "range":  {
            "type": "string",
            "description": "ReadOnly, Comma separated min,max values for this
temperature on this device"
           }
          }
         }
        },
        "type": "object",
        "allOf": [
         {"$ref": "oic.core.json#/definitions/oic.core"},
         {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
         {"$ref": "#/definitions/oic.r.temperature"}
        ]
       }

      example: |
       {
        "id":     "unique_example_id",
        "range":  "0,100"
       }

```

6.21.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
temperature	number	yes	Read Write	Current temperature setting or measurement
units	enum		Read Only	Units For The Temperature Value
range	string		Read Only	Comma Separated Min,Max Values For This Temperature On This Device

6.21.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/TemperatureResURI		get	post		

6.22 Time Period

6.22.1 Introduction

This resource describes the time period over which any additionally provided information is derived or bounded. The start and stop times are ISO8601 encoded strings.

6.22.2 Example URI

/TimePeriodResURI

6.22.3 Resource Type

The resource type (rt) is defined as: oic.r.time.period.

6.22.4 RAML Definition

```
#%RAML 0.8

title: OICTimePeriod
version: v1.0-20150727

traits:
 - interface :
   queryParameters:

    if:
     enum: ["oic.if.a"]


/TimePeriodResURI:

 description: |
    This resource describes the time period over which any additionally provided
    information is derived or bounded.
    The start and stop times are ISO8601 encoded strings.

  is : ['interface']
  get:
    description: |
      Defines a time period for information retrieval, action or other behaviour.

    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.time.period#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Time Period",
                "definitions": {
                  "oic.r.time.period": {
                    "type": "object",
                    "properties": {
                      "startTime":{
                        "type": "string",
                        "description": "Start time for the time period"
                      },
                      "stopTime": {
                        "type": "string",
                        "description": "Stop time for the time period"
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.time.period"}
                ],
                "required": ["startTime"]
              }

            example: |
              {
                "rt":       "oic.r.time.period",
                "id":       "unique_example_id",
                "startTime":"2015-01-09T14:30Z",
                "stopTime": "2015-01-09T14:45Z"
              }

  post:
    description: |
      Sets or updates a time period for information retrieval, action or other behavior.

    body:
      application/json:
        schema: |
          {
            "id": "http://openinterconnect.org/schemas/oic.r.time.period#",
            "$schema": "http://json-schema.org/draft-04/schema#",
            "title": "Time Period",
            "definitions": {
              "oic.r.time.period": {
                "type": "object",
                "properties": {
                  "startTime":{
                    "type": "string",
                    "description": "Start time for the time period"
                  },
                  "stopTime": {
                    "type": "string",
                    "description": "Stop time for the time period"
                  }
                }
              }
            },
            "type": "object",
            "allOf": [
              {"$ref": "oic.core.json#/definitions/oic.core"},
              {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
              {"$ref": "#/definitions/oic.r.time.period"}
            ],
            "required": ["startTime"]
          }

        example: |
          {
            "id":       "unique_example_id",
            "startTime":"2015-01-09T14:30Z",
            "stopTime": "2015-01-09T14:45Z"
          }

    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.time.period#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Time Period",
                "definitions": {
                  "oic.r.time.period": {
                    "type": "object",
                    "properties": {
                      "startTime":{
                        "type": "string",
                        "description": "Start time for the time period"
                      },
                      "stopTime": {
                        "type": "string",
                        "description": "Stop time for the time period"
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.time.period"}
                ],
                "required": ["startTime"]
              }

            example: |
              {
                "id":       "unique_example_id",
                "startTime":"2015-01-09T14:30Z",
                "stopTime": "2015-01-09T14:45Z"
              }

```

6.22.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
startTime	string	yes	Read Write	Start time for the time period
stopTime	string		Read Write	Stop time for the time period

6.22.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/TimePeriodResURI		get	post		

6.23 Activity Count

6.23.1 Introduction

This resource specifies an activity count. The resource can be readonly (oic.if.s interface) in which instance it represents a count. The resource can be readwrite (oic.if.a interface) in which instance it represents a goal or target for a count. The count property is an integer representing either the current count or goal value.

6.23.2 Example URI

/ActivityCountResURI

6.23.3 Resource Type

The resource type (rt) is defined as: oic.r.sensor.activity.count.

6.23.4 RAML Definition

```
#%RAML 0.8
title: OICActivityCount
version: v1.0-20150727
traits:
 - interface :
     queryParameters:
       if:
         enum: ["oic.if.s", "oic.if.a"]

/ActivityCountResURI:
  description: |
    This resource specifies an activity count.
    The resource can be readonly (oic.if.s interface) in which instance it represents a count.
    The resource can be readwrite (oic.if.a interface) in which instance it represents a goal or
target for a count.
    The count property is an integer representing either the current count or goal value.

  is : ['interface']
  get:
    description: |
      Retrieves the current activity count.

    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.sensor.activity.count.json#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Activity Count Sensor",
                "definitions": {
                  "oic.r.sensor.activity.count": {
                    "properties": {
                      "count": {
                        "type": "integer",
                        "description": "Current or Target count."
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.sensor.activity.count"}
                ],
                "required": ["count"]
              }

            example: |
              {
                "rt":     "oic.r.sensor.activity.count",
                "id":     "unique_example_id",
                "count":  2500
              }

  post:
    description: |
      Sets the count target

    body:
      application/json:
        schema: |
          {
            "id": "http://openinterconnect.org/schemas/oic.r.sensor.activity.count.json#",
            "$schema": "http://json-schema.org/draft-04/schema#",
            "title": "Activity Count Sensor",
            "definitions": {
              "oic.r.sensor.activity.count": {
                "properties": {
                  "count": {
                    "type": "integer",
                    "description": "Current or Target count."
                  }
                }
              }
            },
            "type": "object",
            "allOf": [
              {"$ref": "oic.core.json#/definitions/oic.core"},
              {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
              {"$ref": "#/definitions/oic.r.sensor.activity.count"}
            ],
            "required": ["count"]
          }

        example: |
          {
            "id":     "unique_example_id",
            "count":  5000
          }

    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.sensor.activity.count.json#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Activity Count Sensor",
                "definitions": {
                  "oic.r.sensor.activity.count": {
                    "properties": {
                      "count": {
                        "type": "integer",
                        "description": "Current or Target count."
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.sensor.activity.count"}
                ],
                "required": ["count"]
              }

            example: |
              {
                "id":     "unique_example_id",
                "count":  5000
              }

```

6.23.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
count	integer	yes	Read Write	Current or Target count.

6.23.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/ActivityCountResURI		get	post		

6.24 Atmospheric Pressure Sensor

6.24.1 Introduction

This resource provides a measurement of Mean Sea Level Pressure experienced at the measuring point expressed in millibars. The value is float which describes the atmospheric pressure in millibars.

6.24.2 Example URI

/AtmosphericPressureResURI

6.24.3 Resource Type

The resource type (rt) is defined as: oic.r.sensor.atmosphericPressure.

6.24.4 RAML Definition

```
#%RAML 0.8
title: OICAtmosphericPressureSensor
version: v1.0-20150727
traits:
 - interface :
     queryParameters:
       if:
         enum: ["oic.if.s"]

/AtmosphericPressureResURI:
  description: |
    This resource provides a measurement of Mean Sea Level Pressure experienced at the measuring
point expressed in millibars.
    The value is float which describes the atmospheric pressure in millibars.

  is : ['interface']
  get:
    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.sensor.atmosphericPressure.json#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Atmospheric Pressure Sensor",
                "definitions": {
                  "oic.r.sensor.atmosphericPressure": {
                    "properties": {
                      "atmosphericPressure": {
                        "type": "number",
                        "description": "ReadOnly, Current atmospheric pressure in mbar."
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.sensor.atmosphericPressure"}
                ],
                "required": ["atmosphericPressure"]
              }

            example: |
              {
                "rt":                   "oic.r.sensor.atmosphericPressure",
                "id":                   "unique_example_id",
                "atmosphericPressure":  1000
              }
```

6.24.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
atmosphericPressure	number	yes	Read Only	Current Atmospheric Pressure In Mbar.

6.24.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/AtmosphericPressureResURI		get			

6.25 Audio Controls

6.25.1 Introduction

This resource defines basic audio control functions. The volume is an integer containing a percentage [0,100]. A volume of 0 (zero) means no sound produced. A volume of 100 means maximum sound production. The mute control is implemented as a boolean. A mute value of true means that the device is muted (no audio). A mute value of false means that the device is not muted (audio).

6.25.2 Example URI

/AudioResURI

6.25.3 Resource Type

The resource type (rt) is defined as: oic.r.audio.

6.25.4 RAML Definition

```
#%RAML 0.8

title: OICAudio
version: v1.0-20150727

traits:
 - interface :
     queryParameters:
       if:
         enum: ["oic.if.a"]


/AudioResURI:
  description: |
    This resource defines basic audio control functions.
    The volume is an integer containing a percentage [0,100].
    A volume of 0 (zero) means no sound produced.
    A volume of 100 means maximum sound production.
    The mute control is implemented as a boolean.
    A mute value of true means that the device is muted (no audio).
    A mute value of false means that the device is not muted (audio).

  is : ['interface']
  get:
    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.audio#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "definitions": {
                  "oic.r.audio": {
                    "type": "object",
                    "properties": {
                      "volume":   {
                        "type": "integer",
                        "description": "Volume setting of an audio rendering device."
                      },
                      "mute":    {
                        "type": "boolean",
                        "description": "Mute setting of an audio rendering device"
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.audio"}
                ],
                "required": ["volume","mute"]
            }

          example: |
            {
              "rt":     "oic.r.audio",
              "id":     "unique_example_id",
              "volume": 50,
              "mute":   false
            }

  post:
    body:
      application/json:
        schema: |
          {
            "id": "http://openinterconnect.org/schemas/oic.r.audio#",
            "$schema": "http://json-schema.org/draft-04/schema#",
            "definitions": {
              "oic.r.audio": {
                "type": "object",
                "properties": {
                  "volume":   {
                    "type": "integer",
                    "description": "Volume setting of an audio rendering device."
                  },
                  "mute":     {
                    "type": "boolean",
                    "description": "Mute setting of an audio rendering device"
                  }
                }
              }
            },
            "type": "object",
            "allOf": [
              {"$ref": "oic.core.json#/definitions/oic.core"},
              {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
              {"$ref": "#/definitions/oic.r.audio"}
            ],
            "required": ["volume","mute"]
          }

        example: |
          {
            "id":     "unique_example_id",
            "volume": 75,
            "mute":   false
          }

    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.audio#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "definitions": {
                  "oic.r.audio": {
                    "type": "object",
                    "properties": {
                      "volume":   {
                        "type": "integer",
                        "description": "Volume setting of an audio rendering device."
                      },
                      "mute":    {
                        "type": "boolean",
                        "description": "Mute setting of an audio rendering device"
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.audio"}
                ],
                "required": ["volume","mute"]
              }

            example: |
              {
                "id":     "unique_example_id",
                "volume": 75,
                "mute":   false
              }

```

6.25.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
volume	integer	yes	Read Write	Volume setting of an audio rendering device.
mute	boolean	yes	Read Write	Mute setting of an audio rendering device

6.25.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/AudioResURI		get	post		

6.26 Auto Focus

6.26.1 Introduction

This resource describes an auto focus on/off feature (on/off). The value is a boolean. An AutoFocus value of 'true' means that the switch is on. An AutoFocus value of 'false' means that the switch is off. Note that when PTZ is used the autofocus works only the selected area.

6.26.2 Example URI

/AutoFocusResURI

6.26.3 Resource Type

The resource type (rt) is defined as: oic.r.autofocus.

6.26.4 RAML Definition

```
#%RAML 0.8
title: OICAutoFocus
version: v1.0-20150727
traits:
 - interface :
     queryParameters:
       if:
         enum: ["oic.if.a"]


/AutoFocusResURI:
  description: |
    This resource describes an auto focus on/off feature (on/off).
    The value is a boolean.
    An AutoFocus value of 'true' means that the switch is on.
    An AutoFocus value of 'false' means that the switch is off.
    Note that when PTZ is used the autofocus works only the selected area.

  is : ['interface']
  get:
    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.autofocus#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Auto Focus",
                "definitions": {
                  "oic.r.autofocus": {
                    "type": "object",
                    "properties": {
                      "autoFocus": {
                        "type": "boolean",
                        "description": "Status of the Auto Focus"
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.autofocus"}
                ],
                "required": [ "autoFocus" ]
              }

            example: |
              {
                "rt":         "oic.r.autofocus",
                "id":         "unique_example_id",
                "autoFocus":  false
              }

  post:
    body:
      application/json:
        schema: |
          {
            "id": "http://openinterconnect.org/schemas/oic.r.autofocus#",
            "$schema": "http://json-schema.org/draft-04/schema#",
            "title": "Auto Focus",
            "definitions": {
              "oic.r.autofocus": {
                "type": "object",
                "properties": {
                  "autoFocus": {
                    "type": "boolean",
                    "description": "Status of the Auto Focus"
                  }
                }
              }
            },
            "type": "object",
            "allOf": [
              {"$ref": "oic.core.json#/definitions/oic.core"},
              {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
              {"$ref": "#/definitions/oic.r.autofocus"}
            ],
            "required": [ "autoFocus" ]
          }

        example: |
          {
            "id":         "unique_example_id",
            "autoFocus":  true
          }

    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.autofocus#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Auto Focus",
                "definitions": {
                  "oic.r.autofocus": {
                    "type": "object",
                    "properties": {
                      "autoFocus": {
                        "type": "boolean",
                        "description": "Status of the Auto Focus"
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.autofocus"}
                ],
                "required": [ "autoFocus" ]
              }

            example: |
              {
                "id":         "unique_example_id",
                "autoFocus":  true
              }
```

6.26.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
autoFocus	boolean	yes	Read Write	Status of the Auto Focus

6.26.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/AutoFocusResURI		get	post		

6.27 Automatic Document Feeder

6.27.1 Introduction

This resource describes the state of an automatic document feeder, typically used with a scanner. The states are read only. The adfStates is a comma separated list of the possible operational states. adfProcessing – the OK state, other states are errors or require ‘user attention’. The currentAdfState is the current value of the ADF state on the device.

6.27.2 Example URI

/AutomaticDocumentFeederResURI

6.27.3 Resource Type

The resource type (rt) is defined as: oic.r.automaticDocumentFeeder.

6.27.4 RAML Definition

```
#%RAML 0.8
title: OICAutomaticDocumentFeeder
version: v1.0-20150727
traits:
 - interface :
     queryParameters:
       if:
         enum: ["oic.if.s"]


/AutomaticDocumentFeederResURI:
  description: |
    This resource describes the state of an automatic document feeder, typically used with a
scanner.
    The states are read only.
    The adfStates is a comma separated list of the possible operational states.
    adfProcessing – the OK state, other states are errors or require ‘user attention’.
    The currentAdfState is the current value of the ADF state on the device.

  is : ['interface']
  get:
    description: |
      Retrieves the current automatic document feeder state.

    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.automaticDocumentFeeder#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Automatic Document Feeder",
                "definitions": {
                  "oic.r.automaticDocumentFeeder": {
                    "type": "object",
                    "properties": {
                      "adfStates": {
                        "type": "string",
                        "description": "ReadOnly, Comma separated list of the possible adf states."
                      },
                      "currentAdfState": {
                        "type": "string",
                        "description": "ReadOnly, Current adf state."
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.automaticDocumentFeeder"}
                ],
                "required": ["adfStates", "currentAdfState"]
              }

            example: |
              {
                "rt":               "oic.r.automaticDocumentFeeder",
                "id":               "unique_example_id",
                "adfStates":        "adfProcessing, adfEmpty, adfJam, adfLoaded, adfMispick,
adfHatchOpen, adfDuplexPageTooShort, adfDuplexPageTooLong, adfMultipickDetected, adfInputTrayFailed,
adfInputTrayOverloaded",
                "currentAdfState":  "adfProcessing"
              }
```

6.27.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
adfStates	string	yes	Read Only	Comma Separated List Of The Possible Adf States.
currentAdfState	string	yes	Read Only	Current Adf State.

6.27.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/AutomaticDocumentFeederResURI		get			

6.28 Button Switch

6.28.1 Introduction

This resource describes the operation of a button style switch. The value is a boolean. A value of True means that the button is being pushed/pressed. A value of False means that the button is not being pushed/pressed.

6.28.2 Example URI

/ButtonResURI

6.28.3 Resource Type

The resource type (rt) is defined as: oic.r.button.

6.28.4 RAML Definition

```
#%RAML 0.8

title: OICButton
version: v1.0-20150727

traits:
 - interface :
     queryParameters:

       if:
         enum: ["oic.if.s"]


/ButtonResURI:

  description: |
    This resource describes the operation of a button style switch.
    The value is a boolean.
    A value of True means that the button is being pushed/pressed.
    A value of False means that the button is not being pushed/pressed.

  is : ['interface']

  get:
    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.button.json#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Button Switch",
                "definitions": {
                  "oic.r.button": {
                    "properties": {
                      "value": {
                        "type": "boolean",
                        "description": "ReadOnly, Status of the button"
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.button"}
                ],
                "required": ["value"]
              }

            example: |
              {
                "rt":     "oic.r.button",
                "id":     "unique_example_id",
                "value":  true
              }

```

6.28.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
value	boolean	yes	Read Only	Status Of The Button

6.28.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/ButtonResURI		get			

6.29 Carbon Dioxide Sensor

6.29.1 Introduction

This resource describes whether carbon dioxide has been sensed or not. The value is a boolean. A value of True means that carbon dioxide has been detected. A value of False means that carbon dioxide has not been detected.

6.29.2 Example URI

/CarbonDioxideResURI

6.29.3 Resource Type

The resource type (rt) is defined as: oic.r.sensor.carbonDioxide.

6.29.4 RAML Definition

```
#%RAML 0.8
title: OICCarbonDioxideSensor
version: v1.0-20157527
traits:
 - interface :
     queryParameters:
       if:
         enum: ["oic.if.s"]

/CarbonDioxideResURI:
  description: |
    This resource describes whether carbon dioxide has been sensed or not.
    The value is a boolean.
    A value of True means that carbon dioxide has been detected.
    A value of False means that carbon dioxide has not been detected.

  is : ['interface']
  get:
    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.sensor.carbonDioxide.json#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Carbon Dioxide Sensor",
                "definitions": {
                  "oic.r.sensor.carbonDioxide": {
                    "allOf": [
                      {"$ref": "oic.r.sensor.json#/definitions/oic.r.sensor"}
                    ]
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.sensor.carbonDioxide"}
                ],
                "required": ["value"]
              }

            example: |
              {
                "rt":    "oic.r.sensor.carbonDioxide",
                "id":    "unique_example_id",
                "value": true
              }
```

6.29.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
value	boolean	yes	Read Only	True = Sensed, False = Not Sensed.

6.29.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/CarbonDioxideResURI		get			

6.30 Carbon Monoxide Sensor

6.30.1 Introduction

This resource describes whether carbon monoxide has been sensed or not. The value is a boolean. A value of True means that carbon monoxide has been detected. A value of False means that carbon monoxide has not been detected.

6.30.2 Example URI

/CarbonMonoxideResURI

6.30.3 Resource Type

The resource type (rt) is defined as: oic.r.sensor.carbonMonoxide.

6.30.4 RAML Definition

```
#%RAML 0.8
title: OICCarbonMonoxideSensor
version: v1.0-20150727
traits:
 - interface :
     queryParameters:
       if:
         enum: ["oic.if.s"]

/CarbonMonoxideResURI:
  description: |
    This resource describes whether carbon monoxide has been sensed or not.
    The value is a boolean.
    A value of True means that carbon monoxide has been detected.
    A value of False means that carbon monoxide has not been detected.

  is : ['interface']
  get:
    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.sensor.carbonMonoxide.json#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Carbon Monoxide Sensor",
                "definitions": {
                  "oic.r.sensor.carbonMonoxide": {
                    "allOf": [
                      {"$ref": "oic.r.sensor.json#/definitions/oic.r.sensor"}
                    ]
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.sensor.carbonMonoxide"}
                ],
                "required": ["value"]
              }

            example: |
              {
                "rt":     "oic.r.sensor.carbonMonoxide",
                "id":     "unique_example_id",
                "value":  true
              }

```

6.30.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
value	boolean	yes	Read Only	True = Sensed, False = Not Sensed.

6.30.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/CarbonMonoxideResURI		get			

6.31 Auto White Balance

6.31.1 Introduction

This resource describes an auto balance on/off feature (on/off). The value is a boolean. An AutoWhiteBalance value of 'true' means that the switch is on. An AutoWhiteBalance value of 'false' means that the switch is off.

6.31.2 Example URI

/AutoWhiteBalanceResURI

6.31.3 Resource Type

The resource type (rt) is defined as: oic.r.colour.autowhitebalance.

6.31.4 RAML Definition

```
#%RAML 0.8
title: OICAutoWhiteBalance
version: v1.0-20150727
traits:
 - interface :
     queryParameters:
       if:
         enum: ["oic.if.a"]


/AutoWhiteBalanceResURI:
  description: |
    This resource describes an auto balance on/off feature (on/off).
    The value is a boolean.
    An AutoWhiteBalance value of 'true' means that the switch is on.
    An AutoWhiteBalance value of 'false' means that the switch is off.

  is : ['interface']
  get:
    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.colour.autowhitebalance#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Auto White Balance",
                "definitions": {
                  "oic.r.colour.autowhitebalance": {
                    "type": "object",
                    "properties": {
                      "autoWhiteBalance": {
                        "type": "boolean",
                        "description": "Status of the Auto White balance"
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.colour.autowhitebalance"}
                ],
                "required": [ "autoWhiteBalance" ]
              }

            example: |
              {
                "rt":               "oic.r.colour.autowhitebalance",
                "id":               "unique_example_id",
                "autoWhiteBalance": false
              }

  post:
    body:
      application/json:
        schema: |
          {
            "id": "http://openinterconnect.org/schemas/oic.r.colour.autowhitebalance#",
            "$schema": "http://json-schema.org/draft-04/schema#",
            "title": "Auto White Balance",
            "definitions": {
              "oic.r.colour.autowhitebalance": {
                "type": "object",
                "properties": {
                  "autoWhiteBalance": {
                    "type": "boolean",
                    "description": "Status of the Auto White balance"
                  }
                }
              }
            },
            "type": "object",
            "allOf": [
              {"$ref": "oic.core.json#/definitions/oic.core"},
              {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
              {"$ref": "#/definitions/oic.r.colour.autowhitebalance"}
            ],
            "required": [ "autoWhiteBalance" ]
          }

        example: |
          {
            "id":               "unique_example_id",
            "autoWhiteBalance": true
          }

    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.colour.autowhitebalance#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Auto White Balance",
                "definitions": {
                  "oic.r.colour.autowhitebalance": {
                    "type": "object",
                    "properties": {
                      "autoWhiteBalance": {
                        "type": "boolean",
                        "description": "Status of the Auto White balance"
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.colour.autowhitebalance"}
                ],
                "required": [ "autoWhiteBalance" ]
              }

            example: |
              {
                "id":               "unique_example_id",
                "autoWhiteBalance": true
              }
```

6.31.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
autoWhiteBalance	boolean	yes	Read Write	Status of the Auto White balance

6.31.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/AutoWhiteBalanceResURI		get	post		

6.32 Colour Saturation

6.32.1 Introduction

This resource describes a Colour saturation value. The value is an integer. A coloursaturation has a range of [0,100]. A coloursaturation value of 0 means producing black and white images. A coloursaturation value of 50 means producing device specific normal colour images. A coloursaturation value of 100 means producing device very full colour images.

6.32.2 Example URI

/ColourSaturationResURI

6.32.3 Resource Type

The resource type (rt) is defined as: oic.r.colour.saturation.

6.32.4 RAML Definition

```
#%RAML 0.8

title: OICColourSaturation
version: v1.0-20150727

traits:
 - interface :
     queryParameters:

       if:
         enum: ["oic.if.a"]
```

```

/ColourSaturationResURI:
  description: |
    This resource describes a Colour saturation value.
    The value is an integer.
    A coloursaturation has a range of [0,100].
    A coloursaturation value of 0 means producing black and white images.
    A coloursaturation value of 50 means producing device specific normal colour images.
    A coloursaturation value of 100 means producing device very full colour images.

  is : ['interface']
  get:
    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.colour.saturation#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Colour Saturation",
                "definitions": {
                  "oic.r.colour.saturation": {
                    "type": "object",
                    "properties": {
                      "colourSaturation": {
                        "type": "integer",
                        "description": "The colour saturation value",
                        "minimum": 0,
                        "maximum": 100
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.colour.saturation"}
                ],
                "required": [ "colourSaturation" ]
              }

            example: |
              {
                "rt":               "oic.r.colour.saturation",
                "id":               "unique_example_id",
                "colourSaturation": 50
              }

  post:
    body:
      application/json:
        schema: |
          {
            "id": "http://openinterconnect.org/schemas/oic.r.colour.saturation#",
            "$schema": "http://json-schema.org/draft-04/schema#",
            "title": "Colour Saturation",
            "definitions": {
              "oic.r.colour.saturation": {
                "type": "object",
                "properties": {
                  "colourSaturation": {
                    "type": "integer",
                    "description": "The colour saturation value",
                    "minimum": 0,
                    "maximum": 100
                  }
                }
              }
            },
            "type": "object",
            "allOf": [
              {"$ref": "oic.core.json#/definitions/oic.core"},
              {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
              {"$ref": "#/definitions/oic.r.colour.saturation"}
            ],
            "required": [ "colourSaturation" ]
          }

        example: |
          {
            "id":               "unique_example_id",
            "colourSaturation": 60
          }

    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.colour.saturation#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Colour Saturation",
                "definitions": {
                  "oic.r.colour.saturation": {
                    "type": "object",
                    "properties": {
                      "colourSaturation": {
                        "type": "integer",
                        "description": "The colour saturation value",
                        "minimum": 0,
                        "maximum": 100
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.colour.saturation"}
                ],
                "required": [ "colourSaturation" ]
              }

            example: |
              {
                "id":               "unique_example_id",
                "colourSaturation": 60
              }
```

6.32.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
colourSaturation	integer	yes	Read Write	The colour saturation value
minimum				

maximum				

6.32.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/ColourSaturationResURI		get	post		

6.33 Contact Sensor

6.33.1 Introduction

This resource describes whether a contact sensor has been tripped or not. Typical use case is in Security Systems detecting window or door open. The value is a boolean. A value of True means that contact has been broken (open). A value of False means that contact is in place (closed).

6.33.2 Example URI

/ContactResURI

6.33.3 Resource Type

The resource type (rt) is defined as: oic.r.sensor.contact.

6.33.4 RAML Definition

```
#%RAML 0.8
title: OICContactSensor
version: v1.0-20150727
traits:
 - interface :
     queryParameters:
       if:
         enum: ["oic.if.s"]

/ContactResURI:
  description: |
    This resource describes whether a contact sensor has been tripped or not.
    Typical use case is in Security Systems detecting window or door open.
    The value is a boolean.
    A value of True means that contact has been broken (open).
    A value of False means that contact is in place (closed).

  is : ['interface']
  get:
    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.sensor.contact.json#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Contact Sensor",
                "definitions": {
                  "oic.r.sensor.contact": {
                    "allOf": [
                      {"$ref": "oic.r.sensor.json#/definitions/oic.r.sensor"}
                    ]
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.sensor.contact"}
                ],
                "required": ["value"]
              }

            example: |
              {
                "rt":     "oic.r.sensor.contact",
                "id":     "unique_example_id",
                "value":  true
              }

```

6.33.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
value	boolean	yes	Read Only	True = Sensed, False = Not Sensed.

6.33.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/ContactResURI		get			

6.34 Demand Response Load Control (DRLC).

6.34.1 Introduction

This resource describes any to be applied or currently being applied DRLC signal. The DRType is the ApplianceLoadReductionType defined in Zigbee/HA Smart Energy Profile 2.0. Start is a string containing an ISO8601 encoded start time. The duration value is in minutes. Override indicates whether the consumer has overridden the request (true) or not (false).

6.34.2 Example URI

/DRLCResURI

6.34.3 Resource Type

The resource type (rt) is defined as: oic.r.energy.drlc.

6.34.4 RAML Definition

```
#%RAML 0.8
title: OICDRLC
version: v1.0-20150727
traits:
 - interface :
     queryParameters:
       if:
         enum: ["oic.if.lb"]


/DRLCResURI:
  description: |
    This resource describes any to be applied or currently being applied DRLC signal.
    The DRType is the ApplianceLoadReductionType defined in Zigbee/HA Smart Energy Profile 2.0.
    Start is a string containing an ISO8601 encoded start time.
    The duration value is in minutes.
    Override indicates whether the consumer has overridden the request (true) or not (false).

  is : ['interface']
  get:
    description: |
      Provides the current DRLC action that is being applied.

    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.energy.drlc#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "definitions": {
                  "oic.r.energy.drlc": {
                    "type": "object",
                    "properties": {
                      "DRType":   {
                        "type": "integer",
                        "description": "The to be applied demand-response type"
                      },
                      "start":    {
                        "type": "string",
                        "description": "The start time for the application of DR"
                      },
                      "duration": {
                        "type": "integer",
                        "description": "The duration of the to be applied DR type"
                      },
                      "override": {
                        "type": "boolean",
                        "description": "Whether the consumer has overriden the application of DR"
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.energy.drlc"}
                ],
                "required": ["DRType"]
              }

            example: |
              {
                "rt":       "oic.r.energy.drlc",
                "id":       "unique_example_id",
                "DRType":   1,
                "start":    "2015-01-09T16:45Z",
                "duration": 10,
                "override": false
              }

  put:
    description: |
      Provides the DRLC action to be applied to the device or updates an existing action.

    body:
      application/json:
        schema: |
          {
            "id": "http://openinterconnect.org/schemas/oic.r.energy.drlc#",
            "$schema": "http://json-schema.org/draft-04/schema#",
            "definitions": {
              "oic.r.energy.drlc": {
                "type": "object",
                "properties": {
                  "DRType":   {
                    "type": "integer",
                    "description": "The to be applied demand-response type"
                  },
                  "start":    {
                    "type": "string",
                    "description": "The start time for the application of DR"
                  },
                  "duration": {
                    "type": "integer",
                    "description": "The duration of the to be applied DR type"
                  },
                  "override": {
                    "type": "boolean",
                    "description": "Whether the consumer has overriden the application of DR"
                  }
                }
              }
            },
            "type": "object",
            "allOf": [
              {"$ref": "oic.core.json#/definitions/oic.core"},
              {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
              {"$ref": "#/definitions/oic.r.energy.drlc"}
            ],
            "required": ["DRType"]
          }

        example: |
          {
            "rt":       "oic.r.energy.drlc",
            "id":       "unique_example_id",
            "DRType":   1,
            "start":    "2015-01-09T16:45Z",
            "duration": 10
          }

    responses :
      200:
        description: |
          Indicates that the target DRLC resource was changed.
          The new resource attributes are provided in the response.

        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.energy.drlc#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "definitions": {
                  "oic.r.energy.drlc": {
                    "type": "object",
                    "properties": {
                      "DRType":   {
                        "type": "integer",
                        "description": "The to be applied demand-response type"
                      },
                      "start":    {
                        "type": "string",
                        "description": "The start time for the application of DR"
                      },
                      "duration": {
                        "type": "integer",
                        "description": "The duration of the to be applied DR type"
                      },
                      "override": {
                        "type": "boolean",
                        "description": "Whether the consumer has overriden the application of DR"
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.energy.drlc"}
                ],
                "required": ["DRType"]
              }

            example: |
              {
                "DRType":   1,
                "id":       "unique_example_id",
                "start":    "2015-01-09T17:00Z",
                "duration": 15,
                "override": false
              }

      201:
        description: |
          Indicates successful creation of the DRLC resource with the attributes provided.
          The response includes the URI of the created resource.

        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "type": "object",
                "definitions": {
                  "oic.create": {
                    "type": "object",
                    "properties": {
                      "ResURI": { "type": "string"}
                    }
                  }
                },
                "type": "object",
                "$ref": "#/definitions/oic.create"
              }

            example: |
              {
                "ResURI":   "/MyDevice/MyDRLCURI"
              }
```

6.34.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
DRType	integer	yes	Read Write	The to be applied demand-response type
start	string		Read Write	The start time for the application of DR
duration	integer		Read Write	The duration of the to be applied DR type
override	boolean		Read Write	Whether the consumer has overriden the application of DR

6.34.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify

/DRLCResURI	put	get			

6.35 Energy Overload/Circuit Breaker

6.35.1 Introduction

This resource describes whether an energy overload detector/circuit breaker has been tripped. The value is a boolean. A value of True means that energy overload has been tripped. A value of False means that energy overload has not been tripped.

6.35.2 Example URI

/EnergyOverloadResURI

6.35.3 Resource Type

The resource type (rt) is defined as: oic.r.energy.overload.

6.35.4 RAML Definition

```
#%RAML 0.8
title: OICEnergyOverload
version: v1.0-20150727
traits:
 - interface :
     queryParameters:
       if:
         enum: ["oic.if.s"]

/EnergyOverloadResURI:
  description: |
    This resource describes whether an energy overload detector/circuit breaker
      has been tripped.
    The value is a boolean.
    A value of True means that energy overload has been tripped.
    A value of False means that energy overload has not been tripped.

  is : ['interface']
  get:
    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.energy.overload.json#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Energy Overload Sensor",
                "definitions": {
                  "oic.r.energy.overload": {
                    "allOf": [
                      {"$ref": "oic.r.sensor.json#/definitions/oic.r.sensor"}
                    ]
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.energy.overload"}
                ],
                "required": ["value"]
              }

            example: |
              {
                "rt":    "oic.r.energy.overload",
                "id":    "unique_example_id",
                "value": true
              }

```

6.35.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
value	boolean	yes	Read Only	True = Sensed, False = Not Sensed.

6.35.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/EnergyOverloadResURI		get			

6.36 Generic Sensor

6.36.1 Introduction

This resource describes whether some value or property or entity has been sensed or not. The value is a boolean. A value of True means that the target has been sensed. A value of False means that the target has not been sensed.

6.36.2 Example URI

/GenericSensorResURI

6.36.3 Resource Type

The resource type (rt) is defined as: oic.r.sensor.

6.36.4 RAML Definition

```
#%RAML 0.8
title: OICGenericSensor
version: v1.0-20150727
traits:
 - interface :
     queryParameters:
       if:
         enum: ["oic.if.s"]

/GenericSensorResURI:
  description: |
    This resource describes whether some value or property or entity has been sensed or not.
    The value is a boolean.
    A value of True means that the target has been sensed.
    A value of False means that the target has not been sensed.

  is : ['interface']
  get:
    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.sensor.json#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Generic Sensor",
                "definitions": {
                  "oic.r.sensor": {
                    "type": "object",
                    "properties": {
                      "value": {
                        "type": "boolean",
                        "description": "ReadOnly, true = sensed, false = not sensed."
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.sensor"}
                ]
              }

            example: |
              {
                "rt":    "oic.r.sensor",
                "id":    "unique_example_id",
                "value": true
              }

```

6.36.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
value	boolean		Read Only	True = Sensed, False = Not Sensed.

6.36.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/GenericSensorResURI		get			

6.37 Glass Break Sensor

6.37.1 Introduction

This resource describes a glass break sensor. The value is a boolean. A value of True means that glass break has been sensed. A value of False means that glass break not been sensed.

6.37.2 Example URI

/GlassBreakResURI

6.37.3 Resource Type

The resource type (rt) is defined as: oic.r.sensor.glassBreak.

6.37.4 RAML Definition

```
#%RAML 0.8
title: OICGlassBreakSensor
version: v1.0-20150727
traits:
 - interface :
     queryParameters:
       if:
         enum: ["oic.if.s"]


/GlassBreakResURI:
  description: |
    This resource describes a glass break sensor.
    The value is a boolean.
    A value of True means that glass break has been sensed.
    A value of False means that glass break not been sensed.

  is : ['interface']
  get:
    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.sensor.glassBreak.json#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Glass Break Sensor",
                "definitions": {
                  "oic.r.sensor.glassBreak": {
                    "allOf": [
                      {"$ref": "oic.r.sensor.json#/definitions/oic.r.sensor"}
                    ]
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.sensor.glassBreak"}
                ],
                "required": ["value"]
              }

            example: |
              {
                "rt":     "oic.r.sensor.glassBreak",
                "id":     "unique_example_id",
                "value":  true
              }
```

6.37.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
value	boolean	yes	Read Only	True = Sensed, False = Not Sensed.

6.37.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/GlassBreakResURI		get			

6.38 Heart Rate Zone

6.38.1 Introduction

This resource describes a measured heart rate by the current Zone using the Zoladz method The Zoladz method defines Zones based on maximum heart rate; Zone 1 is the lowest, Zone 5 is the highest. The heartRateZone is an enumeration containing one of: "Zone1", "Zone2", "Zone3", "Zone4", "Zone5".

6.38.2 Example URI

/HeartRateZoneResURI

6.38.3 Resource Type

The resource type (rt) is defined as: oic.r.sensor.heart.zone.

6.38.4 RAML Definition

```
#%RAML 0.8
title: OICHeartRateZone
version: v1.0-20150727
traits:
 - interface :
     queryParameters:
       if:
         enum: ["oic.if.s"]


/HeartRateZoneResURI:
  description: |
    This resource describes a measured heart rate by the current Zone using the Zoladz method
    The Zoladz method defines Zones based on maximum heart rate; Zone 1 is the lowest, Zone 5 is
the highest.
    The heartRateZone is an enumeration containing one of: "Zone1", "Zone2", "Zone3", "Zone4",
"Zone5".

  is : ['interface']
  get:
    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.sensor.heart.zone.json#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Heart Rate Zone",
                "definitions": {
                  "oic.r.sensor.heart.zone": {
                    "properties": {
                      "heartRateZone": {
                          "enum": ["Zone1","Zone2","Zone3", "Zone4", "Zone5"],
                          "description": "ReadOnly, current heart rate zone based on the Zoladz
system."
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.sensor.heart.zone"}
                ],
                "required": ["heartRateZone"]
              }

            example: |
              {
                "rt":             "oic.r.sensor.heart.zone",
                "id":             "unique_example_id",
                "heartRateZone":  "Zone3"
              }

```

6.38.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
heartRateZone	enum	yes	Read Only	Current Heart Rate Zone Based On The Zoladz System.

6.38.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/HeartRateZoneResURI		get			

6.39 Illuminance Sensor

6.39.1 Introduction

This resource describes an illuminance sensor illuminance is a float and represents the sensed luminous flux per unit area in lux.

6.39.2 Example URI

/IlluminanceSensorResURI

6.39.3 Resource Type

The resource type (rt) is defined as: oic.r.sensor.illuminance.

6.39.4 RAML Definition

```
#%RAML 0.8
title: OICIlluminanceSensor
version: v1.0-20150727
traits:
 - interface :
     queryParameters:
       if:
         enum: ["oic.if.s"]

/IlluminanceSensorResURI:
  description: |
    This resource describes an illuminance sensor
    illuminance is a float and represents the sensed luminous flux per unit area in lux.

  is : ['interface']
  get:
    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.sensor.illuminance.json#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Illuminance Sensor",
                "definitions": {
                  "oic.r.sensor.illuminance": {
                    "properties": {
                      "illuminance": {
                        "type": "number",
                        "description": "ReadOnly, sensed luminous flux per unit area in lux."
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.sensor.illuminance"}
                ],
                "required": ["illuminance"]
              }

            example: |
              {
                "rt":           "oic.r.sensor.illuminance",
                "id":           "unique_example_id",
                "illuminance":  450
              }
```

6.39.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
illuminance	number	yes	Read Only	Sensed Luminous Flux Per Unit Area In Lux.

6.39.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/IlluminanceSensorResURI		get			

6.40 Magnetic Field Direction Sensor

6.40.1 Introduction

This resource describes the direction of the Earth’s magnetic field at the observer's current point in space. Typical use case includes measurement of compass readings on a personal device. The value is a CSV of Hx, Hy, Hz. Each of Hx, Hy and Hz are expressed in A/m (Amperes per metre)

6.40.2 Example URI

/MagneticFieldDirectionResURI

6.40.3 Resource Type

The resource type (rt) is defined as: oic.r.sensor.magneticFieldDirection.

6.40.4 RAML Definition

```
#%RAML 0.8

title: OICMagneticFieldDirection
version: v1.0-20150727

traits:
 - interface :
     queryParameters:

       if:
         enum: ["oic.if.s"]


/MagneticFieldDirectionResURI:

  description: |
    This resource describes the direction of the Earth’s magnetic field at the observer's current
point in space.
    Typical use case includes measurement of compass readings on a personal device.
    The value is a CSV of Hx, Hy, Hz.
    Each of Hx, Hy and Hz are expressed in A/m (Amperes per metre)


  is : ['interface']

  get:
    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id":
"http://openinterconnect.org/schemas/oic.r.sensor.magneticFieldDirection.json#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Magnetic Field Direction Sensor",
                "definitions": {
                  "oic.r.sensor.magneticFieldDirection": {
                    "properties": {
                      "value": {
                        "type": "string",
                        "description": "ReadOnly, CSV containing Hx, Hy, Hz."
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.sensor.magneticFieldDirection"}
                ],
                "required": ["value"]
              }

            example: |
              {
                "rt":     "oic.r.sensor.magneticFieldDirection",
                "id":     "unique_example_id",
                "value":  "100,15,90"
              }

```

6.40.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
value	string	yes	Read Only	Csv Containing Hx, Hy, Hz.

6.40.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/MagneticFieldDirectionResURI		get			

6.41 Media

6.41.1 Introduction

This resource specifies the media that an OIC Server (Camera) supports. The resource is an array of media elements. Each element contains: A URL at which the specified media type can be accessed. A string array containing the definition of the media using SDP. Each entry in the sdp array is an SDP line. Each line shall follow the SDP description syntax as defined in the SDP specification. The SDP specification can be found at http://tools.ietf.org/html/rfc4566. The mime subtype video/H264 indicates video resource and the mime subtype video/jpeg indicates still image resource.

6.41.2 Example URI

/MediaResURI

6.41.3 Resource Type

The resource type (rt) is defined as: oic.r.media.

6.41.4 RAML Definition

```
#%RAML 0.8
title: OICMedia
version: v1.0-20150727
traits:
 - interface :
     queryParameters:
       if:
         enum: ["oic.if.s"]


/MediaResURI:
  description: |
    This resource specifies the media that an OIC Server (Camera) supports.
    The resource is an array of media elements
     Each element contains:
        A URL at which the specified media type can be accessed.
        A string array containing the definition of the media using SDP.
        Each entry in the sdp array is an SDP line.
        Each line shall follow the SDP description syntax as defined in the SDP specification.
    The SDP specification can be found at http://tools.ietf.org/html/rfc4566.
    The mime subtype video/H264 indicates video resource and the mime subtype video/jpeg indicates
still image resource.

  is : ['interface']
  get:
    description: |
      Retrieves the current media resource.

    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.media#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Media",
                "definitions": {
                  "oic.r.media": {
                    "media": {
                      "type": "array",
                      "items": {
                        "type": "object",
                        "properties": {
                          "url": {
                            "type": "string",
                            "description": "url for the media instance"
                          },
                          "sdp": {
                            "type": "array",
                            "description": "Array of strings, one per SDP line",
                            "items": {
                              "type": "string",
                              "description": "SDP media or attribute line"
                            }
                          }
                        }
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.media"}
                ],
                "required": ["media"]
              }

            example: |
              {
                "rt": "oic.r.media",
                "id": "unique_example_id",
                "media": [
                  {
                    "url": "some example url",
                    "sdp": [
                      "m=video 1 RTP/AVP 96",
                      "a=rtpmap:96 H264/9000",
                      "a=fmtp:96 profile-level-id=42A028;packetization-mode=1"
                    ]
                  },
                  {
                    "url": "some other example1 url",
                    "sdp": [
                      "m=audio 2 RTP/AVP 97",
                      "a=rtpmap:97 MP4A-LATM/90000"
                    ]
                  },
                  {
                    "url": "some other example2 url",
                    "sdp": [
                      "m=video 3 RTP/AVP 98",
                      "a=rtpmap:98 jpeg/90000",
                      "a=fmtp:98 sampling=YCbCr-4:2:0;width=256;height=256"
                    ]
                  }
                ]
              }
```

6.41.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
media	array	yes		
url	string		Read Write	url for the media instance
sdp	array		Read Write	Array of strings, one per SDP line

6.41.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/MediaResURI		get			

6.42 Media Source List

6.42.1 Introduction

This resource provides the list of media sources available on the device. The sources are an array of mediaSource(s) as separately defined. The basic resource type oic.r.mediaSourceList does not provide any indications whether the source is input or output. Hence, two specializations of this resource exist. When a device exposes input sources then an instance of this resource with a resource type of oic.r.media.input is exposed. When a device exposes output sources then an instance of this resource with a resource type of oic.r.media.output is exposed. A device that exposes both input and output media sources then exposes two instances of this resource, one with a resource type or oic.r.media.input and one with a resource type of oic.r.media.output

6.42.2 Example URI

/mediaSourceListResURI

6.42.3 Resource Type

The resource type (rt) is defined as: oic.r.mediaSourceList.

6.42.4 RAML Definition

```
#%RAML 0.8
title: OICMediaSourceList
version: v1.0-20151019
traits:
 - interface :
     queryParameters:
       if:
         enum: ["oic.if.a"]


/mediaSourceListResURI:
  description: |
    This resource provides the list of media sources available on the device.
    The sources are an array of mediaSource(s) as separately defined.
    The basic resource type oic.r.mediaSourceList does not provide any indications whether the
source is input or output.
    Hence, two specializations of this resource exist.
    When a device exposes input sources then an instance of this resource with a resource type of
oic.r.media.input is exposed.
    When a device exposes output sources then an instance of this resource with a resource type of
oic.r.media.output is exposed.
    A device that exposes both input and output media sources then exposes two instances of this
resource,
    one with a resource type or oic.r.media.input and one with a resource type of
oic.r.media.output

  is : ['interface']
  get:
    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "$schema": "http://json-schemas.org/draft-04/schema#",
                "id": "http://openinterconnect.org/schemas/oic.r.mediaSourceList.json#",
                "title": "Media Source List",
                "definitions": {
                  "oic.r.mediaSourceList": {
                    "properties": {
                      "sources": {
                        "type": "array",
                        "items": {
                          "oneOf": [
                            { "$ref": "oic.r.mediaSource.json#" }
                          ]
                        }
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  { "$ref": "oic.core.json#/definitions/oic.core"},
                  { "$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  { "$ref": "#/definitions/oic.r.mediaSourceList"}
                ],
                "required": ["sources"]
              }

            example: |
              {
                "rt": "oic.r.mediaSourceList",
                "id": "unique_example_id",
                "sources": [
                  {
                    "sourceName": "HDMI-CEC",
                    "sourceNumber": "1",
                    "sourceType": "audioPlusVideo",
                    "status": true
                  },
                  {
                    "sourceName": "dualRCA",
                    "sourceNumber": "1",
                    "sourceType": "audioOnly",
                    "status": false
                  }
                ]
              }

  post:
    description: |
      Changes the status of the source(s).
      Allows changes of the sourceName and the status.

    body:
      application/json:
        schema: |
          {
            "$schema": "http://json-schemas.org/draft-04/schema#",
            "id": "http://openinterconnect.org/schemas/oic.r.mediaSourceList.json#",
            "title": "Media Source List",
            "definitions": {
              "oic.r.mediaSourceList": {
                "properties": {
                  "sources": {
                    "type": "array",
                    "items": {
                      "oneOf": [
                        { "$ref": "oic.r.mediaSource.json#" }
                      ]
                    }
                  }
                }
              }
            },
            "type": "object",
            "allOf": [
              { "$ref": "oic.core.json#/definitions/oic.core"},
              { "$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
              { "$ref": "#/definitions/oic.r.mediaSourceList"}
            ],
            "required": ["sources"]
          }

        example: |
          {
            "id":     "unique_example_id",
            "sources": [
              {
               "sourceName":    "my new name",
               "sourceNumber":  "1",
               "status":        true
              }
            ]
          }

    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "$schema": "http://json-schemas.org/draft-04/schema#",
                "id": "http://openinterconnect.org/schemas/oic.r.mediaSourceList.json#",
                "title": "Media Source List",
                "definitions": {
                  "oic.r.mediaSourceList": {
                    "properties": {
                      "sources": {
                        "type": "array",
                        "items": {
                          "oneOf": [
                            { "$ref": "oic.r.mediaSource.json#" }
                          ]
                        }
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  { "$ref": "oic.core.json#/definitions/oic.core"},
                  { "$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  { "$ref": "#/definitions/oic.r.mediaSourceList"}
                ],
                "required": ["sources"]
              }

            example: |
              {
                "id":     "unique_example_id",
                "sources": [
                  {
                    "sourceName":   "my new name",
                    "sourceNumber": "1",
                    "status":       true
                  }
                ]
              }
```

6.42.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
sources	array	yes		
sourceName	string	yes	Read Write	Specifies a pre-defined media input or output
sourceNumber	integer		Read Only	Numeric Identifier To Specify The Instance
sourceType	enum		Read Only	Specifies The Type Of The Source
status	boolean	yes	Read Write	Specifies if the specific source instance is selected or not

6.42.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/mediaSourceListResURI		get	post		

6.42.7 Referenced JSON schemas

6.42.7.1 oic.r.mediaSource.json

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "id": "http://openinterconnect.org/schemas/oic.r.mediaSource.json#",
  "title": "Media Source",
  "definitions": {
    "oic.r.mediaSource": {
      "properties": {
        "sourceName": {
          "type": "string",
          "description": "Specifies a pre-defined media input or output"
        },
        "sourceNumber": {
          "type": [ "integer", "string" ],
          "description": "ReadOnly, Numeric identifier to specify the instance"
        },
        "sourceType": {
          "enum": [ "audioOnly", "videoOnly", "audioPlusVideo" ],
          "description": "ReadOnly, Specifies the type of the source"
        },
        "status": {
          "type": "boolean",
          "description": "Specifies if the specific source instance is selected or not"
        }
      }
    }
  },
  "type": "object",
  "allOf": [
    {"$ref": "oic.core.json#/definitions/oic.core"},
    {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
    {"$ref": "#/definitions/oic.r.mediaSource"}
  ],
  "required": ["sourceName","status"]
}
```

6.43 Motion Sensor

6.43.1 Introduction

This resource describes whether motion has been sensed or not. The value is a boolean. A value of True means that motion has been sensed. A value of False means that motion not been sensed.

6.43.2 Example URI

/MotionResURI

6.43.3 Resource Type

The resource type (rt) is defined as: oic.r.sensor.motion.

6.43.4 RAML Definition

```
#%RAML 0.8

title: OICMotionSensor
version: v1.0-20150727

traits:
 - interface :
     queryParameters:

       if:
         enum: ["oic.if.s"]
```

```

/MotionResURI:
  description: |
    This resource describes whether motion has been sensed or not.
    The value is a boolean.
    A value of True means that motion has been sensed.
    A value of False means that motion not been sensed.

  is : ['interface']
  get:
    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.sensor.motion.json#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Motion Sensor",
                "definitions": {
                  "oic.r.sensor.motion": {
                    "allOf": [
                      {"$ref": "oic.r.sensor.json#/definitions/oic.r.sensor"}
                    ]
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.sensor.motion"}
                ],
                "required": ["value"]
              }

            example: |
              {
                "rt":    "oic.r.sensor.motion",
                "id":    "unique_example_id",
                "value": true
              }

```

6.43.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
value	boolean	yes	Read Only	True = Sensed, False = Not Sensed.

6.43.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/MotionResURI		get			

6.44 Night Mode

6.44.1 Introduction

This resource describes a night mode on/off feature (on/off). A nightMode value of 'true' means that the feature is on. A nightMode value of 'false' means that the feature is off.

6.44.2 Example URI

/NightModeResURI

6.44.3 Resource Type

The resource type (rt) is defined as: oic.r.nightMode.

6.44.4 RAML Definition

```
#%RAML 0.8
title: OICNightMode
version: v1.0-20150727
traits:
 - interface :
     queryParameters:
       if:
         enum: ["oic.if.a"]


/NightModeResURI:
  description: |
    This resource describes a night mode on/off feature (on/off).
    A nightMode value of 'true' means that the feature is on.
    A nightMode value of 'false' means that the feature is off.

  is : ['interface']
  get:
    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.nightMode.json#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Night Mode",
                "definitions": {
                  "oic.r.nightMode": {
                    "type": "object",
                    "properties": {
                      "nightMode": {
                        "type": "boolean",
                        "description": "Status of the Night Mode"
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.nightMode"}
                ],
                "required": [ "nightMode" ]
              }

            example: |
              {
                "rt":         "oic.r.nightMode",
                "id":         "unique_example_id",
                "nightMode":  false
              }

  post:
    body:
      application/json:
        schema: |
          {
            "id": "http://openinterconnect.org/schemas/oic.r.nightMode.json#",
            "$schema": "http://json-schema.org/draft-04/schema#",
            "title": "Night Mode",
            "definitions": {
              "oic.r.nightMode": {
                "type": "object",
                "properties": {
                  "nightMode": {
                    "type": "boolean",
                    "description": "Status of the Night Mode"
                  }
                }
              }
            },
            "type": "object",
            "allOf": [
              {"$ref": "oic.core.json#/definitions/oic.core"},
              {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
              {"$ref": "#/definitions/oic.r.nightMode"}
            ],
            "required": [ "nightMode" ]
          }

        example: |
          {
            "id":         "unique_example_id",
            "nightMode":  true
          }

    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.nightMode.json#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Night Mode",
                "definitions": {
                  "oic.r.nightMode": {
                    "type": "object",
                    "properties": {
                      "nightMode": {
                        "type": "boolean",
                        "description": "Status of the Night Mode"
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.nightMode"}
                ],
                "required": [ "nightMode" ]
              }

            example: |
              {
                "id":         "unique_example_id",
                "nightMode":  true
              }
```

6.44.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
nightMode	boolean	yes	Read Write	Status of the Night Mode

6.44.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/NightModeResURI		get	post		

6.45 Presence Sensor

6.45.1 Introduction

This resource describes whether presence has been sensed or not. The value is a boolean. A value of True means that presence has been sensed. A value of False means that presence not been sensed.

6.45.2 Example URI

/PresenceResURI

6.45.3 Resource Type

The resource type (rt) is defined as: oic.r.sensor.presence.

6.45.4 RAML Definition

```
#%RAML 0.8
title: OICPresenceSensor
version: v1.0-20150727
traits:
 - interface :
     queryParameters:
       if:
         enum: ["oic.if.s"]

/PresenceResURI:
  description: |
    This resource describes whether presence has been sensed or not.
    The value is a boolean.
    A value of True means that presence has been sensed.
    A value of False means that presence not been sensed.

  is : ['interface']
  get:
    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.sensor.presence.json#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Presence Sensor",
                "definitions": {
                  "oic.r.sensor.presence": {
                    "allOf": [
                      {"$ref": "oic.r.sensor.json#/definitions/oic.r.sensor"}
                    ]
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.sensor.presence"}
                ],
                "required": ["value"]
              }

            example: |
              {
                "rt":    "oic.r.sensor.presence",
                "id":    "unique_example_id",
                "value": true
              }
```

6.45.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
value	boolean	yes	Read Only	True = Sensed, False = Not Sensed.

6.45.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/PresenceResURI		get			

6.46 Pan Tilt Zoom Movement

6.46.1 Introduction

This resource specifies the pan tilt and zoom capabilities of a device. The resource rt is dynamic and reflects whether the values apply to physical movement of the device or digital/virtual enhancements to the image. For physical movement the rt is 'oic.r.movement.ptz'. For digital/virtual image enhancements the rt is 'oic.r.image.ptz'. The Pan and Tilt are specified in degrees. The Zoom Factor is a value in the range 1-100 for linear (optical) zoom. The Zoom Factor is a value in the range [1x, 2x, 4x, 8x, 16x, 32x] for digital zoom. If there is no zoom value to set the Zoom Factor shall be '1x'. The value 0 degrees means neutral, this is the vendor defined setting. Note that this resource also can be used to create an offset for physical movement. When that is the case, the rt value is: oic.r.movement.offset.ptz Note that this resource also can be used to create an offset for image movement. When that is the case, the rt value is: oic.r.image.offset.ptz When the pan_range value is omitted, then the range is [-180,180]. If pan is not supported then the range shall be [0,0] When the tilt_range value is omitted, then the range is [-180,180]. If tilt is not supported then the range shall be [0,0] Note: When the range is specified as a float (e.g. 180.0, 180.0) then the pan and tilt values are also floats.

6.46.2 Example URI

/PanTiltZoomResURI

6.46.3 Resource Type

The resource type (rt) is defined as: oic.r.ptz.

6.46.4 RAML Definition

```
#%RAML 0.8

title: OICPanTiltZoom
version: v1.0-20150805

traits:
 - interface :
     queryParameters:

       if:
         enum: ["oic.if.a"]
```

```
/PanTiltZoomResURI:
  description: |
    This resource specifies the pan tilt and zoom capabilities of a device.
    The resource rt is dynamic and reflects whether the values apply to
      physical movement of the device or digital/virtual enhancements to the image.
    For physical movement the rt is 'oic.r.movement.ptz'.
    For digital/virtual image enhancements the rt is 'oic.r.image.ptz'.
    The Pan and Tilt are specified in degrees.
    The Zoom Factor is a value in the range 1-100 for linear (optical) zoom.
    The Zoom Factor is a value in the range [1x, 2x, 4x, 8x, 16x, 32x] for digital zoom.
    If there is no zoom value to set the Zoom Factor shall be '1x'.
    The value 0 degrees means neutral, this is the vendor defined setting.
    Note that this resource also can be used to create an offset for physical movement.
    When that is the case, the rt value is: oic.r.movement.offset.ptz
    Note that this resource also can be used to create an offset for image movement.
    When that is the case, the rt value is: oic.r.image.offset.ptz
    When the pan_range value is omitted, then the range is [-180,180].
    If pan is not supported then the range shall be [0,0]
    When the tilt_range value is omitted, then the range is [-180,180].
    If tilt is not supported then the range shall be [0,0]
    Note: When the range is specified as a float (e.g
    180.0, 180.0) then the
      pan and tilt values are also floats.

  is : ['interface']
  get:
    description: |
      Retrieves the current pan, tilt and zoom setting.

    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.ptz#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Pan Tilt Zoom",
                "definitions": {
                  "oic.r.ptz": {
                    "type": "object",
                    "properties": {
                      "pan":  {
                        "type": "number",
                        "description": "horizontal pan in degrees"
                      },
                      "tilt":   {
                        "type": "number",
                        "description": "vertical tilt in degrees"
                      },
                      "pan_range":  {
                        "type": "string",
                        "description": "ReadOnly, Min and Max values for the pan setting",
                        "format": "csv"
                      },
                      "tilt_range":  {
                        "type": "string",
                        "description": "ReadOnly, Min and Max values for the tilt setting",
                        "format": "csv"
                      },
                      "zoomFactor": {
                        "type": "string",
                        "description": "The Zoomfactor value"
                      },
                      "zoomFactorRange": {
                        "enum": ["linear, 1x, 2x, 4x, 8x, 16x, 32x"],
                        "description": "ReadOnly, allowed Zoom Factor values. Linear equates to a 
1-100 min/max."
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.ptz"}
                ],
                "required": ["pan","tilt","zoomFactor"]
              }

            example: |
              {
                "rt":         "oic.r.ptz",
                "id":         "unique_example_id",
                "pan":        0,
                "tilt":       0,
                "zoomFactor": "2x"
              }

  post:
    description: |
      Sets the current pan, tilt and zoom value

    body:
      application/json:
        schema: |
          {
            "id": "http://openinterconnect.org/schemas/oic.r.ptz#",
            "$schema": "http://json-schema.org/draft-04/schema#",
            "title": "Pan Tilt Zoom",
            "definitions": {
              "oic.r.ptz": {
                "type": "object",
                "properties": {
                  "pan":  {
                    "type": "number",
                    "description": "horizontal pan in degrees"
                  },
                  "tilt":   {
                    "type": "number",
                    "description": "vertical tilt in degrees"
                  },
                  "pan_range":  {
                    "type": "string",
                    "description": "ReadOnly, Min and Max values for the pan setting",
                    "format": "csv"
                  },
                  "tilt_range":  {
                    "type": "string",
                    "description": "ReadOnly, Min and Max values for the tilt setting",
                    "format": "csv"
                  },
                  "zoomFactor": {
                    "type": "string",
                    "description": "The Zoomfactor value"
                  },
                  "zoomFactorRange": {
                    "enum": ["linear, 1x, 2x, 4x, 8x, 16x, 32x"],
                    "description": "ReadOnly, allowed Zoom Factor values. Linear equates to a 1-100 
min/max."
```

```
                  }
                }
              }
            },
            "type": "object",
            "allOf": [
              {"$ref": "oic.core.json#/definitions/oic.core"},
              {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
              {"$ref": "#/definitions/oic.r.ptz"}
            ],
            "required": ["pan","tilt","zoomFactor"]
          }

        example: |
          {
            "id":         "unique_example_id",
            "pan":        10,
            "tilt":       -10,
            "zoomFactor": "4x"
          }

    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.ptz#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Pan Tilt Zoom",
                "definitions": {
                  "oic.r.ptz": {
                    "type": "object",
                    "properties": {
                      "pan":  {
                        "type": "number",
                        "description": "horizontal pan in degrees"
                      },
                      "tilt":   {
                        "type": "number",
                        "description": "vertical tilt in degrees"
                      },
                      "pan_range":  {
                        "type": "string",
                        "description": "ReadOnly, Min and Max values for the pan setting",
                        "format": "csv"
                      },
                      "tilt_range":  {
                        "type": "string",
                        "description": "ReadOnly, Min and Max values for the tilt setting",
                        "format": "csv"
                      },
                      "zoomFactor": {
                        "type": "string",
                        "description": "The Zoomfactor value"
                      },
                      "zoomFactorRange": {
                        "enum": ["linear, 1x, 2x, 4x, 8x, 16x, 32x"],
                        "description": "ReadOnly, allowed Zoom Factor values. Linear equates to a
1-100 min/max."
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.ptz"}
                ],
                "required": ["pan","tilt","zoomFactor"]
              }

            example: |
              {
                "id":          "unique_example_id",
                "pan":         10,
                "tilt":        -10,
                "zoomFactor":  "4x"
              }
```

6.46.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
pan	number	yes	Read Write	horizontal pan in degrees
tilt	number	yes	Read Write	vertical tilt in degrees
pan_range	string		Read Only	Min And Max Values For The Pan Setting
tilt_range	string		Read Only	Min And Max Values For The Tilt Setting
zoomFactor	string	yes	Read Write	The Zoomfactor value
zoomFactorRange	enum		Read Only	Allowed Zoom Factor Values. Linear Equates To A 1-100 Min/Max.

6.46.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/PanTiltZoomResURI		get	post		

6.47 Signal Strength

6.47.1 Introduction

This resource describes the strength of a signal by means of lqi and rssi. The lqi is a floating point number that represents Link Quality Indicator. The rssi is a floating point number that represents the received signal strength indicator.

6.47.2 Example URI

/SignalStrengthResURI

6.47.3 Resource Type

The resource type (rt) is defined as: oic.r.signalStrength.

6.47.4 RAML Definition

```
#%RAML 0.8

title: OICSignalStrength
version: v1.0-20150727

traits:
 - interface :
     queryParameters:

       if:
         enum: ["oic.if.s"]


/SignalStrengthResURI:
```

```
description: |
  This resource describes the strength of a signal by means of lqi and rssi.
  The lqi is a floating point number that represents Link Quality Indicator.
  The rssi is a floating point number that represents the received signal strength indicator.

is : ['interface']
get:
  responses :
    200:
      body:
        application/json:
          schema: |
            {
              "id": "http://openinterconnect.org/schemas/oic.r.signalStrength.json#",
              "$schema": "http://json-schema.org/draft-04/schema#",
              "title": "Signal Strength",
              "definitions": {
                "oic.r.signalStrength": {
                  "type": "object",
                  "properties": {
                    "lqi": {
                      "type": "number",
                      "description": "ReadOnly, current value of Link Quality Indicator"
                    },
                    "rssi": {
                      "type": "number",
                      "description": "ReadOnly, current value of Received Signal Strength
Indicator"
                    }
                  }
                }
              },
              "type": "object",
              "allOf": [
                {"$ref": "oic.core.json#/definitions/oic.core"},
                {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                {"$ref": "#/definitions/oic.r.signalStrength"}
              ],
              "required": ["lqi","rssi"]
            }

          example: |
            {
              "rt":   "oic.r.signalStrength",
              "id":   "unique_example_id",
              "lqi":  10,
              "rssi": 55
            }
```

6.47.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
lqi	number	yes	Read Only	Current Value Of Link Quality Indicator
rssi	number	yes	Read Only	Current Value Of Received Signal Strength Indicator

6.47.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/SignalStrengthResURI		get			

6.48 Speech Synthesis-TTS

6.48.1 Introduction

This resource may be created on the OIC Server that is capable of rendering speech by an OIC Client or may be created on the OIC Server by some resident application. The audio rendered is at this stage local to the Server (i.e. not streamed). The utterance is an SSML document. The supportedLanguages is a comma separate value list of the RFC 5646 defined language tags that are supported. The supportedVoices is an SSML document fragment indicating the voices that are supported.

6.48.2 Example URI

/SpeechTTSResURI

6.48.3 Resource Type

The resource type (rt) is defined as: oic.r.speech.tts.

6.48.4 RAML Definition

```
#%RAML 0.8
title: OICSpeechTTS
version: v1.0-20150727
traits:
 - interface :
     queryParameters:
       if:
         enum: ["oic.if.a"]

/SpeechTTSResURI:
  description: |
    This resource may be created on the OIC Server that is capable of rendering speech by an OIC
Client
     or may be created on the OIC Server by some resident application.
    The audio rendered is at this stage local to the Server (i.e
    not streamed).
    The utterance is an SSML document.
    The supportedLanguages is a comma separate value list of the RFC 5646 defined language tags
that are supported.
    The supportedVoices is an SSML document fragment indicating the voices that are supported.

  is : ['interface']
  get:
    description: |
      Utterance in the example shall be a properly escaped (JSON rules) SSML document
      An example is given below:
        "<?xml version=\"1.0\" encoding=\"ISO-8859-1\"?>\n\r
        <speak version=\"1.1\" xmlns=\"http:\/\/www.w3.org\/2001\/10\/synthesis\"\n\r
        \txmlns:xsi=\"http:\/\/www.w3.org\/2001\/XMLSchema-instance\"\n\r
        \txsi:schemaLocation=\"http:\/\/www.w3.org\/2001\/10\/synthesis\n\r
        \thttp:\/\/www.w3.org\/TR\/speech-synthesis11\/synthesis.xsd\"\n\r
        \txml:lang=\"en-US\">\n\r
        \n\r
        \tThe title of the movie is:\n\r
        \t\"La vita è bella\"\n\r
        \t(Life is beautiful),\n\r
        \twhich is directed by Roberto Benigni.\n\r
        <\/speak"

    responses :
      200:
        body:
          application/json:
```

```
      schema: |
       {
        "id": "http://openinterconnect.org/schemas/oic.r.speech.tts.json#",
        "$schema": "http://json-schema.org/draft-04/schema#",
        "title": "Speech Synthesis-TTS",
        "definitions": {
         "oic.r.speech.tts": {
          "type": "object",
          "properties": {
           "utterance": {
            "type": "string",
            "description": "SSML document including the speak body"
           },
           "supportedLanguages": {
            "type": "string",
            "description": "ReadOnly, comma separated list of supported language tags"
           },
           "supportedVoices": {
            "type": "string",
            "description": "ReadOnly, SSML document fragment indicating supported
voices"
           }
          }
         }
        },
        "type": "object",
        "allOf": [
         {"$ref": "oic.core.json#/definitions/oic.core"},
         {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
         {"$ref": "#/definitions/oic.r.speech.tts"}
        ],
        "required": ["utterance"]
       }

      example: |
       {
        "rt":                 "oic.r.speech.tts",
        "id":                 "unique_example_id",
        "utterance":          "SSML Document",
        "supportedLanguages": "en-US, en-GB, fr-CA",
        "supportedVoices":    "<voice gender=\"female\" variant=\"2\"><\/voice>\n\r<voice
name=\"Mike\"><\/voice>"
       }

 post:
  description: |
   Changes the utterance being rendered.
   Example shows a change in language selected.

  body:
   application/json:
    schema: |
     {
      "id": "http://openinterconnect.org/schemas/oic.r.speech.tts.json#",
      "$schema": "http://json-schema.org/draft-04/schema#",
      "title": "Speech Synthesis-TTS",
      "definitions": {
       "oic.r.speech.tts": {
        "type": "object",
        "properties": {
         "utterance": {
          "type": "string",
          "description": "SSML document including the speak body"
         },
         "supportedLanguages": {
          "type": "string",
          "description": "ReadOnly, comma separated list of supported language tags"
```

```
                  },
                  "supportedVoices": {
                    "type": "string",
                    "description": "ReadOnly, SSML document fragment indicating supported voices"
                  }
                }
              }
            },
            "type": "object",
            "allOf": [
              {"$ref": "oic.core.json#/definitions/oic.core"},
              {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
              {"$ref": "#/definitions/oic.r.speech.tts"}
            ],
            "required": ["utterance"]
          }

        example: |
          {
            "rt":         "oic.r.speech.tts",
            "id":         "unique_example_id",
            "utterance":  "SSML Document"
          }

    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.speech.tts.json#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Speech Synthesis-TTS",
                "definitions": {
                  "oic.r.speech.tts": {
                    "type": "object",
                    "properties": {
                      "utterance": {
                        "type": "string",
                        "description": "SSML document including the speak body"
                      },
                      "supportedLanguages": {
                        "type": "string",
                        "description": "ReadOnly, comma separated list of supported language tags"
                      },
                      "supportedVoices": {
                        "type": "string",
                        "description": "ReadOnly, SSML document fragment indicating supported
voices"
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.speech.tts"}
                ],
                "required": ["utterance"]
              }

            example: |
              {
                "rt":         "oic.r.speech.tts",
                "id":         "unique_example_id",
                "utterance":  "SSML Document"
              }

```

6.48.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
utterance	string	yes	Read Write	SSML document including the speak body
supportedLanguages	string		Read Only	Comma Separated List Of Supported Language Tags
supportedVoices	string		Read Only	Ssml Document Fragment Indicating Supported Voices

6.48.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/SpeechTTSResURI		get	post		

6.49 Touch Sensor

6.49.1 Introduction

This resource describes whether touch has been sensed or not. The value is a boolean. A value of True means that touch has been sensed. A value of False means that touch not been sensed.

6.49.2 Example URI

/TouchResURI

6.49.3 Resource Type

The resource type (rt) is defined as: oic.r.sensor.touch.

6.49.4 RAML Definition

```
#%RAML 0.8
title: OICTouchSensor
version: v1.0-20150727
traits:
 - interface :
     queryParameters:
       if:
         enum: ["oic.if.s"]


/TouchResURI:
  description: |
    This resource describes whether touch has been sensed or not.
    The value is a boolean.
    A value of True means that touch has been sensed.
    A value of False means that touch not been sensed.


  is : ['interface']
  get:
    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.sensor.touch.json#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Touch Sensor",
                "definitions": {
                  "oic.r.sensor.touch": {
                    "allOf": [
                      {"$ref": "oic.r.sensor.json#/definitions/oic.r.sensor"}
                    ]
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.sensor.touch"}
                ],
                "required": ["value"]
              }

            example: |
              {
                "rt":    "oic.r.sensor.touch",
                "id":    "unique_example_id",
                "value": true
              }
```

6.49.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
value	boolean	yes	Read Only	True = Sensed, False = Not Sensed.

6.49.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/TouchResURI		get			

6.50 UV Radiation

6.50.1 Introduction

This resource specifies UV radiation measurement. The measurement is the current measured UV Index

6.50.2 Example URI

/UVRadiationResURI

6.50.3 Resource Type

The resource type (rt) is defined as: oic.r.sensor.radiation.uv.

6.50.4 RAML Definition

```
#%RAML 0.8
title: OICUVRadiation
version: v1.0-20150805
traits:
 - interface :
     queryParameters:
       if:
         enum: ["oic.if.s"]


/UVRadiationResURI:
  description: |
    This resource specifies UV radiation measurement.
    The measurement is the current measured UV Index

  is : ['interface']
  get:
    description: |
      Retrieves the current UV Radiation value

    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.sensor.radiation.uv#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "UV Radiation",
                "definitions": {
                  "oic.r.sensor.radiation.uv": {
                    "type": "object",
                    "properties": {
                      "measurement":   {
                        "type": "number",
                        "description": "ReadOnly, the measured UV Index"
                      }
                    }
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.sensor.radiation.uv"}
                ],
                "required": ["measurement"]
              }

            example: |
              {
                "rt":           "oic.r.sensor.radiation.uv",
                "id":           "unique_example_id",
                "measurement":  3.5
              }

```

6.50.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
measurement	number	yes	Read Only	The Measured Uv Index

6.50.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/UVRadiationResURI		get			

6.51 Water Sensor

6.51.1 Introduction

This resource describes whether water has been sensed or not. The value is a boolean. A value of True means that water has been sensed. A value of False means that water not been sensed.

6.51.2 Example URI

/WaterResURI

6.51.3 Resource Type

The resource type (rt) is defined as: oic.r.sensor.water.

6.51.4 RAML Definition

```
#%RAML 0.8
title: OICWaterSensor
version: v1.0-20150727
traits:
 - interface :
     queryParameters:
       if:
         enum: ["oic.if.a"]

/WaterResURI:
  description: |
    This resource describes whether water has been sensed or not.
    The value is a boolean.
    A value of True means that water has been sensed.
    A value of False means that water not been sensed.

  is : ['interface']
  get:
    responses :
      200:
        body:
          application/json:
            schema: |
              {
                "id": "http://openinterconnect.org/schemas/oic.r.sensor.water.json#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "title": "Water Sensor",
                "definitions": {
                  "oic.r.sensor.water": {
                    "allOf": [
                      {"$ref": "oic.r.sensor.json#/definitions/oic.r.sensor"}
                    ]
                  }
                },
                "type": "object",
                "allOf": [
                  {"$ref": "oic.core.json#/definitions/oic.core"},
                  {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
                  {"$ref": "#/definitions/oic.r.sensor.water"}
                ],
                "required": ["value"]
              }

            example: |
              {
                "rt":    "oic.r.sensor.water",
                "id":    "unique_example_id",
                "value": true
              }

```

6.51.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
value	boolean	yes	Read Only	True = Sensed, False = Not Sensed.

6.51.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify

/WaterResURI		get			