



# OIC RESOURCE TYPE SPECIFICATION V1.0.0

Open Interconnect Consortium (OIC)  
[admin@openinterconnect.org](mailto:admin@openinterconnect.org)

3 Legal Disclaimer

4  
5 NOTHING CONTAINED IN THIS DOCUMENT SHALL BE DEEMED AS  
6 GRANTING YOU ANY KIND OF LICENSE IN ITS CONTENT, EITHER  
7 EXPRESSLY OR IMPLIEDLY, OR TO ANY INTELLECTUAL PROPERTY OWNED  
8 OR CONTROLLED BY ANY OF THE AUTHORS OR DEVELOPERS OF THIS  
9 DOCUMENT. THE INFORMATION CONTAINED HEREIN IS PROVIDED ON AN  
10 "AS IS" BASIS, AND TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE  
11 LAW, THE AUTHORS AND DEVELOPERS OF THIS SPECIFICATION HEREBY  
12 DISCLAIM ALL OTHER WARRANTIES AND CONDITIONS, EITHER EXPRESS  
13 OR IMPLIED, STATUTORY OR AT COMMON LAW, INCLUDING, BUT NOT  
14 LIMITED TO, IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS  
15 FOR A PARTICULAR PURPOSE. OPEN INTERCONNECT CONSORTIUM, INC.  
16 FURTHER DISCLAIMS ANY AND ALL WARRANTIES OF NON-INFRINGEMENT,  
17 ACCURACY OR LACK OF VIRUSES.

18  
19 The OIC logo is a trademark of Open Interconnect Consortium, Inc. in the United  
20 States or other countries. \*Other names and brands may be claimed as the  
21 property of others.

22  
23 Copyright © 2015 Open Interconnect Consortium, Inc. All rights reserved.

24  
25 Copying or other form of reproduction and/or distribution of these works are  
26 strictly prohibited

27

28

# CONTENTS

29

30	1	Scope .....	13
31	2	Normative references .....	13
32	3	Terms, definitions, symbols and abbreviations .....	13
33	3.1	Terms and definitions .....	13
34	3.3	Symbols and abbreviations .....	14
35	3.4	Conventions .....	14
36	4	Document conventions and organization .....	14
37	4.1	Notation.....	14
38	4.2	Data types .....	15
39	5	Baseline Model Constructs .....	15
40	5.1	URI.....	15
41	5.2	Interfaces .....	16
42	5.3	RAML definition .....	16
43	5.4	Property definition .....	17
44	5.4.1	Common Properties.....	17
45	5.4.2	Resource Properties.....	17
46	5.4.3	Basic Resource Schema .....	18
47	5.4.4	CRUDN Operation Response Codes .....	20
48	5.4.5	'id' property .....	21
49	5.5	Example Resource Definitions .....	21
50	5.6	Observable Resources .....	24
51	5.7	Composite resources .....	24
52	6	Resource Type definitions .....	25
53	6.1	Air Flow .....	28
54	6.1.1	Introduction .....	28
55	6.1.2	Example URI .....	28
56	6.1.3	Resource Type .....	28
57	6.1.4	RAML Definition .....	28
58	6.1.5	Property Definition .....	31
59	6.1.6	CRUDN behavior.....	32
60	6.2	Air Flow Control.....	32
61	6.2.1	Introduction .....	32
62	6.2.2	Example URI .....	32
63	6.2.3	Resource Type .....	32
64	6.2.4	RAML Definition .....	32
65	6.2.5	Property Definition .....	36
66	6.2.6	CRUDN behavior.....	36
67	6.3	Battery .....	36
68	6.3.1	Introduction .....	36

69	6.3.2	Example URI .....	36
70	6.3.3	Resource Type .....	36
71	6.3.4	RAML Definition .....	36
72	6.3.5	Property Definition .....	37
73	6.3.6	CRUDN behavior .....	37
74	6.4	Binary Switch .....	37
75	6.4.1	Introduction .....	37
76	6.4.2	Example URI .....	37
77	6.4.3	Resource Type .....	37
78	6.4.4	RAML Definition .....	37
79	6.4.5	Property Definition .....	39
80	6.4.6	CRUDN behavior .....	39
81	6.5	Brightness .....	39
82	6.5.1	Introduction .....	39
83	6.5.2	Example URI .....	39
84	6.5.3	Resource Type .....	40
85	6.5.4	RAML Definition .....	40
86	6.5.5	Property Definition .....	42
87	6.5.6	CRUDN behavior .....	42
88	6.6	Colour Chroma .....	42
89	6.6.1	Introduction .....	42
90	6.6.2	Example URI .....	42
91	6.6.3	Resource Type .....	42
92	6.6.4	RAML Definition .....	42
93	6.6.5	Property Definition .....	45
94	6.6.6	CRUDN behavior .....	45
95	6.7	Colour RGB .....	45
96	6.7.1	Introduction .....	45
97	6.7.2	Example URI .....	45
98	6.7.3	Resource Type .....	45
99	6.7.4	RAML Definition .....	45
100	6.7.5	Property Definition .....	47
101	6.7.6	CRUDN behavior .....	48
102	6.8	Dimming .....	48
103	6.8.1	Introduction .....	48
104	6.8.2	Example URI .....	48
105	6.8.3	Resource Type .....	48
106	6.8.4	RAML Definition .....	48
107	6.8.5	Property Definition .....	51
108	6.8.6	CRUDN behavior .....	51
109	6.9	Door .....	51
110	6.9.1	Introduction .....	51
111	6.9.2	Example URI .....	52
112	6.9.3	Resource Type .....	52

113	6.9.4	RAML Definition .....	52
114	6.9.5	Property Definition .....	54
115	6.9.6	CRUDN behavior .....	54
116	6.10	Energy Consumption .....	54
117	6.10.1	Introduction .....	54
118	6.10.2	Example URI .....	54
119	6.10.3	Resource Type .....	54
120	6.10.4	RAML Definition .....	54
121	6.10.5	Property Definition .....	55
122	6.10.6	CRUDN behavior .....	56
123	6.11	Energy Usage .....	56
124	6.11.1	Introduction .....	56
125	6.11.2	Example URI .....	56
126	6.11.3	Resource Type .....	56
127	6.11.4	RAML Definition .....	56
128	6.11.5	CRUDN behavior .....	57
129	6.12	Humidity .....	57
130	6.12.1	Introduction .....	57
131	6.12.2	Example URI .....	57
132	6.12.3	Resource Type .....	57
133	6.12.4	RAML Definition .....	57
134	6.12.5	Property Definition .....	59
135	6.12.6	CRUDN behavior .....	60
136	6.13	Ice Maker .....	60
137	6.13.1	Introduction .....	60
138	6.13.2	Example URI .....	60
139	6.13.3	Resource Type .....	60
140	6.13.4	RAML Definition .....	60
141	6.13.5	Property Definition .....	63
142	6.13.6	CRUDN behavior .....	63
143	6.14	Lock .....	63
144	6.14.1	Introduction .....	63
145	6.14.2	Example URI .....	63
146	6.14.3	Resource Type .....	63
147	6.14.4	RAML Definition .....	63
148	6.14.5	Property Definition .....	66
149	6.14.6	CRUDN behavior .....	66
150	6.15	Lock Code .....	66
151	6.15.1	Introduction .....	66
152	6.15.2	Example URI .....	66
153	6.15.3	Resource Type .....	66
154	6.15.4	RAML Definition .....	66
155	6.15.5	Property Definition .....	68
156	6.15.6	CRUDN behavior .....	68

157	6.16	Mode .....	69
158	6.16.1	Introduction .....	69
159	6.16.2	Example URI .....	69
160	6.16.3	Resource Type .....	69
161	6.16.4	RAML Definition .....	69
162	6.16.5	Property Definition .....	72
163	6.16.6	CRUDN behavior .....	72
164	6.17	Open Level .....	72
165	6.17.1	Introduction .....	72
166	6.17.2	Example URI .....	72
167	6.17.3	Resource Type .....	72
168	6.17.4	RAML Definition .....	72
169	6.17.5	Property Definition .....	75
170	6.17.6	CRUDN behavior .....	76
171	6.18	Operational State .....	76
172	6.18.1	Introduction .....	76
173	6.18.2	Example URI .....	76
174	6.18.3	Resource Type .....	76
175	6.18.4	RAML Definition .....	76
176	6.18.5	Property Definition .....	80
177	6.18.6	CRUDN behavior .....	80
178	6.19	Ramp Time .....	80
179	6.19.1	Introduction .....	80
180	6.19.2	Example URI .....	80
181	6.19.3	Resource Type .....	80
182	6.19.4	RAML Definition .....	80
183	6.19.5	Property Definition .....	83
184	6.19.6	CRUDN behavior .....	83
185	6.20	Refrigeration .....	84
186	6.20.1	Introduction .....	84
187	6.20.2	Example URI .....	84
188	6.20.3	Resource Type .....	84
189	6.20.4	RAML Definition .....	84
190	6.20.5	Property Definition .....	86
191	6.20.6	CRUDN behavior .....	87
192	6.21	Temperature .....	87
193	6.21.1	Introduction .....	87
194	6.21.2	Example URI .....	87
195	6.21.3	Resource Type .....	87
196	6.21.4	RAML Definition .....	87
197	6.21.5	Property Definition .....	91
198	6.21.6	CRUDN behavior .....	91
199	6.22	Time Period .....	91
200	6.22.1	Introduction .....	91

201	6.22.2	Example URI .....	91
202	6.22.3	Resource Type .....	91
203	6.22.4	RAML Definition .....	91
204	6.22.5	Property Definition .....	94
205	6.22.6	CRUDN behavior .....	94
206	6.23	Activity Count .....	94
207	6.23.1	Introduction .....	94
208	6.23.2	Example URI .....	94
209	6.23.3	Resource Type .....	94
210	6.23.4	RAML Definition .....	94
211	6.23.5	Property Definition .....	96
212	6.23.6	CRUDN behavior .....	96
213	6.24	Atmospheric Pressure Sensor .....	96
214	6.24.1	Introduction .....	96
215	6.24.2	Example URI .....	96
216	6.24.3	Resource Type .....	96
217	6.24.4	RAML Definition .....	96
218	6.24.5	Property Definition .....	97
219	6.24.6	CRUDN behavior .....	97
220	6.25	Audio Controls .....	97
221	6.25.1	Introduction .....	97
222	6.25.2	Example URI .....	98
223	6.25.3	Resource Type .....	98
224	6.25.4	RAML Definition .....	98
225	6.25.5	Property Definition .....	100
226	6.25.6	CRUDN behavior .....	100
227	6.26	Auto Focus .....	100
228	6.26.1	Introduction .....	100
229	6.26.2	Example URI .....	100
230	6.26.3	Resource Type .....	100
231	6.26.4	RAML Definition .....	100
232	6.26.5	Property Definition .....	102
233	6.26.6	CRUDN behavior .....	102
234	6.27	Automatic Document Feeder .....	102
235	6.27.1	Introduction .....	102
236	6.27.2	Example URI .....	103
237	6.27.3	Resource Type .....	103
238	6.27.4	RAML Definition .....	103
239	6.27.5	Property Definition .....	104
240	6.27.6	CRUDN behavior .....	104
241	6.28	Button Switch .....	104
242	6.28.1	Introduction .....	104
243	6.28.2	Example URI .....	104
244	6.28.3	Resource Type .....	104

245	6.28.4	RAML Definition .....	104
246	6.28.5	Property Definition .....	105
247	6.28.6	CRUDN behavior .....	105
248	6.29	Carbon Dioxide Sensor .....	105
249	6.29.1	Introduction .....	105
250	6.29.2	Example URI .....	105
251	6.29.3	Resource Type .....	105
252	6.29.4	RAML Definition .....	105
253	6.29.5	Property Definition .....	106
254	6.29.6	CRUDN behavior .....	106
255	6.30	Carbon Monoxide Sensor .....	107
256	6.30.1	Introduction .....	107
257	6.30.2	Example URI .....	107
258	6.30.3	Resource Type .....	107
259	6.30.4	RAML Definition .....	107
260	6.30.5	Property Definition .....	108
261	6.30.6	CRUDN behavior .....	108
262	6.31	Auto White Balance .....	108
263	6.31.1	Introduction .....	108
264	6.31.2	Example URI .....	108
265	6.31.3	Resource Type .....	108
266	6.31.4	RAML Definition .....	108
267	6.31.5	Property Definition .....	110
268	6.31.6	CRUDN behavior .....	110
269	6.32	Colour Saturation .....	110
270	6.32.1	Introduction .....	110
271	6.32.2	Example URI .....	110
272	6.32.3	Resource Type .....	110
273	6.32.4	RAML Definition .....	110
274	6.32.5	Property Definition .....	112
275	6.32.6	CRUDN behavior .....	113
276	6.33	Contact Sensor .....	113
277	6.33.1	Introduction .....	113
278	6.33.2	Example URI .....	113
279	6.33.3	Resource Type .....	113
280	6.33.4	RAML Definition .....	113
281	6.33.5	Property Definition .....	114
282	6.33.6	CRUDN behavior .....	114
283	6.34	Demand Response Load Control (DRLC) .....	114
284	6.34.1	Introduction .....	114
285	6.34.2	Example URI .....	114
286	6.34.3	Resource Type .....	114
287	6.34.4	RAML Definition .....	114
288	6.34.5	Property Definition .....	117



289	6.34.6	CRUDN behavior .....	117
290	6.35	Energy Overload/Circuit Breaker .....	118
291	6.35.1	Introduction .....	118
292	6.35.2	Example URI .....	118
293	6.35.3	Resource Type .....	118
294	6.35.4	RAML Definition .....	118
295	6.35.5	Property Definition .....	119
296	6.35.6	CRUDN behavior .....	119
297	6.36	Generic Sensor .....	119
298	6.36.1	Introduction .....	119
299	6.36.2	Example URI .....	119
300	6.36.3	Resource Type .....	119
301	6.36.4	RAML Definition .....	119
302	6.36.5	Property Definition .....	120
303	6.36.6	CRUDN behavior .....	120
304	6.37	Glass Break Sensor .....	120
305	6.37.1	Introduction .....	120
306	6.37.2	Example URI .....	120
307	6.37.3	Resource Type .....	120
308	6.37.4	RAML Definition .....	120
309	6.37.5	Property Definition .....	121
310	6.37.6	CRUDN behavior .....	121
311	6.38	Heart Rate Zone .....	121
312	6.38.1	Introduction .....	121
313	6.38.2	Example URI .....	121
314	6.38.3	Resource Type .....	122
315	6.38.4	RAML Definition .....	122
316	6.38.5	Property Definition .....	123
317	6.38.6	CRUDN behavior .....	123
318	6.39	Illuminance Sensor .....	123
319	6.39.1	Introduction .....	123
320	6.39.2	Example URI .....	123
321	6.39.3	Resource Type .....	123
322	6.39.4	RAML Definition .....	123
323	6.39.5	Property Definition .....	124
324	6.39.6	CRUDN behavior .....	124
325	6.40	Magnetic Field Direction Sensor .....	124
326	6.40.1	Introduction .....	124
327	6.40.2	Example URI .....	124
328	6.40.3	Resource Type .....	124
329	6.40.4	RAML Definition .....	124
330	6.40.5	Property Definition .....	125
331	6.40.6	CRUDN behavior .....	125
332	6.41	Media .....	125

333	6.41.1	Introduction .....	125
334	6.41.2	Example URI .....	125
335	6.41.3	Resource Type .....	125
336	6.41.4	RAML Definition .....	126
337	6.41.5	Property Definition .....	127
338	6.41.6	CRUDN behavior .....	127
339	6.42	Media Source List .....	127
340	6.42.1	Introduction .....	127
341	6.42.2	Example URI .....	128
342	6.42.3	Resource Type .....	128
343	6.42.4	RAML Definition .....	128
344	6.42.5	Property Definition .....	130
345	6.42.6	CRUDN behavior .....	131
346	6.42.7	Referenced JSON schemas .....	131
347	6.43	Motion Sensor .....	131
348	6.43.1	Introduction .....	131
349	6.43.2	Example URI .....	131
350	6.43.3	Resource Type .....	131
351	6.43.4	RAML Definition .....	131
352	6.43.5	Property Definition .....	132
353	6.43.6	CRUDN behavior .....	132
354	6.44	Night Mode .....	132
355	6.44.1	Introduction .....	132
356	6.44.2	Example URI .....	132
357	6.44.3	Resource Type .....	133
358	6.44.4	RAML Definition .....	133
359	6.44.5	Property Definition .....	135
360	6.44.6	CRUDN behavior .....	135
361	6.45	Presence Sensor .....	135
362	6.45.1	Introduction .....	135
363	6.45.2	Example URI .....	135
364	6.45.3	Resource Type .....	135
365	6.45.4	RAML Definition .....	135
366	6.45.5	Property Definition .....	136
367	6.45.6	CRUDN behavior .....	136
368	6.46	Pan Tilt Zoom Movement .....	136
369	6.46.1	Introduction .....	136
370	6.46.2	Example URI .....	136
371	6.46.3	Resource Type .....	136
372	6.46.4	RAML Definition .....	136
373	6.46.5	Property Definition .....	140
374	6.46.6	CRUDN behavior .....	140
375	6.47	Signal Strength .....	140
376	6.47.1	Introduction .....	140

377	6.47.2	Example URI .....	140
378	6.47.3	Resource Type .....	140
379	6.47.4	RAML Definition .....	140
380	6.47.5	Property Definition .....	141
381	6.47.6	CRUDN behavior .....	141
382	6.48	Speech Synthesis-TTS .....	142
383	6.48.1	Introduction .....	142
384	6.48.2	Example URI .....	142
385	6.48.3	Resource Type .....	142
386	6.48.4	RAML Definition .....	142
387	6.48.5	Property Definition .....	145
388	6.48.6	CRUDN behavior .....	145
389	6.49	Touch Sensor .....	145
390	6.49.1	Introduction .....	145
391	6.49.2	Example URI .....	145
392	6.49.3	Resource Type .....	145
393	6.49.4	RAML Definition .....	145
394	6.49.5	Property Definition .....	146
395	6.49.6	CRUDN behavior .....	146
396	6.50	UV Radiation .....	146
397	6.50.1	Introduction .....	146
398	6.50.2	Example URI .....	146
399	6.50.3	Resource Type .....	146
400	6.50.4	RAML Definition .....	146
401	6.50.5	Property Definition .....	147
402	6.50.6	CRUDN behavior .....	147
403	6.51	Water Sensor .....	147
404	6.51.1	Introduction .....	147
405	6.51.2	Example URI .....	147
406	6.51.3	Resource Type .....	147
407	6.51.4	RAML Definition .....	148
408	6.51.5	Property Definition .....	148
409	6.51.6	CRUDN behavior .....	148
410			
411			

412  
413  
414  
415  
416

## Figures

**No table of figures entries found.**

## Tables

417  
418  
419  
420  
421  
422  
423  
424  
425  
426  
427  
428

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

## 429 **1 Scope**

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

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

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

## 440 **2 Normative references**

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

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

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

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

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

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

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

456 [http://www.cie.co.at/index.php/Publications/index.php?i\\_ca\\_id=435](http://www.cie.co.at/index.php/Publications/index.php?i_ca_id=435)

457

## 458 **3 Terms, definitions, symbols and abbreviations**

### 459 **3.1 Terms and definitions**

#### 460 **3.1.1**

##### 461 **Actuator**

462 OIC Resource with support of the update operation.

#### 463 **3.1.2**

##### 464 **Sensor**

465 OIC Resource without support of the update operation.

466 **3.2**  
467 **3.2.1**  
468 **TBD**  
469 To Be Determined

470 **3.2.2**  
471 **Resource**  
472 Term used as OIC Resource.

473 **3.3 Symbols and abbreviations**

474 **3.3.1**  
475 **CRUDN**  
476 Create Read Update Delete Notify  
477 This is an acronym indicating which operations are possible on the resource.

478 **3.3.2**  
479 **CSV**  
480 Comma Separated Value List  
481 Comma Separated Value List is a construction to have more fields in 1 string separated by  
482 commas. If a value contains a comma then the comma can be escaped by adding “\” in front of  
483 the comma.

484 **3.3.3**  
485 **OIC**  
486 Open Interconnect Consortium  
487 OIC is the standards organization which created and owns this specification.

488 **3.3.4**  
489 **RAML**  
490 RESTful API modelling language  
491 RAML is a simple and succinct way of describing practically-**RESTful APIs**. See **RAML**.

492 **3.3.5**  
493 **REST**  
494 Representational State Transfer  
495 REST is an architecture style for designing networked applications and relies on a stateless,  
496 client-server, cacheable communications protocol.

497 **3.4 Conventions**

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

502 **4 Document conventions and organization**

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

506 For the purposes of this document, the terms and definitions given in OIC Core Specification  
507 apply.

508 **4.1 Notation**

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

511 Required (or shall or mandatory).

512 These basic features shall be implemented to comply with OIC Core Architecture. The  
513 phrases “shall not”, and “PROHIBITED” indicate behaviour that is prohibited, i.e. that if  
514 performed means the implementation is not in compliance.

515 Recommended (or should).

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

523 Allowed (or allowed).

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

527 DEPRECATED

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

534 Conditionally allowed (CA)

535 The definition or behaviour depends on a condition. If the specified condition is met, then the  
536 definition or behaviour is allowed, otherwise it is not allowed.

537 Conditionally required (CR)

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

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

542 Words that are emphasized are printed in *italic*.

## 543 **4.2 Data types**

544 See OIC Core Specification.

## 545 **5 Baseline Model Constructs**

### 546 **5.1 URI**

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

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



552 **5.2 Interfaces**

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

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

559 **5.2.1.1 Retrieve Behaviour**

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

562 **5.2.1.2 Update Behaviour**

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

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

568 An Update request is allowed to omit optional properties.

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

570 **5.3 RAML definition**

571 The RAML definitions used in this document are normative.

572 The RAML definitions are used to describe the payloads of the CRUDN operations on the  
573 specified Resource. The CRUDN operations are defined in the OIC Core Specification. The Core  
574 also specifies additional properties in the payloads of the CRUDN operations. A vendor can't use  
575 the RAML definitions in this document directly to create an implementation, additional properties  
576 defined in the Core specification needs to be added to create a compliant implementation. The  
577 CRUDN operations in this document are defined only for success path scenarios, failure cases  
578 and responses thereon align with the core resource model definitions for CRUDN. This  
579 specification defines a set of '200 class' responses indicating a successful operation; specifics  
580 on the use of these responses are defined in Table 5-6 Return codes behaviour in RAML Table  
581 5-6. Note that the actual values of success and error conditions are defined in the OIC Core  
582 Specification.

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

584 **Table 5-1 Conversion between OIC CRUDN and RAML definitions<sup>1</sup>**

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

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

---

<sup>1</sup> Please refer to OIC Core Specification Table 30 for detailed semantics around the appropriate use of CoAP request methods

587 **5.4 Property definition**

588 **5.4.1 Common Properties**

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

596 Further, should an OIC Client apply the Core defined batch interface (oic.if.b) to a RETRIEVE  
597 operation then the returned set of resource representations shall consist of the resource type  
598 name 'rt' and 'if' Common Properties in addition to the resource properties as defined in Table  
599 5-3 Property definitions of a Resource in the JSON schema.

600 Table 5-2 Common Properties for OIC Resources

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

601

602 **5.4.2 Resource Properties**

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

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

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

610

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

611

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

612

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

### 618 5.4.3 Basic Resource Schema

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

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

626

**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" }
  ]
}

```

628

#### 629 5.4.4 CRUDN Operation Response Codes

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

633

**Table 5-6 Return codes behaviour in RAML**

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

403	<p>Case 1:</p> <p>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.</p> <p>The response payload will include the allowed values for the queryParameter.</p> <p>Case 2:</p> <p>The server could not create or update the resource due to a problem with the provided payload.</p> <p>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).</p>
-----	---

634 **5.4.5 'id' property**

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

638 **5.5 Example Resource Definitions**

639 **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:
application/json:
schema: |
{
    "id": "http://openinterconnect.org/schemas/oic.r.baseResource#",
    "$schema": "http://json-schema.org/schema#",
    "title": "ActuatorExample",
    "definitions": {
        "oic.r.baseResource": {
            "type": "object",
            "properties": {
                "value": { "type": "string" },
                "range": { "type": "string" }
            }
        }
    },
    "type": "object",
    "allOf": [
        {"$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": "ActuatorExample",
    "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:

640

641

**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"
    }

```

642

## 643 5.6 Observable Resources

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

## 646 5.7 Composite resources

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

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

654 The linking is done by using an array of web-links; refer to the OIC Core Specification section  
 655 7.1.6.2 for more details. Note that the example listed below contains a partial schema of this  
 656 definition as it is for descriptive purpose only. The property name of the array is "resources". The  
 657 relationship type shall be "contains", denoting that the composite contains other resources that  
 658 make up the composite resource.

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

661 **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"
              }
            ]
          }

```

662

## 663 6 Resource Type definitions

664

Table 6-1 Alphabetical list of resource types

665

Friendly	Name	Resource Type (rt)	Section
----------	------	--------------------	---------

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

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

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

666

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

669

## 670 **6.1 Air Flow**

### 671 **6.1.1 Introduction**

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

### 676 **6.1.2 Example URI**

677 /AirFlowResURI

### 678 **6.1.3 Resource Type**

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

### 680 **6.1.4 RAML Definition**

```
681 #%RAML 0.8
682 title: OICAirFlow
683 version: v1.0-20150805
684 traits:
685   - interface :
686     queryParameters:
687       if:
688         enum: ["oic.if.a"]
689
690 /AirFlowResURI:
691   description: |
692     This resource describes the properties associated with air flow.
693     The direction is the directionality of the air flow if applicable.
694     Direction values are dependent on the capabilities of the unit.
695     The speed is an integer representing the current speed level for the unit.
696     The range is the min,max values for the speed level.
697
698   is : ['interface']
699   get:
700     description: |
```

```

701     Retrieves the current air flow values.
702
703 responses :
704     200:
705         body:
706             application/json:
707                 schema: |
708                     {
709                         "id": "http://openinterconnect.org/schemas/oic.r.airFlow#",
710                         "$schema": "http://json-schema.org/draft-04/schema#",
711                         "title": "Air Flow",
712                         "definitions": {
713                             "oic.r.airFlow": {
714                                 "type": "object",
715                                 "properties": {
716                                     "direction": {
717                                         "type": "string",
718                                         "description": "Directionality of the air flow"
719                                     },
720                                     "speed": {
721                                         "type": "integer",
722                                         "description": "Current speed level"
723                                     },
724                                     "range": {
725                                         "type": "string",
726                                         "description": "ReadOnly, Min,max values for the speed level"
727                                     }
728                                 }
729                             }
730                         },
731                         "type": "object",
732                         "allOf": [
733                             {"$ref": "oic.core.json#/definitions/oic.core"},
734                             {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
735                             {"$ref": "#/definitions/oic.r.airFlow"}
736                         ],
737                         "required": ["speed"]
738                     }
739
740                 example: |
741                     {
742                         "rt": "oic.r.airFlow",
743                         "id": "unique_example_id",
744                         "direction": "left",
745                         "speed": 5,
746                         "range": "1,7"
747                     }
748
749     post:
750         description: |
751             Sets the current air flow values.
752             Only direction and speed may be set by an update operation.
753
754         body:
755             application/json:
756                 schema: |
757                     {
758                         "id": "http://openinterconnect.org/schemas/oic.r.airFlow#",
759                         "$schema": "http://json-schema.org/draft-04/schema#",
760                         "title": "Air Flow",
761                         "definitions": {
762                             "oic.r.airFlow": {
763                                 "type": "object",
764                                 "properties": {

```

```

765         "direction": {
766             "type": "string",
767             "description": "Directionality of the air flow"
768         },
769         "speed": {
770             "type": "integer",
771             "description": "Current speed level"
772         },
773         "range": {
774             "type": "string",
775             "description": "ReadOnly, Min,max values for the speed level"
776         }
777     }
778 }
779 },
780 "type": "object",
781 "allOf": [
782     {"$ref": "oic.core.json#/definitions/oic.core"},
783     {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
784     {"$ref": "#/definitions/oic.r.airFlow"}
785 ],
786 "required": ["speed"]
787 }
788
789 example: |
790 {
791     "id":          "unique_example_id",
792     "direction":  "right",
793     "speed":      3
794 }
795
796 responses :
797 200:
798   body:
799     application/json:
800       schema: |
801         {
802             "id": "http://openinterconnect.org/schemas/oic.r.airFlow#",
803             "$schema": "http://json-schema.org/draft-04/schema#",
804             "title": "Air Flow",
805             "definitions": {
806                 "oic.r.airFlow": {
807                     "type": "object",
808                     "properties": {
809                         "direction": {
810                             "type": "string",
811                             "description": "Directionality of the air flow"
812                         },
813                         "speed": {
814                             "type": "integer",
815                             "description": "Current speed level"
816                         },
817                         "range": {
818                             "type": "string",
819                             "description": "ReadOnly, Min,max values for the speed level"
820                         }
821                     }
822                 }
823             },
824             "type": "object",
825             "allOf": [
826                 {"$ref": "oic.core.json#/definitions/oic.core"},
827                 {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
828                 {"$ref": "#/definitions/oic.r.airFlow"}
829             ],
830             "required": ["speed"]

```

```

831     }
832
833     example: |
834     {
835         "id":          "unique_example_id",
836         "direction":  "right",
837         "speed":      3
838     }
839
840     403:
841     description: |
842         This response is generated by the OIC Server when the client sends:
843         An update with an invalid property value for direction.
844         An update with an out of range property value for speed.
845         The server responds with the current resource representation.
846
847     body:
848     application/json:
849     schema: |
850     {
851         "id": "http://openinterconnect.org/schemas/oic.r.airFlow#",
852         "$schema": "http://json-schema.org/draft-04/schema#",
853         "title": "Air Flow",
854         "definitions": {
855             "oic.r.airFlow": {
856                 "type": "object",
857                 "properties": {
858                     "direction": {
859                         "type": "string",
860                         "description": "Directionality of the air flow"
861                     },
862                     "speed": {
863                         "type": "integer",
864                         "description": "Current speed level"
865                     },
866                     "range": {
867                         "type": "string",
868                         "description": "ReadOnly, Min,max values for the speed level"
869                     }
870                 }
871             }
872         },
873         "type": "object",
874         "allOf": [
875             {"$ref": "oic.core.json#/definitions/oic.core"},
876             {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
877             {"$ref": "#/definitions/oic.r.airFlow"}
878         ],
879         "required": ["speed"]
880     }
881
882     example: |
883     {
884         "id":          "unique_example_id",
885         "direction":  "right",
886         "speed":      3
887     }
888

```

### 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
-------	--------	--	-----------	------------------------------------

## 890 6.1.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/AirFlowResURI		get	post		

## 891 6.2 Air Flow Control

### 892 6.2.1 Introduction

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

### 896 6.2.2 Example URI

897 /AirFlowControlResURI

### 898 6.2.3 Resource Type

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

### 900 6.2.4 RAML Definition

```

901 #%RAML 0.8
902 title: OICAirFlowControl
903 version: v1.0-20150805
904 traits:
905   - interface-b :
906     queryParameters:
907       if:
908         enum: ["oic.if.b"]
909   - interface-all :
910     queryParameters:
911       if:
912         enum: ["oic.if.ll", "oic.if.b"]
913
914 /AirFlowControlResURI:
915   description: |
916     This resource describes the attributes associated with control of air flow.
917     For example as modelled by a Thermostat (fan), Room A/C or other device.
918     The resource is a composite resource being made up as a collection of:
919     AirFlow Resource
920     BinarySwitch Resource
921
922   is : ['interface-all']
923   get:
924     description: |
925       Retrieves the current air flow control values.
926
927   responses :
928     200:
929       body:
930         application/json:
931           schema: |
932             {
933               "id": "http://openinterconnect.org/schemas/oic.r.airFlowControl#",
934               "$schema": "http://json-schema.org/draft-04/schema#",
935               "title": "Air Flow Control",
936               "definitions": {
937                 "oic.r.airFlowControl": {

```

```

938         "type": "object",
939         "properties": {
940             "airFlowControl": {
941                 "type": "array",
942                 "minItems": 2,
943                 "maxItems": 2,
944                 "items": {
945                     "$ref": "oic.web-link.json#"
946                 }
947             }
948         }
949     },
950     "type": "object",
951     "allOf": [
952         {"$ref": "oic.core.json#/definitions/oic.core"},
953         {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
954         {"$ref": "#/definitions/oic.r.airFlowControl"}
955     ],
956     "required": ["airFlowControl"]
957 }
958
959

```

```

960 example: |
961 {
962     "rt": "oic.r.airFlowControl",
963     "id": "unique_example_id",
964     "airFlowControl": [
965         {
966             "href": "/BinarySwitchResURI",
967             "rel": "contains",
968             "rt": "oic.r.switch.binary",
969             "if": "oic.if.a"
970         },
971         {
972             "href": "/AirFlowResURI",
973             "rel": "contains",
974             "rt": "oic.r.airFlow",
975             "if": "oic.if.a"
976         }
977     ]
978 }
979

```

```

980 post:
981     description: |
982         Sets the current air flow control values using the batch interface
983

```

```

984 body:
985     application/json:
986         schema: |
987             {
988                 "id": "http://openinterconnect.org/schemas/oic.r.airFlowControl#",
989                 "$schema": "http://json-schema.org/draft-04/schema#",
990                 "title": "Air Flow Control",
991                 "definitions": {
992                     "oic.r.airFlowControl": {
993                         "type": "object",
994                         "properties": {
995                             "airFlowControl": {
996                                 "type": "array",
997                                 "items": {
998                                     "anyOf": [
999                                         {"$ref": "oic.r.switch.binary.json#"},
1000                                         {"$ref": "oic.r.airFlow.json#" }
1001                                     ]
1002                                 }
1003                             }
1004                         }
1005                     }
1006                 }
1007             }

```

```

1005     }
1006   },
1007   "type": "object",
1008   "allOf": [
1009     {"$ref": "oic.core.json#/definitions/oic.core"},
1010     {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
1011     {"$ref": "#/definitions/oic.r.airFlowControl"}
1012   ],
1013   "required": ["airFlowControl"]
1014 }
1015
1016 example: |
1017 {
1018   "id": "unique_example_id",
1019   "airFlowControl": [
1020     {
1021       "id": "unique_example_id",
1022       "value": true
1023     },
1024     {
1025       "id": "unique_example_id",
1026       "direction": "right",
1027       "speed": 3
1028     }
1029   ]
1030 }
1031
1032 responses :
1033   200:
1034     body:
1035       application/json:
1036         schema: |
1037           {
1038             "id": "http://openinterconnect.org/schemas/oic.r.airFlowControl#",
1039             "$schema": "http://json-schema.org/draft-04/schema#",
1040             "title": "Air Flow Control",
1041             "definitions": {
1042               "oic.r.airFlowControl": {
1043                 "type": "object",
1044                 "properties": {
1045                   "airFlowControl": {
1046                     "type": "array",
1047                     "items": {
1048                       "anyOf": [
1049                         {"$ref": "oic.r.switch.binary.json#"},
1050                         {"$ref": "oic.r.airFlow.json#"}
1051                       ]
1052                     }
1053                   }
1054                 }
1055               }
1056             },
1057             "type": "object",
1058             "allOf": [
1059               {"$ref": "oic.core.json#/definitions/oic.core"},
1060               {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
1061               {"$ref": "#/definitions/oic.r.airFlowControl"}
1062             ],
1063             "required": ["airFlowControl"]
1064           }
1065
1066 example: |
1067 {
1068   "id": "unique_example_id",
1069   "airFlowControl": [
1070     {

```

```

1071         "id":    "unique_example_id",
1072         "value": true
1073     },
1074     {
1075         "id":    "unique_example_id",
1076         "direction": "right",
1077         "speed":  3
1078     }
1079 ]
1080 }
1081
1082 403:
1083 description: |
1084     This response is generated by the OIC Server when the client sends:
1085     An update with an invalid property value for direction.
1086     An update with an out of range property value for speed.
1087     The server responds with the current resource representation.
1088
1089 body:
1090 application/json:
1091     schema: |
1092         {
1093             "id": "http://openinterconnect.org/schemas/oic.r.airFlowControl#",
1094             "$schema": "http://json-schema.org/draft-04/schema#",
1095             "title": "Air Flow Control",
1096             "definitions": {
1097                 "oic.r.airFlowControl": {
1098                     "type": "object",
1099                     "properties": {
1100                         "airFlowControl": {
1101                             "type": "array",
1102                             "items": {
1103                                 "anyOf": [
1104                                     {"$ref": "oic.r.switch.binary.json#"},
1105                                     {"$ref": "oic.r.airFlow.json#"}
1106                                 ]
1107                             }
1108                         }
1109                     }
1110                 }
1111             },
1112             "type": "object",
1113             "allOf": [
1114                 {"$ref": "oic.core.json#/definitions/oic.core"},
1115                 {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
1116                 {"$ref": "#/definitions/oic.r.airFlowControl"}
1117             ],
1118             "required": ["airFlowControl"]
1119         }
1120
1121     example: |
1122         {
1123             "id":    "unique_example_id",
1124             "airFlowControl": [
1125                 {
1126                     "id":    "unique_example_id",
1127                     "value": true
1128                 },
1129                 {
1130                     "id":    "unique_example_id",
1131                     "direction": "right",
1132                     "speed":  3
1133                 }
1134             ]
1135         }
1136

```

## 1137 6.2.5 Property Definition

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

## 1138 6.2.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/AirFlowControlResURI		get	post		

## 1139 6.3 Battery

### 1140 6.3.1 Introduction

1141 This resource describes a battery function. The charge is an integer showing the current battery  
1142 charge level. The charge is a percentage in the range 0-100. A value of 0 means fully discharged.  
1143 A value of 100 means fully charged.

### 1144 6.3.2 Example URI

1145 /BatteryResURI

### 1146 6.3.3 Resource Type

1147 The resource type (rt) is defined as: oic.r.energy.battery.

### 1148 6.3.4 RAML Definition

```
1149 #%RAML 0.8
1150 title: OICBattery
1151 version: v1.0-20150727
1152 traits:
1153   - interface :
1154     queryParameters:
1155       if:
1156         enum: ["oic.if.s"]
1157
1158 /BatteryResURI:
1159   description: |
1160     This resource describes a battery function.
1161     The charge is an integer showing the current battery charge level.
1162     The charge is a percentage in the range 0-100.
1163     A value of 0 means fully discharged.
1164     A value of 100 means fully charged.
1165
1166   is : ['interface']
1167   get:
1168     description: |
1169       Retrieves the state of the battery.
1170
1171   responses :
1172     200:
1173       body:
1174         application/json:
1175           schema: |
1176             {
1177               "id": "http://openinterconnect.org/schemas/oic.r.energy.battery#",
1178               "$schema": "http://json-schema.org/draft-04/schema#",
1179               "title": "Battery",
1180               "definitions": {
1181                 "oic.r.energy.battery": {
1182                   "type": "object",
```

```

1183         "properties": {
1184             "charge" : {
1185                 "type": "integer",
1186                 "description": "ReadOnly, The current charge percentage."
1187             }
1188         }
1189     },
1190     "type": "object",
1191     "allOf": [
1192         {"$ref": "oic.core.json#/definitions/oic.core"},
1193         {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
1194         {"$ref": "#/definitions/oic.r.energy.battery"}
1195     ],
1196     "required": [ "charge" ]
1197 }
1198
1199
1200 example: |
1201 {
1202     "rt":      "oic.r.energy.battery",
1203     "id":      "unique_example_id",
1204     "charge": 50
1205 }
1206

```

### 1207 6.3.5 Property Definition

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

### 1208 6.3.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/BatteryResURI		get			

## 1209 6.4 Binary Switch

### 1210 6.4.1 Introduction

1211 This resource describes a binary switch (on/off). The value is a boolean. A value of 'true' means  
1212 that the switch is on. A value of 'false' means that the switch is off.

### 1213 6.4.2 Example URI

1214 /BinarySwitchResURI

### 1215 6.4.3 Resource Type

1216 The resource type (rt) is defined as: oic.r.switch.binary.

### 1217 6.4.4 RAML Definition

```

1218 #%RAML 0.8
1219 title: OICBinarySwitch
1220 version: v1.0-20150727
1221 traits:
1222   - interface :
1223       queryParameters:
1224           if:
1225               enum: ["oic.if.a"]
1226
1227 /BinarySwitchResURI:
1228   description: |
1229     This resource describes a binary switch (on/off).
1230     The value is a boolean.
1231     A value of 'true' means that the switch is on.
1232     A value of 'false' means that the switch is off.
1233

```

```

1234     is : ['interface']
1235     get:
1236         responses :
1237             200:
1238                 body:
1239                     application/json:
1240                         schema: |
1241                             {
1242                                 "id": "http://openinterconnect.org/schemas/oic.r.switch.binary#",
1243                                 "$schema": "http://json-schema.org/draft-04/schema#",
1244                                 "title": "Binary Switch",
1245                                 "definitions": {
1246                                     "oic.r.switch.binary": {
1247                                         "type": "object",
1248                                         "properties": {
1249                                             "value": {
1250                                                 "type": "boolean",
1251                                                 "description": "Status of the switch"
1252                                             }
1253                                         }
1254                                     }
1255                                 },
1256                                 "type": "object",
1257                                 "allOf": [
1258                                     {"$ref": "oic.core.json#/definitions/oic.core"},
1259                                     {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
1260                                     {"$ref": "#/definitions/oic.r.switch.binary"}
1261                                 ],
1262                                 "required": [ "value" ]
1263                             }
1264
1265                         example: |
1266                             {
1267                                 "rt":      "oic.r.switch.binary",
1268                                 "id":      "unique_example_id",
1269                                 "value":   false
1270                             }
1271
1272     post:
1273         body:
1274             application/json:
1275                 schema: |
1276                     {
1277                         "id": "http://openinterconnect.org/schemas/oic.r.switch.binary#",
1278                         "$schema": "http://json-schema.org/draft-04/schema#",
1279                         "title": "Binary Switch",
1280                         "definitions": {
1281                             "oic.r.switch.binary": {
1282                                 "type": "object",
1283                                 "properties": {
1284                                     "value": {
1285                                         "type": "boolean",
1286                                         "description": "Status of the switch"
1287                                     }
1288                                 }
1289                             }
1290                         },
1291                         "type": "object",
1292                         "allOf": [
1293                             {"$ref": "oic.core.json#/definitions/oic.core"},
1294                             {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
1295                             {"$ref": "#/definitions/oic.r.switch.binary"}
1296                         ],
1297                         "required": [ "value" ]

```

```

1298     }
1299
1300     example: |
1301     {
1302         "id":      "unique_example_id",
1303         "value": true
1304     }
1305
1306     responses :
1307     200:
1308         body:
1309             application/json:
1310                 schema: |
1311                 {
1312                     "id": "http://openinterconnect.org/schemas/oic.r.switch.binary#",
1313                     "$schema": "http://json-schema.org/draft-04/schema#",
1314                     "title": "Binary Switch",
1315                     "definitions": {
1316                         "oic.r.switch.binary": {
1317                             "type": "object",
1318                             "properties": {
1319                                 "value": {
1320                                     "type": "boolean",
1321                                     "description": "Status of the switch"
1322                                 }
1323                             }
1324                         },
1325                     },
1326                     "type": "object",
1327                     "allOf": [
1328                         { "$ref": "oic.core.json#/definitions/oic.core" },
1329                         { "$ref": "oic.baseResource.json#/definitions/oic.r.baseResource" },
1330                         { "$ref": "#/definitions/oic.r.switch.binary" }
1331                     ],
1332                     "required": [ "value" ]
1333                 }
1334
1335             example: |
1336             {
1337                 "id":      "unique_example_id",
1338                 "value": true
1339             }
1340

```

#### 1341 6.4.5 Property Definition

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

#### 1342 6.4.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/BinarySwitchResURI		get	post		

### 1343 6.5 Brightness

#### 1344 6.5.1 Introduction

1345 This resource describes the brightness of a light or lamp. The brightness percentage is an  
 1346 integer showing the current brightness level. A brightness of 0 is the minimum for the resource. A  
 1347 brightness of 100 is the maximum for the resource.

#### 1348 6.5.2 Example URI

1349 /BrightnessResURI



### 1350 6.5.3 Resource Type

1351 The resource type (rt) is defined as: oic.r.light.brightness.

### 1352 6.5.4 RAML Definition

```
1353 #%RAML 0.8
1354 title: OICBrightness
1355 version: v1.0-20150727
1356 traits:
1357   - interface :
1358     queryParameters:
1359       if:
1360         enum: ["oic.if.a"]
1361
1362 /BrightnessResURI:
1363   description: |
1364     This resource describes the brightness of a light or lamp.
1365     The brightness percentage is an integer showing the current brightness level.
1366     A brightness of 0 is the minimum for the resource.
1367     A brightness of 100 is the maximum for the resource.
1368
1369   is : ['interface']
1370   get:
1371     description: |
1372       Retrieves the current brightness level.
1373
1374   responses :
1375     200:
1376       body:
1377         application/json:
1378           schema: |
1379             {
1380               "id": "http://openinterconnect.org/schemas/oic.r.light.brightness#",
1381               "$schema": "http://json-schema.org/draft-04/schema#",
1382               "title": "Brightness",
1383               "definitions": {
1384                 "oic.r.light.brightness": {
1385                   "type": "object",
1386                   "properties": {
1387                     "brightness": {
1388                       "type": "integer",
1389                       "description": "Current sensed or set value for Brightness"}
1390                 }
1391               },
1392               "type": "object",
1393               "allOf": [
1394                 {"$ref": "oic.core.json#/definitions/oic.core"},
1395                 {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
1396                 {"$ref": "#/definitions/oic.r.light.brightness"}
1397               ],
1398               "required": [ "brightness" ]
1399             }
1400
1401
1402   example: |
1403     {
1404       "rt":          "oic.r.light.brightness",
1405       "id":          "unique_example_id",
1406       "brightness": 50
1407     }
1408
```

```

1409 post:
1410   description: |
1411     Sets the desired brightness level.
1412
1413   body:
1414     application/json:
1415       schema: |
1416         {
1417           "id": "http://openinterconnect.org/schemas/oic.r.light.brightness#",
1418           "$schema": "http://json-schema.org/draft-04/schema#",
1419           "title": "Brightness",
1420           "definitions": {
1421             "oic.r.light.brightness": {
1422               "type": "object",
1423               "properties": {
1424                 "brightness": {
1425                   "type": "integer",
1426                   "description": "Current sensed or set value for Brightness"}
1427             }
1428           },
1429           "type": "object",
1430           "allOf": [
1431             {"$ref": "oic.core.json#/definitions/oic.core"},
1432             {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
1433             {"$ref": "#/definitions/oic.r.light.brightness"}
1434           ],
1435           "required": [ "brightness" ]
1436         }
1437
1438       example: |
1439         {
1440           "id": "unique_example_id",
1441           "brightness": 10
1442         }
1443
1444
1445 responses :
1446   200:
1447     description: |
1448       Indicates that the brightness was changed.
1449       The new brightness level is provided in the response.
1450
1451     body:
1452       application/json:
1453         schema: |
1454           {
1455             "id": "http://openinterconnect.org/schemas/oic.r.light.brightness#",
1456             "$schema": "http://json-schema.org/draft-04/schema#",
1457             "title": "Brightness",
1458             "definitions": {
1459               "oic.r.light.brightness": {
1460                 "type": "object",
1461                 "properties": {
1462                   "brightness": {
1463                     "type": "integer",
1464                     "description": "Current sensed or set value for Brightness"}
1465                 }
1466             },
1467             "type": "object",
1468             "allOf": [
1469               {"$ref": "oic.core.json#/definitions/oic.core"},
1470               {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
1471               {"$ref": "#/definitions/oic.r.light.brightness"}
1472             ]
1473           }

```

```

1473         ],
1474         "required": [ "brightness" ]
1475     }
1476
1477     example: |
1478         {
1479             "id":          "unique_example_id",
1480             "brightness": 10
1481         }
1482

```

## 1483 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

## 1484 6.5.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/BrightnessResURI		get	post		

## 1485 6.6 Colour Chroma

### 1486 6.6.1 Introduction

1487 This resource describes the colour using chroma conventions. Properties are hue, saturation and  
1488 colorspacevalue. Hue and saturation are integer values as defined by the CIECAM02 model  
1489 definition. Colourspacevalue is a CSV of chromaX, chromaY, colourTemperature (X,Y,T).  
1490 ChromaX and chromaY are defined by CIE. ColourTemperature is the Mired color temperature.

### 1491 6.6.2 Example URI

1492 /ColourChromaResURI

### 1493 6.6.3 Resource Type

1494 The resource type (rt) is defined as: oic.r.colour.chroma.

### 1495 6.6.4 RAML Definition

```

1496 #%RAML 0.8
1497 title: OICColourChroma
1498 version: v1.0-20150727
1499 traits:
1500   - interface :
1501       queryParameters:
1502           if:
1503               enum: ["oic.if.a"]
1504
1505 /ColourChromaResURI:
1506     description: |
1507         This resource describes the colour using chroma conventions.
1508         Properties are hue, saturation and colorspacevalue.
1509         Hue and saturation are integer values as defined by the CIECAM02 model definition.
1510         Colourspacevalue is a CSV of chromaX, chromaY, colourTemperature (X,Y,T).
1511         ChromaX and chromaY are defined by CIE.
1512         ColourTemperature is the Mired color temperature.
1513
1514     is : ['interface']
1515     get:
1516         description: |
1517             Provides the colour using chroma conventions.
1518

```

```

1519     responses :
1520         200:
1521             body:
1522                 application/json:
1523                     schema: |
1524                         {
1525                             "id": "http://openinterconnect.org/schemas/oic.r.colour.chroma#",
1526                             "$schema": "http://json-schema.org/draft-04/schema#",
1527                             "title": "Colour Chroma",
1528                             "definitions": {
1529                                 "oic.r.colour.chroma": {
1530                                     "type": "object",
1531                                     "properties": {
1532                                         "hue": {
1533                                             "type": "integer",
1534                                             "description": "Hue as defined by the CIECAM02 model definition"
1535                                         },
1536                                         "saturation": {
1537                                             "type": "integer",
1538                                             "description": "Saturation as defined by the CIECAM02 model definition"
1539                                         },
1540                                         "colourspacevalue": {
1541                                             "type": "string",
1542                                             "description": "CSV of chromaX, chromaY, colourTemperature (X,Y,T).\"
1543                                         }
1544                                     }
1545                                 }
1546                             },
1547                             "type": "object",
1548                             "allOf": [
1549                                 { "$ref": "oic.core.json#/definitions/oic.core" },
1550                                 { "$ref": "oic.baseResource.json#/definitions/oic.r.baseResource" },
1551                                 { "$ref": "#/definitions/oic.r.colour.chroma" }
1552                             ],
1553                             "required": [ "hue", "saturation", "colourspacevalue" ]
1554                         }
1555
1556                     example: |
1557                         {
1558                             "rt": "oic.r.colour.chroma",
1559                             "id": "unique_example_id",
1560                             "hue": 13088,
1561                             "saturation": 212,
1562                             "colourspacevalue": "0.51, 0.41, 467"
1563                         }
1564
1565
1566         post:
1567             description: |
1568                 Sets current colour chroma values
1569
1570             body:
1571                 application/json:
1572                     schema: |
1573                         {
1574                             "id": "http://openinterconnect.org/schemas/oic.r.colour.chroma#",
1575                             "$schema": "http://json-schema.org/draft-04/schema#",
1576                             "title": "Colour Chroma",
1577                             "definitions": {
1578                                 "oic.r.colour.chroma": {
1579                                     "type": "object",
1580                                     "properties": {
1581                                         "hue": {
1582                                             "type": "integer",
1583                                             "description": "Hue as defined by the CIECAM02 model definition"

```

```

1584         },
1585         "saturation": {
1586             "type": "integer",
1587             "description": "Saturation as defined by the CIECAM02 model definition"
1588         },
1589         "colourspacevalue": {
1590             "type": "string",
1591             "description": "CSV of chromaX, chromaY, colourTemperature (X,Y,T).\"
1592         }
1593     }
1594 }
1595 },
1596 "type": "object",
1597 "allOf": [
1598     {"$ref": "oic.core.json#/definitions/oic.core"},
1599     {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
1600     {"$ref": "#/definitions/oic.r.colour.chroma"}
1601 ],
1602 "required": [ "hue", "saturation", "colourspacevalue" ]
1603 }
1604 }
1605
1606 example: |
1607 {
1608     "id": "unique_example_id",
1609     "hue": 13088,
1610     "saturation": 212,
1611     "colourspacevalue": "0.51, 0.41, 467"
1612 }
1613
1614 responses :
1615 200:
1616   body:
1617     application/json:
1618       schema: |
1619         {
1620             "id": "http://openinterconnect.org/schemas/oic.r.colour.chroma#",
1621             "$schema": "http://json-schema.org/draft-04/schema#",
1622             "title": "Colour Chroma",
1623             "definitions": {
1624                 "oic.r.colour.chroma": {
1625                     "type": "object",
1626                     "properties": {
1627                         "hue": {
1628                             "type": "integer",
1629                             "description": "Hue as defined by the CIECAM02 model definition"
1630                         },
1631                         "saturation": {
1632                             "type": "integer",
1633                             "description": "Saturation as defined by the CIECAM02 model definition"
1634                         },
1635                         "colourspacevalue": {
1636                             "type": "string",
1637                             "description": "CSV of chromaX, chromaY, colourTemperature (X,Y,T).\"
1638                         }
1639                     }
1640                 }
1641             },
1642             "type": "object",
1643             "allOf": [
1644                 {"$ref": "oic.core.json#/definitions/oic.core"},
1645                 {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
1646                 {"$ref": "#/definitions/oic.r.colour.chroma"}
1647             ],
1648             "required": [ "hue", "saturation", "colourspacevalue" ]
1649         }

```

```

1650     }
1651
1652     example: |
1653     {
1654         "id":             "unique_example_id",
1655         "hue":            13088,
1656         "saturation":    212,
1657         "colourspacevalue": "0.51, 0.41, 467"
1658     }
1659

```

## 1660 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).

## 1661 6.6.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/ColourChromaResURI		get	post		

## 1662 6.7 Colour RGB

### 1663 6.7.1 Introduction

1664 This resource specifies the actual colour in the RGB space represented as a string. Each colour  
 1665 value is described with a Red, Green, Blue component. These colour values are encoded as  
 1666 comma separated values in the string. The minimum and maximum colour value per component  
 1667 is described by the range value. When the range value is omitted, then the range is [0,255].

### 1668 6.7.2 Example URI

1669 /ColourRGBResURI

### 1670 6.7.3 Resource Type

1671 The resource type (rt) is defined as: oic.r.colour.rgb.

### 1672 6.7.4 RAML Definition

```

1673 #%RAML 0.8
1674 title: OICColourRGB
1675 version: v1.0-20150727
1676 traits:
1677   - interface :
1678     queryParameters:
1679       if:
1680         enum: ["oic.if.a"]
1681
1682 /ColourRGBResURI:
1683   description: |
1684     This resource specifies the actual colour in the RGB space represented as a string.
1685     Each colour value is described with a Red, Green, Blue component.
1686     These colour values are encoded as comma separated values in the string.
1687     The minimum and maximum colour value per component is described by the range value.
1688     When the range value is omitted, then the range is [0,255].
1689
1690   is : ['interface']

```

```

1691  get:
1692      description: |
1693          Retrieves the current colour in RGB.
1694          Value is an CSV of integer values in the order R,G,B.
1695
1696  responses :
1697      200:
1698          body:
1699              application/json:
1700                  schema: |
1701                      {
1702                          "id": "http://openinterconnect.org/schemas/oic.r.colour.rgb#",
1703                          "$schema": "http://json-schema.org/draft-04/schema#",
1704                          "title": "Colour RGB",
1705                          "definitions": {
1706                              "oic.r.colour.rgb": {
1707                                  "type": "object",
1708                                  "properties": {
1709                                      "rgbValue": {
1710                                          "type": "string",
1711                                          "description": "RGB value"
1712                                      },
1713                                      "range": {
1714                                          "type": "string",
1715                                          "description": "min max value of RGB"
1716                                      }
1717                                  },
1718                                  "type": "object",
1719                                  "allOf": [
1720                                      {"$ref": "oic.core.json#/definitions/oic.core"},
1721                                      {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
1722                                      {"$ref": "#/definitions/oic.r.colour.rgb"}
1723                                  ],
1724                                  "required": ["rgbValue"]
1725                              }
1726                          }
1727
1728                  example: |
1729                      {
1730                          "rt": "oic.r.colour.rgb",
1731                          "id": "unique_example_id",
1732                          "rgbValue": "255,255,255",
1733                          "range": "0,255"
1734                      }
1735
1736
1737  post:
1738      description: |
1739          Sets the current colourRGB value
1740
1741      body:
1742          application/json:
1743              schema: |
1744                  {
1745                      "id": "http://openinterconnect.org/schemas/oic.r.colour.rgb#",
1746                      "$schema": "http://json-schema.org/draft-04/schema#",
1747                      "title": "Colour RGB",
1748                      "definitions": {
1749                          "oic.r.colour.rgb": {
1750                              "type": "object",
1751                              "properties": {
1752                                  "rgbValue": {
1753                                      "type": "string",
1754                                      "description": "RGB value"

```

```

1755         },
1756         "range": {
1757             "type": "string",
1758             "description": "min max value of RGB"
1759         }
1760     }
1761 },
1762 "type": "object",
1763 "allof": [
1764     {"$ref": "oic.core.json#/definitions/oic.core"},
1765     {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
1766     {"$ref": "#/definitions/oic.r.colour.rgb"}
1767 ],
1768 "required": ["rgbValue"]
1769 }
1770
1771
1772 example: |
1773 {
1774     "id": "unique_example_id",
1775     "rgbValue": "255,0,0"
1776 }
1777
1778 responses :
1779 200:
1780 body:
1781 application/json:
1782 schema: |
1783 {
1784     "id": "http://openinterconnect.org/schemas/oic.r.colour.rgb#",
1785     "$schema": "http://json-schema.org/draft-04/schema#",
1786     "title": "Colour RGB",
1787     "definitions": {
1788         "oic.r.colour.rgb": {
1789             "type": "object",
1790             "properties": {
1791                 "rgbValue": {
1792                     "type": "string",
1793                     "description": "RGB value"
1794                 },
1795                 "range": {
1796                     "type": "string",
1797                     "description": "min max value of RGB"
1798                 }
1799             }
1800         }
1801     },
1802     "type": "object",
1803     "allof": [
1804         {"$ref": "oic.core.json#/definitions/oic.core"},
1805         {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
1806         {"$ref": "#/definitions/oic.r.colour.rgb"}
1807     ],
1808     "required": ["rgbValue"]
1809 }
1810
1811 example: |
1812 {
1813     "id": "unique_example_id",
1814     "rgbValue": "255,0,0"
1815 }
1816

```

## 1817 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

1818 **6.7.6 CRUDN behavior**

Resource	Create	Read	Update	Delete	Notify
/ColourRGBResURI		get	post		

1819 **6.8 Dimming**

1820 **6.8.1 Introduction**

1821 This resource describes a dimming function. The value is an integer showing the current  
 1822 dimming level. The step is the increment between dimmer values. The range is the maximum and  
 1823 minimum values for the dimming value. If the range is omitted [0,100] is assumed. A value of 0  
 1824 means total dimming; a value of 100 means no dimming.

1825 **6.8.2 Example URI**

1826 /DimmingResURI

1827 **6.8.3 Resource Type**

1828 The resource type (rt) is defined as: oic.r.light.dimming.

1829 **6.8.4 RAML Definition**

```

1830 #%RAML 0.8
1831 title: OICDimming
1832 version: v1.0-20150727
1833 traits:
1834   - interface :
1835     queryParameters:
1836       if:
1837         enum: ["oic.if.a"]
1838
1839 /DimmingResURI:
1840   description: |
1841     This resource describes a dimming function.
1842     The value is an integer showing the current dimming level.
1843     The step is the increment between dimmer values.
1844     The range is the maximum and minimum values for the dimming value.
1845     If the range is omitted [0,100] is assumed.
1846     A value of 0 means total dimming; a value of 100 means no dimming.
1847
1848   is : ['interface']
1849   get:
1850     description: |
1851       Retrieves the current dimming level.
1852
1853   responses :
1854     200:
1855       body:
1856         application/json:
1857           schema: |
1858             {
1859               "id": "http://openinterconnect.org/schemas/oic.r.light.dimming#",
1860               "$schema": "http://json-schema.org/draft-04/schema#",
1861               "title": "Dimming",
1862               "definitions": {
1863                 "oic.r.light.dimming": {
1864                   "type": "object",
1865                   "properties": {

```

```

1866         "dimmingSetting": {
1867             "type": "integer",
1868             "description": "Current dimming value"
1869         },
1870         "step": {
1871             "type": "integer",
1872             "description": "ReadOnly, step increment for dimming values"
1873         },
1874         "range": {
1875             "type": "string",
1876             "description": "ReadOnly, Min and Max values for the dimming setting"
1877         }
1878     }
1879 }
1880 },
1881 "type": "object",
1882 "allOf": [
1883     {"$ref": "oic.core.json#/definitions/oic.core"},
1884     {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
1885     {"$ref": "#/definitions/oic.r.light.dimming"}
1886 ],
1887 "required": ["dimmingSetting"]
1888 }
1889

```

```

1890 example: |
1891 {
1892     "rt":          "oic.r.light.dimming",
1893     "id":          "unique_example_id",
1894     "dimmingSetting": 30,
1895     "step":        5,
1896     "range":       "0,100"
1897 }
1898

```

1899 **post:**

```

1900 description: |
1901     Sets the desired dimming level.
1902

```

1903 **body:**  
1904 **application/json:**

```

1905 schema: |
1906 {
1907     "id": "http://openinterconnect.org/schemas/oic.r.light.dimming#",
1908     "$schema": "http://json-schema.org/draft-04/schema#",
1909     "title": "Dimming",
1910     "definitions": {
1911         "oic.r.light.dimming": {
1912             "type": "object",
1913             "properties": {
1914                 "dimmingSetting": {
1915                     "type": "integer",
1916                     "description": "Current dimming value"
1917                 },
1918                 "step": {
1919                     "type": "integer",
1920                     "description": "ReadOnly, step increment for dimming values"
1921                 },
1922                 "range": {
1923                     "type": "string",
1924                     "description": "ReadOnly, Min and Max values for the dimming setting"
1925                 }
1926             }
1927         }
1928     },
1929     "type": "object",
1930     "allOf": [
1931         {"$ref": "oic.core.json#/definitions/oic.core"},
1932         {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},

```

```

1933         {"$ref": "#/definitions/oic.r.light.dimming"}
1934     ],
1935     "required": ["dimmingSetting"]
1936 }
1937
1938 example: |
1939     {
1940         "id":                "unique_example_id",
1941         "dimmingSetting": 40
1942     }
1943
1944 responses :
1945     200:
1946         description: |
1947             Indicates that the dimming was changed.
1948             The new dimming level is provided in the response.
1949
1950     body:
1951         application/json:
1952             schema: |
1953                 {
1954                     "id": "http://openinterconnect.org/schemas/oic.r.light.dimming#",
1955                     "$schema": "http://json-schema.org/draft-04/schema#",
1956                     "title": "Dimming",
1957                     "definitions": {
1958                         "oic.r.light.dimming": {
1959                             "type": "object",
1960                             "properties": {
1961                                 "dimmingSetting": {
1962                                     "type": "integer",
1963                                     "description": "Current dimming value"
1964                                 },
1965                                 "step": {
1966                                     "type": "integer",
1967                                     "description": "ReadOnly, step increment for dimming values"
1968                                 },
1969                                 "range": {
1970                                     "type": "string",
1971                                     "description": "ReadOnly, Min and Max values for the dimming setting"
1972                                 }
1973                             }
1974                         }
1975                     },
1976                     "type": "object",
1977                     "allOf": [
1978                         {"$ref": "oic.core.json#/definitions/oic.core"},
1979                         {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
1980                         {"$ref": "#/definitions/oic.r.light.dimming"}
1981                     ],
1982                     "required": ["dimmingSetting"]
1983                 }
1984
1985             example: |
1986                 {
1987                     "id":                "unique_example_id",
1988                     "dimmingSetting": 40
1989                 }
1990
1991     403:
1992         description: |
1993             This response is generated by the OIC Server when the client sends:
1994             An update with an out of range property value for dimmingSetting.
1995             The server responds with the current resource representation.
1996

```

```

1997     body:
1998         application/json:
1999             schema: |
2000                 {
2001                     "id": "http://openinterconnect.org/schemas/oic.r.light.dimming#",
2002                     "$schema": "http://json-schema.org/draft-04/schema#",
2003                     "title": "Dimming",
2004                     "definitions": {
2005                         "oic.r.light.dimming": {
2006                             "type": "object",
2007                             "properties": {
2008                                 "dimmingSetting": {
2009                                     "type": "integer",
2010                                     "description": "Current dimming value"
2011                                 },
2012                                 "step": {
2013                                     "type": "integer",
2014                                     "description": "ReadOnly, step increment for dimming values"
2015                                 },
2016                                 "range": {
2017                                     "type": "string",
2018                                     "description": "ReadOnly, Min and Max values for the dimming setting"
2019                                 }
2020                             }
2021                         }
2022                     },
2023                     "type": "object",
2024                     "allOf": [
2025                         {"$ref": "oic.core.json#/definitions/oic.core"},
2026                         {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
2027                         {"$ref": "#/definitions/oic.r.light.dimming"}
2028                     ],
2029                     "required": ["dimmingSetting"]
2030                 }
2031
2032             example: |
2033                 {
2034                     "id": "unique_example_id",
2035                     "dimmingSetting": 40
2036                 }
2037

```

## 2038 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

## 2039 6.8.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/DimmingResURI		get	post		

## 2040 6.9 Door

### 2041 6.9.1 Introduction

2042 This resource describes a door. A door is modelled by means of openState (Open/Closed).  
 2043 openDuration (ISO 8601 Time). openAlarm (boolean). For openState, the value 'Open' indicates  
 2044 the door is open. The value 'Closed' indicates the door is closed. The type of openDuration is an  
 2045 ISO 8601 Time encoded string. The openAlarm value 'true' indicates that the open alarm is  
 2046 active. The openAlarm value 'false' indicates that open alarm is not active.

```

2047 6.9.2 Example URI
2048 /DoorResURI
2049 6.9.3 Resource Type
2050 The resource type (rt) is defined as: oic.r.door.
2051 6.9.4 RAML Definition
2052 #%RAML 0.8
2053 title: OICDoor
2054 version: v1.0-20150727
2055 traits:
2056 - interface :
2057   queryParameters:
2058     if:
2059       enum: ["oic.if.a"]
2060
2061 /DoorResURI:
2062   description: |
2063     This resource describes a door.
2064     A door is modelled by means of
2065     openState (Open/Closed).
2066     openDuration (ISO 8601 Time).
2067     openAlarm (boolean).
2068     For openState, the value 'Open' indicates the door is open.
2069     The value 'Closed' indicates the door is closed.
2070     The type of openDuration is an ISO 8601 Time encoded string.
2071     The openAlarm value 'true' indicates that the open alarm is active.
2072     The openAlarm value 'false' indicates that open alarm is not active.
2073
2074   is : ['interface']
2075   get:
2076     description: |
2077       retrieves the state of the Door.
2078
2079   responses :
2080     200:
2081       body:
2082         application/json:
2083           schema: |
2084             {
2085               "id": "http://openinterconnect.org/schemas/oic.r.door#",
2086               "$schema": "http://json-schema.org/draft-04/schema#",
2087               "title": "Door",
2088               "definitions": {
2089                 "oic.r.door": {
2090                   "type": "object",
2091                   "properties": {
2092                     "openState": {
2093                       "enum": ["Open","Closed"],
2094                       "description": "ReadOnly, The state of the door (open or closed)"
2095                     },
2096                     "openDuration": {
2097                       "type": "string",
2098                       "description": "ReadOnly, The time duration the door has been open"
2099                     },
2100                     "openAlarm": {
2101                       "type": "boolean",
2102                       "description": "The state of the door open alarm"
2103                     }
2104                   }
2105                 }
2106             }

```

```

2106         },
2107         "type": "object",
2108         "allOf": [
2109             {"$ref": "oic.core.json#/definitions/oic.core"},
2110             {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
2111             {"$ref": "#/definitions/oic.r.door"}
2112         ],
2113         "required": ["openState"]
2114     }
2115
2116     example: |
2117     {
2118         "rt" :           "oic.r.door",
2119         "id":            "unique_example_id",
2120         "openState":     "Open",
2121         "openDuration": "P0Y0M0DT2H25M5S",
2122         "openAlarm":    true
2123     }
2124
2125     post:
2126     description: |
2127         Sets the current Door properties.
2128         The only property that can be set as part of an update operation is
2129         the openAlarm.
2130         This can be made active (true) or inactive (false)
2131
2132     body:
2133     application/json:
2134     schema: |
2135     {
2136         "id": "http://openinterconnect.org/schemas/oic.r.door#",
2137         "$schema": "http://json-schema.org/draft-04/schema#",
2138         "title": "Door",
2139         "definitions": {
2140             "oic.r.door": {
2141                 "type": "object",
2142                 "properties": {
2143                     "openAlarm": {
2144                         "type": "boolean",
2145                         "description": "The state of the door open alarm"
2146                     }
2147                 }
2148             }
2149         },
2150         "type": "object",
2151         "allOf": [
2152             {"$ref": "oic.core.json#/definitions/oic.core"},
2153             {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
2154             {"$ref": "#/definitions/oic.r.door"}
2155         ]
2156     }
2157
2158     example: |
2159     {
2160         "id":            "unique_example_id",
2161         "openAlarm":    false
2162     }
2163
2164     responses :
2165     200:
2166     body:
2167     application/json:
2168     schema: |

```

```

2169     {
2170       "id": "http://openinterconnect.org/schemas/oic.r.door#",
2171       "$schema": "http://json-schema.org/draft-04/schema#",
2172       "title": "Door",
2173       "definitions": {
2174         "oic.r.door": {
2175           "type": "object",
2176           "properties": {
2177             "openAlarm": {
2178               "type": "boolean",
2179               "description": "The state of the door open alarm"
2180             }
2181           }
2182         }
2183       },
2184       "type": "object",
2185       "allOf": [
2186         {"$ref": "oic.core.json#/definitions/oic.core"},
2187         {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
2188         {"$ref": "#/definitions/oic.r.door"}
2189       ]
2190     }
2191
2192     example: |
2193       {
2194         "id": "unique_example_id",
2195         "openAlarm": false
2196       }
2197

```

## 2198 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

## 2199 6.9.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/DoorResURI		get	post		

## 2200 6.10 Energy Consumption

### 2201 6.10.1 Introduction

2202 This resource describes the energy consumed by the device since power up. It provides the  
 2203 instantaneous power draw of the device at the time the resource was queried. The power value  
 2204 is in Watts [W]. The energy value is in Watt Hours [Wh].

### 2205 6.10.2 Example URI

2206 /EnergyConsumptionResURI

### 2207 6.10.3 Resource Type

2208 The resource type (rt) is defined as: oic.r.energy.consumption.

### 2209 6.10.4 RAML Definition

```

2210 #%RAML 0.8
2211 title: OICEnergyConsumption
2212 version: v1.0-20150727
2213 traits:
2214   - interface :
2215     queryParameters:

```

```

2216         if:
2217             enum: ["oic.if.s"]
2218
2219 /EnergyConsumptionResURI:
2220     description: |
2221         This resource describes the energy consumed by the device since power up.
2222         It provides the instantaneous power draw of the device at the time the resource was queried.
2223         The power value is in Watts [W].
2224         The energy value is in Watt Hours [Wh].
2225
2226     is : ['interface']
2227
2228     get:
2229         description: |
2230             Provides the current power draw and cumulative energy usage.
2231
2232     responses :
2233         200:
2234             body:
2235                 application/json:
2236                     schema: |
2237                         {
2238                             "id": "http://openinterconnect.org/schemas/oic.r.energy.consumption#",
2239                             "$schema": "http://json-schema.org/draft-04/schema#",
2240                             "title": "Energy Consumption",
2241                             "definitions": {
2242                                 "oic.r.energy.consumption": {
2243                                     "type": "object",
2244                                     "properties": {
2245                                         "power": {
2246                                             "type": "number",
2247                                             "description": "ReadOnly, Instantaneous Power"
2248                                         },
2249                                         "energy": {
2250                                             "type": "number",
2251                                             "description": "ReadOnly, Energy consumed"
2252                                         }
2253                                     },
2254                                     "type": "object",
2255                                     "allOf": [
2256                                         {"$ref": "oic.core.json#/definitions/oic.core"},
2257                                         {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
2258                                         {"$ref": "#/definitions/oic.r.energy.consumption"}
2259                                     ],
2260                                     "required": ["power", "energy"]
2261                                 }
2262                             }
2263
2264                     example: |
2265                         {
2266                             "rt": "oic.r.energy.consumption",
2267                             "id": "unique_example_id",
2268                             "power": 2000,
2269                             "energy": 3500
2270                         }
2271

```

### 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



2272 **6.10.6 CRUDN behavior**

Resource	Create	Read	Update	Delete	Notify
/EnergyConsumptionResURI		get			

2273 **6.11 Energy Usage**

2274 **6.11.1 Introduction**

2275 This resource describes an energy usage query. The values in the start and stop time strings are  
2276 encoded according to the rules defined in ISO 8601 and equate to time period over which the  
2277 query applies. The energy consumption is the separately defined OIC resource.

2278 **6.11.2 Example URI**

2279 /EnergyUsageResURI

2280 **6.11.3 Resource Type**

2281 The resource type (rt) is defined as: oic.r.energy.usage.

2282 **6.11.4 RAML Definition**

```
2283 #%RAML 0.8
2284 title: OICEnergyUsage
2285 version: v1.1-20150805
2286 traits:
2287   - interface :
2288       queryParameters:
2289           if:
2290               enum: ["oic.if.ll", "oic.if.b"]
2291
2292 /EnergyUsageResURI:
2293   description: |
2294     This resource describes an energy usage query.
2295     The values in the start and stop time strings are encoded according
2296     to the rules defined in ISO 8601 and equate to time period over which
2297     the query applies.
2298     The energy consumption is the separately defined OIC resource.
2299
2300   is : ['interface']
2301   get:
2302     description: |
2303       Retrieves the energy usage information as a composite of consumption over time.
2304
2305   responses :
2306     200:
2307       body:
2308         application/json:
2309           schema: |
2310             {
2311               "id": "http://openinterconnect.org/schemas/oic.r.energy.usage#",
2312               "$schema": "http://json-schema.org/draft-04/schema#",
2313               "title": "Energy Usage",
2314               "definitions": {
2315                 "oic.r.energy.usage": {
2316                   "type": "object",
2317                   "properties": {
2318                     "resources": {
2319                       "type": "array",
2320                       "minItems": 2,
2321                       "maxItems": 2,
2322                       "items": {
2323                         "$ref": "oic.web-link.json#"

```

```

2324         }
2325     }
2326 }
2327 }
2328 },
2329 "type": "object",
2330 "allof": [
2331     {"$ref": "oic.core.json#/definitions/oic.core"},
2332     {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
2333     {"$ref": "#/definitions/oic.r.energy.usage"}
2334 ],
2335 "required": ["resources"]
2336 }
2337
2338 example: |
2339 {
2340     "rt": "oic.r.energy.usage",
2341     "id": "unique_example_id",
2342     "resources": [
2343         {
2344             "href": "/TimeIntervalResURI",
2345             "rel": "contains",
2346             "rt": "oic.r.time.period",
2347             "if": "oic.if.a"
2348         },
2349         {
2350             "href": "/EnergyConsumptionResURI",
2351             "rel": "contains",
2352             "rt": "oic.r.energy.consumption",
2353             "if": "oic.if.s"
2354         }
2355     ]
2356 }
2357

```

2358 **6.11.5 CRUDN behavior**

Resource	Create	Read	Update	Delete	Notify
/EnergyUsageResURI		get			

2359 **6.12 Humidity**

2360 **6.12.1 Introduction**

2361 This resource describes a sensed or desired humidity. The value humidity is an integer  
 2362 describing the percentage measured relative humidity. The value desiredHumidity is an integer  
 2363 showing the desired target relative humidity.

2364 **6.12.2 Example URI**

2365 /HumidityResURI

2366 **6.12.3 Resource Type**

2367 The resource type (rt) is defined as: oic.r.humidity.

2368 **6.12.4 RAML Definition**

```

2369 #%RAML 0.8
2370 title: OICHumidity
2371 version: v1.0-20150727
2372 traits:
2373   - interface :
2374       queryParameters:
2375           if:
2376               enum: ["oic.if.a", "oic.if.s"]

```

2377  
 2378 /HumidityResURI:

```

2379 description: |
2380     This resource describes a sensed or desired humidity.
2381     The value humidity is an integer describing the percentage measured relative humidity.
2382     The value desiredHumidity is an integer showing the desired target relative humidity.
2383
2384 is : ['interface']
2385
2386 get:
2387     description: |
2388         Retrieves the current (relative) humidity level.
2389
2390     responses :
2391         200:
2392             body:
2393                 application/json:
2394                     schema: |
2395                         {
2396                             "id": "http://openinterconnect.org/schemas/oic.r.humidity#",
2397                             "$schema": "http://json-schema.org/draft-04/schema#",
2398                             "title": "Humidity",
2399                             "definitions": {
2400                                 "oic.r.humidity": {
2401                                     "type": "object",
2402                                     "properties": {
2403                                         "humidity": {
2404                                             "type": "integer",
2405                                             "description": "ReadOnly, Current sensed value for Humidity"
2406                                         },
2407                                         "desiredHumidity": {
2408                                             "type": "integer",
2409                                             "description": "Desired value for Humidity"
2410                                         }
2411                                     }
2412                                 },
2413                                 "type": "object",
2414                                 "allOf": [
2415                                     {"$ref": "oic.core.json#/definitions/oic.core"},
2416                                     {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
2417                                     {"$ref": "#/definitions/oic.r.humidity"}
2418                                 ],
2419                                 "required": ["humidity"]
2420                             }
2421
2422                     example: |
2423                         {
2424                             "rt": "oic.r.humidity",
2425                             "id": "unique_example_id",
2426                             "humidity": 40,
2427                             "desiredHumidity": 40
2428                         }
2429
2430 post:
2431     description: |
2432         Sets the desired relative humidity level.
2433
2434     body:
2435         application/json:
2436             schema: |
2437                 {
2438                     "id": "http://openinterconnect.org/schemas/oic.r.humidity#",
2439                     "$schema": "http://json-schema.org/draft-04/schema#",
2440                     "title": "Humidity",
2441                     "definitions": {

```

```

2442     "oic.r.humidity": {
2443         "type": "object",
2444         "properties": {
2445             "desiredHumidity": {
2446                 "type": "integer",
2447                 "description": "Desired value for Humidity"
2448             }
2449         }
2450     },
2451 },
2452 "type": "object",
2453 "allOf": [
2454     {"$ref": "oic.core.json#/definitions/oic.core"},
2455     {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
2456     {"$ref": "#/definitions/oic.r.humidity"}
2457 ]
2458 }
2459
2460 example: |
2461 {
2462     "id":                "unique_example_id",
2463     "desiredHumidity" : 45
2464 }
2465
2466 responses :
2467 200:
2468     description: |
2469         Indicates that the relative humidity level was changed.
2470         The new relative humidity level is provided in the response.
2471
2472 body:
2473 application/json:
2474     schema: |
2475     {
2476         "id": "http://openinterconnect.org/schemas/oic.r.humidity#",
2477         "$schema": "http://json-schema.org/draft-04/schema#",
2478         "title": "Humidity",
2479         "definitions": {
2480             "oic.r.humidity": {
2481                 "type": "object",
2482                 "properties": {
2483                     "desiredHumidity": {
2484                         "type": "integer",
2485                         "description": "Desired value for Humidity"
2486                     }
2487                 }
2488             },
2489         },
2490         "type": "object",
2491         "allOf": [
2492             {"$ref": "oic.core.json#/definitions/oic.core"},
2493             {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
2494             {"$ref": "#/definitions/oic.r.humidity"}
2495         ]
2496     }
2497
2498     example: |
2499     {
2500         "id":                "unique_example_id",
2501         "desiredHumidity": 45
2502     }
2503

```

### 2504 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

2505 **6.12.6 CRUDN behavior**

Resource	Create	Read	Update	Delete	Notify
/HumidityResURI		get	post		

2506 **6.13 Ice Maker**

2507 **6.13.1 Introduction**

2508 This resource describes an Ice Maker. The status is a string containing a value from the set of  
 2509 possible ice maker statuses. The possible statuses are defined by the enumeration [on, off, full]  
 2510 A status of 'on' means that the Ice Maker is operating. A status of 'off' means that the Ice Maker  
 2511 is not operating. A status of 'full' means that the ice collection bin is full (Ice Maker is operating).

2512 **6.13.2 Example URI**

2513 /IceMakerResURI

2514 **6.13.3 Resource Type**

2515 The resource type (rt) is defined as: oic.r.iceMaker.

2516 **6.13.4 RAML Definition**

```

2517 #%RAML 0.8
2518 title: OICIceMaker
2519 version: v1.0-20150727
2520 traits:
2521   - interface :
2522     queryParameters:
2523       if:
2524         enum: ["oic.if.a"]
2525
2526 /IceMakerResURI:
2527   description: |
2528     This resource describes an Ice Maker.
2529     The status is a string containing a value from the set of possible ice maker statuses.
2530     The possible statuses are defined by the enumeration [on, off, full]
2531     A status of 'on' means that the Ice Maker is operating.
2532     A status of 'off' means that the Ice Maker is not operating.
2533     A status of 'full' means that the ice collection bin is full (Ice Maker is operating).
2534
2535   is : ['interface']
2536   get:
2537     description: |
2538       Retrieves the current Ice Maker status.
2539
2540   responses :
2541     200:
2542       body:
2543         application/json:
2544           schema: |
2545             {
2546               "id": "http://openinterconnect.org/schemas/oic.r.iceMaker#",
2547               "$schema": "http://json-schema.org/draft-04/schema#",
2548               "title": "Ice Maker",
2549               "definitions": {
2550                 "oic.r.iceMaker": {

```

```

2551         "type": "object",
2552         "properties": {
2553             "status": {
2554                 "enum": ["on","off","full"],
2555                 "description": "Status of the Ice Maker"
2556             }
2557         }
2558     },
2559 },
2560 "type": "object",
2561 "allOf": [
2562     {"$ref": "oic.core.json#/definitions/oic.core"},
2563     {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
2564     {"$ref": "#/definitions/oic.r.iceMaker"}
2565 ],
2566 "required": ["status"]
2567 }
2568
2569 example: |
2570 {
2571     "rt":      "oic.r.iceMaker",
2572     "id":      "unique_example_id",
2573     "status":  "on"
2574 }
2575
2576 post:
2577 description: |
2578     Sets the desired Ice Maker status.
2579     Only valid settings for status in a Post shall be [on,off].
2580
2581 body:
2582 application/json:
2583 schema: |
2584 {
2585     "id": "http://openinterconnect.org/schemas/oic.r.iceMaker#",
2586     "$schema": "http://json-schema.org/draft-04/schema#",
2587     "title": "Ice Maker",
2588     "definitions": {
2589         "oic.r.iceMaker": {
2590             "type": "object",
2591             "properties": {
2592                 "status": {
2593                     "enum": ["on","off"],
2594                     "description": "Set the status of the Ice Maker"
2595                 }
2596             }
2597         }
2598     },
2599     "type": "object",
2600     "allOf": [
2601         {"$ref": "oic.core.json#/definitions/oic.core"},
2602         {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
2603         {"$ref": "#/definitions/oic.r.iceMaker"}
2604     ],
2605     "required": ["status"]
2606 }
2607
2608 example: |
2609 {
2610     "id":      "unique_example_id",
2611     "status":  "off"
2612 }
2613
2614 responses :
2615 200:

```

```

2616     description: |
2617         Indicates that the Ice Maker status was changed.
2618         The new status is provided in the response.
2619
2620     body:
2621         application/json:
2622             schema: |
2623                 {
2624                     "id": "http://openinterconnect.org/schemas/oic.r.iceMaker#",
2625                     "$schema": "http://json-schema.org/draft-04/schema#",
2626                     "title": "Ice Maker",
2627                     "definitions": {
2628                         "oic.r.iceMaker": {
2629                             "type": "object",
2630                             "properties": {
2631                                 "status": {
2632                                     "enum": ["on","off"],
2633                                     "description": "Set the status of the Ice Maker"
2634                                 }
2635                             }
2636                         }
2637                     },
2638                     "type": "object",
2639                     "allOf": [
2640                         {"$ref": "oic.core.json#/definitions/oic.core"},
2641                         {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
2642                         {"$ref": "#/definitions/oic.r.iceMaker"}
2643                     ],
2644                     "required": ["status"]
2645                 }
2646
2647             example: |
2648                 {
2649                     "id": "unique_example_id",
2650                     "status": "off"
2651                 }
2652
2653     403:
2654         description: |
2655             This response is generated by the OIC Server when the client sends:
2656             An update with an invalid property value for status.
2657             The server responds with the current resource representation.
2658
2659         body:
2660             application/json:
2661                 schema: |
2662                     {
2663                         "id": "http://openinterconnect.org/schemas/oic.r.iceMaker#",
2664                         "$schema": "http://json-schema.org/draft-04/schema#",
2665                         "title": "Ice Maker",
2666                         "definitions": {
2667                             "oic.r.iceMaker": {
2668                                 "type": "object",
2669                                 "properties": {
2670                                     "status": {
2671                                         "enum": ["on","off"],
2672                                         "description": "Set the status of the Ice Maker"
2673                                     }
2674                                 }
2675                             }
2676                         },
2677                         "type": "object",
2678                         "allOf": [
2679                             {"$ref": "oic.core.json#/definitions/oic.core"},
2680                             {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},

```

```

2681         {"$ref": "#/definitions/oic.r.iceMaker"}
2682     },
2683     "required": ["status"]
2684 }
2685
2686     example: |
2687     {
2688         "id":      "unique_example_id",
2689         "status": "off"
2690     }
2691

```

### 2692 6.13.5 Property Definition

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

### 2693 6.13.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/IceMakerResURI		get	post		

## 2694 6.14 Lock

### 2695 6.14.1 Introduction

2696 Resource describing a lock. For the type of lockState, the value 'Locked' indicates that the door  
 2697 is Locked. The value 'Unlocked' indicates that the door is Unlocked.

### 2698 6.14.2 Example URI

2699 /LockStatusResURI

### 2700 6.14.3 Resource Type

2701 The resource type (rt) is defined as: oic.r.lock.status.

### 2702 6.14.4 RAML Definition

```

2703 #%RAML 0.8
2704 title: OICLock
2705 version: v1.0-20150727
2706 traits:
2707   - interface :
2708       queryParameters:
2709           if:
2710               enum: ["oic.if.a"]
2711
2712 /LockStatusResURI:
2713     description: |
2714         Resource describing a lock.
2715         For the type of lockState, the value 'Locked' indicates that the door is Locked.
2716         The value 'Unlocked' indicates that the door is Unlocked.
2717
2718     is : ['interface']
2719     get:
2720         description: |
2721             Retrieves the state of the lock.
2722
2723     responses :
2724         200:
2725             body:
2726                 application/json:
2727                 schema: |

```



```

2728     {
2729     "$id": "http://openinterconnect.org/schemas/oic.r.lock.status#",
2730     "$schema": "http://json-schema.org/draft-04/schema#",
2731     "title": "Lock",
2732     "definitions": {
2733       "oic.r.lock.status": {
2734         "type": "object",
2735         "properties": {
2736           "lockState": {
2737             "type": "string",
2738             "enum": ["Locked", "Unlocked"],
2739             "description": "State of the lock."
2740           }
2741         }
2742       }
2743     },
2744     "type": "object",
2745     "allOf": [
2746       {"$ref": "oic.core.json#/definitions/oic.core"},
2747       {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
2748       {"$ref": "#/definitions/oic.r.lock.status"}
2749     ],
2750     "required": ["lockState"]
2751   }
2752

```

```

2753   example: |
2754     {
2755       "rt":           "oic.r.lock.status",
2756       "id":           "unique_example_id",
2757       "lockState":   "Locked"
2758     }
2759

```

post:

```

2761   description: |
2762     Sets the current lock state.
2763

```

body:

```

2765   application/json:
2766     schema: |
2767       {
2768         "id": "http://openinterconnect.org/schemas/oic.r.lock.status#",
2769         "$schema": "http://json-schema.org/draft-04/schema#",
2770         "title": "Lock",
2771         "definitions": {
2772           "oic.r.lock.status": {
2773             "type": "object",
2774             "properties": {
2775               "lockState": {
2776                 "type": "string",
2777                 "enum": ["Locked", "Unlocked"],
2778                 "description": "State of the lock."
2779               }
2780             }
2781           }
2782         },
2783         "type": "object",
2784         "allOf": [
2785           {"$ref": "oic.core.json#/definitions/oic.core"},
2786           {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
2787           {"$ref": "#/definitions/oic.r.lock.status"}
2788         ],
2789         "required": ["lockState"]
2790       }
2791

```

```

2792   example: |

```

```

2793     {
2794         "id": "unique_example_id",
2795         "lockState": "Unlocked"
2796     }
2797
2798 responses :
2799     200:
2800         body:
2801             application/json:
2802                 schema: |
2803                     {
2804                         "id": "http://openinterconnect.org/schemas/oic.r.lock.status#",
2805                         "$schema": "http://json-schema.org/draft-04/schema#",
2806                         "title": "Lock",
2807                         "definitions": {
2808                             "oic.r.lock.status": {
2809                                 "type": "object",
2810                                 "properties": {
2811                                     "lockState": {
2812                                         "type": "string",
2813                                         "enum": ["Locked", "Unlocked"],
2814                                         "description": "State of the lock."
2815                                     }
2816                                 }
2817                             }
2818                         },
2819                         "type": "object",
2820                         "allOf": [
2821                             {"$ref": "oic.core.json#/definitions/oic.core"},
2822                             {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
2823                             {"$ref": "#/definitions/oic.r.lock.status"}
2824                         ],
2825                         "required": ["lockState"]
2826                     }
2827
2828                 example: |
2829                     {
2830                         "id": "unique_example_id",
2831                         "lockState": "Unlocked"
2832                     }
2833
2834     403:
2835         description: |
2836             This response is generated by the OIC Server when the client sends:
2837             An update with an invalid property value for lockState.
2838             The server responds with the current resource representation.
2839
2840         body:
2841             application/json:
2842                 schema: |
2843                     {
2844                         "id": "http://openinterconnect.org/schemas/oic.r.lock.status#",
2845                         "$schema": "http://json-schema.org/draft-04/schema#",
2846                         "title": "Lock",
2847                         "definitions": {
2848                             "oic.r.lock.status": {
2849                                 "type": "object",
2850                                 "properties": {
2851                                     "lockState": {
2852                                         "type": "string",
2853                                         "enum": ["Locked", "Unlocked"],
2854                                         "description": "State of the lock."
2855                                     }
2856                                 }
2857                             }
2858                         }
2859                     }

```

```

2857     }
2858   },
2859   "type": "object",
2860   "allOf": [
2861     {"$ref": "oic.core.json#/definitions/oic.core"},
2862     {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
2863     {"$ref": "#/definitions/oic.r.lock.status"}
2864   ],
2865   "required": ["lockState"]
2866 }
2867
2868 example: |
2869 {
2870   "lockState": "Unlocked"
2871 }
2872

```

## 2873 6.14.5 Property Definition

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

## 2874 6.14.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/LockStatusResURI		get	post		

## 2875 6.15 Lock Code

### 2876 6.15.1 Introduction

2877 Resource describing a lock code. The lockCodeList is an array of possible codes that may be  
 2878 associated with a lock. These are all presented as strings.

### 2879 6.15.2 Example URI

2880 /LockCodeResURI

### 2881 6.15.3 Resource Type

2882 The resource type (rt) is defined as: oic.r.lock.code.

### 2883 6.15.4 RAML Definition

```

2884 #%RAML 0.8
2885 title: OICLockCode
2886 version: v1.0-20150727
2887 traits:
2888   - interface :
2889       queryParameters:
2890         if:
2891           enum: ["oic.if.a"]
2892
2893 /LockCodeResURI:
2894   description: |
2895     Resource describing a lock code.
2896     The lockCodeList is an array of possible codes that may be associated with a lock.
2897     These are all presented as strings.
2898
2899   is : ['interface']
2900   get:
2901     description: |
2902       Retrieves the current lock code values.
2903
2904   responses :

```

```

2905     200:
2906     body:
2907         application/json:
2908         schema: |
2909             {
2910                 "id": "http://openinterconnect.org/schemas/oic.r.lock.code#",
2911                 "$schema": "http://json-schema.org/draft-04/schema#",
2912                 "title": "Lock Code",
2913                 "definitions": {
2914                     "oic.r.lock.code": {
2915                         "type": "object",
2916                         "properties": {
2917                             "lockCodeList" : {
2918                                 "type": "array",
2919                                 "items": {
2920                                     "type": "string",
2921                                     "description": "Value for the lock code"
2922                                 }
2923                             }
2924                         }
2925                     }
2926                 },
2927                 "type": "object",
2928                 "allOf": [
2929                     {"$ref": "oic.core.json#/definitions/oic.core"},
2930                     {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
2931                     {"$ref": "#/definitions/oic.r.lock.code"}
2932                 ],
2933                 "required": ["lockCodeList"]
2934             }
2935
2936         example: |
2937             {
2938                 "rt": "oic.r.lock.code",
2939                 "id": "unique_example_id",
2940                 "lockCodeList": ["012345", "112233"]
2941             }
2942
2943     post:
2944     description: |
2945         Updates the current lock code values.
2946
2947     body:
2948         application/json:
2949         schema: |
2950             {
2951                 "id": "http://openinterconnect.org/schemas/oic.r.lock.code#",
2952                 "$schema": "http://json-schema.org/draft-04/schema#",
2953                 "title": "Lock Code",
2954                 "definitions": {
2955                     "oic.r.lock.code": {
2956                         "type": "object",
2957                         "properties": {
2958                             "lockCodeList" : {
2959                                 "type": "array",
2960                                 "items": {
2961                                     "type": "string",
2962                                     "description": "Value for the lock code"
2963                                 }
2964                             }
2965                         }
2966                     }
2967                 },
2968                 "type": "object",
2969                 "allOf": [

```

```

2970         {"$ref": "oic.core.json#/definitions/oic.core"},
2971         {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
2972         {"$ref": "#/definitions/oic.r.lock.code"}
2973     ],
2974     "required": ["lockCodeList"]
2975 }
2976
2977 example: |
2978     {
2979         "id":          "unique_example_id",
2980         "lockCodeList": ["543210", "332211"]
2981     }
2982
2983 responses :
2984     200:
2985         body:
2986             application/json:
2987                 schema: |
2988                     {
2989                         "id": "http://openinterconnect.org/schemas/oic.r.lock.code#",
2990                         "$schema": "http://json-schema.org/draft-04/schema#",
2991                         "title": "Lock Code",
2992                         "definitions": {
2993                             "oic.r.lock.code": {
2994                                 "type": "object",
2995                                 "properties": {
2996                                     "lockCodeList" : {
2997                                         "type": "array",
2998                                         "items": {
2999                                             "type": "string",
3000                                             "description": "Value for the lock code"
3001                                         }
3002                                     }
3003                                 }
3004                             },
3005                             "type": "object",
3006                             "allOf": [
3007                                 {"$ref": "oic.core.json#/definitions/oic.core"},
3008                                 {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
3009                                 {"$ref": "#/definitions/oic.r.lock.code"}
3010                             ],
3011                             "required": ["lockCodeList"]
3012                         }
3013                     }
3014
3015                 example: |
3016                     {
3017                         "id":          "unique_example_id",
3018                         "lockCodeList": ["543210", "332211"]
3019                     }
3020

```

### 3021 6.15.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
lockCodeList	array	yes		

### 3022 6.15.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/LockCodeResURI		get	post		

## 3023 6.16 Mode

### 3024 6.16.1 Introduction

3025 This resource describes the modes of operation that a device can provide. The mode can be  
3026 read or set. The supportedModes is a comma separated list of possible modes the device  
3027 supports. The modes are a comma separated list of the currently active mode(s).

### 3028 6.16.2 Example URI

3029 /ModeResURI

### 3030 6.16.3 Resource Type

3031 The resource type (rt) is defined as: oic.r.mode.

### 3032 6.16.4 RAML Definition

```
3033 #%RAML 0.8
3034 title: OICMode
3035 version: v1.0-20150727
3036 traits:
3037   - interface :
3038       queryParameters:
3039           if:
3040               enum: ["oic.if.a"]
3041
3042 /ModeResURI:
3043   description: |
3044     This resource describes the modes of operation that a device can provide.
3045     The mode can be read or set.
3046     The supportedModes is a comma separated list of possible modes the device supports.
3047     The modes are a comma separated list of the currently active mode(s).
3048
3049   is : ['interface']
3050   get:
3051     description: |
3052       Retrieves the current mode.
3053
3054   responses :
3055     200:
3056       body:
3057         application/json:
3058           schema: |
3059             {
3060               "id": "http://openinterconnect.org/schemas/oic.r.mode#",
3061               "$schema": "http://json-schema.org/draft-04/schema#",
3062               "title": "Mode",
3063               "definitions": {
3064                 "oic.r.mode": {
3065                   "type": "object",
3066                   "properties": {
3067                     "supportedModes": {
3068                       "type": "string",
3069                       "description": "ReadOnly, Comma separated list of possible modes the device
3070 supports."
3071                     },
3072                     "modes": {
3073                       "type": "string",
3074                       "description": "Comma separated list of the currently active mode(s)"
3075                     }
3076                   }
3077                 }
3078             },
```

```

3079         "type": "object",
3080         "allOf": [
3081             {"$ref": "oic.core.json#/definitions/oic.core"},
3082             {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
3083             {"$ref": "#/definitions/oic.r.mode"}
3084         ],
3085         "required": ["supportedModes", "modes"]
3086     }
3087
3088     example: |
3089         {
3090             "rt":          "oic.r.mode",
3091             "id":          "unique_example_id",
3092             "supportedModes": "home,away,quiet,sleep",
3093             "modes":       "quiet"
3094         }
3095
3096     post:
3097         description: |
3098             Sets the desired mode.
3099
3100         body:
3101             application/json:
3102                 schema: |
3103                     {
3104                         "id": "http://openinterconnect.org/schemas/oic.r.mode#",
3105                         "$schema": "http://json-schema.org/draft-04/schema#",
3106                         "title": "Mode",
3107                         "definitions": {
3108                             "oic.r.mode": {
3109                                 "type": "object",
3110                                 "properties": {
3111                                     "modes": {
3112                                         "type": "string",
3113                                         "description": "Desired mode"
3114                                     }
3115                                 }
3116                             }
3117                         },
3118                         "type": "object",
3119                         "allOf": [
3120                             {"$ref": "oic.core.json#/definitions/oic.core"},
3121                             {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
3122                             {"$ref": "#/definitions/oic.r.mode"}
3123                         ],
3124                         "required": ["modes"]
3125                     }
3126
3127                 example: |
3128                     {
3129                         "id": "unique_example_id",
3130                         "modes": "sleep"
3131                     }
3132
3133         responses :
3134             200:
3135                 body:
3136                     application/json:
3137                         schema: |
3138                             {
3139                                 "id": "http://openinterconnect.org/schemas/oic.r.mode#",
3140                                 "$schema": "http://json-schema.org/draft-04/schema#",
3141                                 "title": "Mode",

```

```

3142         "definitions": {
3143             "oic.r.mode": {
3144                 "type": "object",
3145                 "properties": {
3146                     "modes": {
3147                         "type": "string",
3148                         "description": "Desired mode"
3149                     }
3150                 }
3151             }
3152         },
3153         "type": "object",
3154         "allOf": [
3155             {"$ref": "oic.core.json#/definitions/oic.core"},
3156             {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
3157             {"$ref": "#/definitions/oic.r.mode"}
3158         ],
3159         "required": ["modes"]
3160     }
3161
3162     example: |
3163     {
3164         "id": "unique_example_id",
3165         "modes": "sleep"
3166     }
3167
3168     403:
3169     description: |
3170         This response is generated by the OIC Server when the client sends:
3171         An update with an value for mode that is not found in supportedModes.
3172         The server responds with the current resource representation.
3173
3174     body:
3175     application/json:
3176     schema: |
3177     {
3178         "id": "http://openinterconnect.org/schemas/oic.r.mode#",
3179         "$schema": "http://json-schema.org/draft-04/schema#",
3180         "title": "Mode",
3181         "definitions": {
3182             "oic.r.mode": {
3183                 "type": "object",
3184                 "properties": {
3185                     "supportedModes": {
3186                         "type": "string",
3187                         "description": "ReadOnly, Comma separated list of possible modes the device
3188 supports."
3189                     },
3190                     "modes": {
3191                         "type": "string",
3192                         "description": "Comma separated list of the currently active mode(s)"
3193                     }
3194                 }
3195             }
3196         },
3197         "type": "object",
3198         "allOf": [
3199             {"$ref": "oic.core.json#/definitions/oic.core"},
3200             {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
3201             {"$ref": "#/definitions/oic.r.mode"}
3202         ],
3203         "required": ["supportedModes", "modes"]
3204     }
3205
3206     example: |

```



```

3207     {
3208         "id": "unique_example_id",
3209         "supportedModes": "home,away,quiet,sleep",
3210         "modes": "quiet"
3211     }
3212

```

### 3213 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)

### 3214 6.16.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/ModeResURI		get	post		

## 3215 6.17 Open Level

### 3216 6.17.1 Introduction

3217 This resource describes how open or ajar an entity such as a window, door, blind or shutter is.  
3218 The openLevel can be read (acting as a sensor). The openLevel can also be set (acting as an  
3219 actuator). The openLevel is device dependent across the range provided. If no range is provided  
3220 then 0 to 100 is assumed. 0 means closed, 100 means fully open. If a range is provided the  
3221 lower bound=closed, upper bound=open. The increment is the step between possible values; if  
3222 not provide 1 is assumed.

### 3223 6.17.2 Example URI

3224 /OpenLevelResURI

### 3225 6.17.3 Resource Type

3226 The resource type (rt) is defined as: oic.r.openLevel.

### 3227 6.17.4 RAML Definition

```

3228 #%RAML 0.8
3229 title: OICOpenLevel
3230 version: v1.0-20150727
3231 traits:
3232   - interface :
3233       queryParameters:
3234           if:
3235               enum: ["oic.if.a"]
3236
3237 /OpenLevelResURI:
3238   description: |
3239       This resource describes how open or ajar an entity such as a window, door, blind or shutter is.
3240       The openLevel can be read (acting as a sensor).
3241       The openLevel can also be set (acting as an actuator).
3242       The openLevel is device dependent across the range provided.
3243       If no range is provided then 0 to 100 is assumed.
3244       0 means closed, 100 means fully open.
3245       If a range is provided the lower bound=closed, upper bound=open.
3246       The increment is the step between possible values; if not provide 1 is assumed.
3247
3248   is : ['interface']
3249   get:
3250       description: |

```

```

3251     Retrieves the current openLevel.
3252
3253     responses :
3254         200:
3255             body:
3256                 application/json:
3257                 schema: |
3258                     {
3259                         "id": "http://openinterconnect.org/schemas/oic.r.openLevel#",
3260                         "$schema": "http://json-schema.org/draft-04/schema#",
3261                         "title": "Open Level",
3262                         "definitions": {
3263                             "oic.r.openLevel": {
3264                                 "type": "object",
3265                                 "properties": {
3266                                     "openLevel": {
3267                                         "type": "integer",
3268                                         "description": "How open or ajar the entity is"
3269                                     },
3270                                     "increment": {
3271                                         "type": "integer",
3272                                         "description": "ReadOnly, The step between possible values"
3273                                     },
3274                                     "range": {
3275                                         "type": "string",
3276                                         "description": "ReadOnly, Lower bound=closed, Upper bound=open"
3277                                     }
3278                                 }
3279                             }
3280                         },
3281                         "type": "object",
3282                         "allOf": [
3283                             {"$ref": "oic.core.json#/definitions/oic.core"},
3284                             {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
3285                             {"$ref": "#/definitions/oic.r.openLevel"}
3286                         ],
3287                         "required": ["openLevel"]
3288                     }
3289
3290                 example: |
3291                     {
3292                         "rt": "oic.r.openLevel",
3293                         "id": "unique_example_id",
3294                         "openLevel": 50,
3295                         "increment": 2,
3296                         "range": "0,100"
3297                     }
3298
3299         post:
3300             description: |
3301                 Sets the desired openLevel.
3302
3303             body:
3304                 application/json:
3305                 schema: |
3306                     {
3307                         "id": "http://openinterconnect.org/schemas/oic.r.openLevel#",
3308                         "$schema": "http://json-schema.org/draft-04/schema#",
3309                         "title": "Open Level",
3310                         "definitions": {
3311                             "oic.r.openLevel": {
3312                                 "type": "object",
3313                                 "properties": {
3314                                     "openLevel": {

```

```

3315         "type": "integer",
3316         "description": "How open or ajar the entity is"
3317     },
3318     "increment": {
3319         "type": "integer",
3320         "description": "ReadOnly, The step between possible values"
3321     },
3322     "range": {
3323         "type": "string",
3324         "description": "ReadOnly, Lower bound=closed, Upper bound=open"
3325     }
3326 }
3327 },
3328 },
3329 "type": "object",
3330 "allOf": [
3331     {"$ref": "oic.core.json#/definitions/oic.core"},
3332     {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
3333     {"$ref": "#/definitions/oic.r.openLevel"}
3334 ],
3335 "required": ["openLevel"]
3336 }
3337
3338 example: |
3339     {
3340         "id": "unique_example_id",
3341         "openLevel": 0
3342     }
3343
3344 responses :
3345     200:
3346         body:
3347             application/json:
3348                 schema: |
3349                     {
3350                         "id": "http://openinterconnect.org/schemas/oic.r.openLevel#",
3351                         "$schema": "http://json-schema.org/draft-04/schema#",
3352                         "title": "Open Level",
3353                         "definitions": {
3354                             "oic.r.openLevel": {
3355                                 "type": "object",
3356                                 "properties": {
3357                                     "openLevel": {
3358                                         "type": "integer",
3359                                         "description": "How open or ajar the entity is"
3360                                     },
3361                                     "increment": {
3362                                         "type": "integer",
3363                                         "description": "ReadOnly, The step between possible values"
3364                                     },
3365                                     "range": {
3366                                         "type": "string",
3367                                         "description": "ReadOnly, Lower bound=closed, Upper bound=open"
3368                                     }
3369                                 }
3370                             }
3371                         },
3372                         "type": "object",
3373                         "allOf": [
3374                             {"$ref": "oic.core.json#/definitions/oic.core"},
3375                             {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
3376                             {"$ref": "#/definitions/oic.r.openLevel"}
3377                         ],
3378                         "required": ["openLevel"]
3379                     }
3380
3381 example: |

```

```

3382     {
3383         "id": "unique_example_id",
3384         "openLevel": 0
3385     }
3386
3387 403:
3388     description: |
3389         This response is generated by the OIC Server when the client sends:
3390         An update with an out of range property value for openLevel.
3391         The server responds with the current resource representation.
3392
3393     body:
3394         application/json:
3395             schema: |
3396                 {
3397                     "id": "http://openinterconnect.org/schemas/oic.r.openLevel#",
3398                     "$schema": "http://json-schema.org/draft-04/schema#",
3399                     "title": "Open Level",
3400                     "definitions": {
3401                         "oic.r.openLevel": {
3402                             "type": "object",
3403                             "properties": {
3404                                 "openLevel": {
3405                                     "type": "integer",
3406                                     "description": "How open or ajar the entity is"
3407                                 },
3408                                 "increment": {
3409                                     "type": "integer",
3410                                     "description": "ReadOnly, The step between possible values"
3411                                 },
3412                                 "range": {
3413                                     "type": "string",
3414                                     "description": "ReadOnly, Lower bound=closed, Upper bound=open"
3415                                 }
3416                             }
3417                         }
3418                     },
3419                     "type": "object",
3420                     "allOf": [
3421                         {"$ref": "oic.core.json#/definitions/oic.core"},
3422                         {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
3423                         {"$ref": "#/definitions/oic.r.openLevel"}
3424                     ],
3425                     "required": ["openLevel"]
3426                 }
3427
3428             example: |
3429                 {
3430                     "id": "unique_example_id",
3431                     "openLevel": 50,
3432                     "increment": 2,
3433                     "range": "0,100"
3434                 }
3435

```

### 3436 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

3437 **6.17.6 CRUDN behavior**

Resource	Create	Read	Update	Delete	Notify
/OpenLevelResURI		get	post		

3438 **6.18 Operational State**

3439 **6.18.1 Introduction**

3440 This resource describes the operational and job states on a device. The states can be read or  
3441 set, setting indicates a desired state. A device may reject an attempt to set a state that would  
3442 result in adverse operational characteristics. The machineStates is a comma separated list of the  
3443 possible operational states. The currentMachineState is the current state of operation of the  
3444 device. The jobStates is a comma separated list of the possible job states. The currentJobState  
3445 is the currently active jobState. The runningTime is the ISO8601 encoded elapsed time in the  
3446 current operational state. The remainingTime is the ISO8601 encoded time till completion of the  
3447 current operational state. The progressPercentage is the percentage completeness of the current  
3448 jobState.

3449 **6.18.2 Example URI**

3450 /OperationalStateResURI

3451 **6.18.3 Resource Type**

3452 The resource type (rt) is defined as: oic.r.operational.state.

3453 **6.18.4 RAML Definition**

```
3454 #%RAML 0.8
3455 title: OICOperation
3456 version: v1.0-20150805
3457 traits:
3458   - interface :
3459     queryParameters:
3460       if:
3461         enum: ["oic.if.a"]
3462
3463 /OperationalStateResURI:
3464   description: |
3465     This resource describes the operational and job states on a device.
3466     The states can be read or set, setting indicates a desired state.
3467     A device may reject an attempt to set a state that would result
3468     in adverse operational characteristics.
3469     The machineStates is a comma separated list of the possible operational states.
3470     The currentMachineState is the current state of operation of the device.
3471     The jobStates is a comma separated list of the possible job states.
3472     The currentJobState is the currently active jobState.
3473     The runningTime is the ISO8601 encoded elapsed time in the current operational state.
3474     The remainingTime is the ISO8601 encoded time till completion of the current operational state.
3475     The progressPercentage is the percentage completeness of the current jobState.
3476
3477   is : ['interface']
3478   get:
3479     description: |
3480       Retrieves the current operational and job states.
3481
3482   responses :
3483     200:
3484       body:
3485         application/json:
3486           schema: |
```

```

3487     {
3488     "id": "http://openinterconnect.org/schemas/oic.r.operational.state#",
3489     "$schema": "http://json-schema.org/draft-04/schema#",
3490     "title": "Operational State",
3491     "definitions": {
3492     "oic.r.operational.state": {
3493     "type": "object",
3494     "properties": {
3495     "machineStates": {
3496     "type": "string",
3497     "description": "ReadOnly, Comma separated list of the possible operational
3498 states."
3499     },
3500     "currentMachineState": {
3501     "type": "string",
3502     "description": "Current state of operation of the device."
3503     },
3504     "jobStates": {
3505     "type": "string",
3506     "description": "ReadOnly, Comma separate list of the possible job states."
3507     },
3508     "currentJobState": {
3509     "type": "string",
3510     "description": "Currently active jobState"
3511     },
3512     "runningTime": {
3513     "type": "string",
3514     "description": "ReadOnly, Elapsed time in the current operational state"
3515     },
3516     "remainingTime": {
3517     "type": "string",
3518     "description": "ReadOnly, Time till completion of the current operational
3519 state"
3520     },
3521     "progressPercentage": {
3522     "type": "integer",
3523     "description": "ReadOnly, Percentage completeness of the current jobState"
3524     }
3525     }
3526     },
3527     "type": "object",
3528     "allOf": [
3529     {"$ref": "oic.core.json#/definitions/oic.core"},
3530     {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
3531     {"$ref": "#/definitions/oic.r.operational.state"}
3532     ],
3533     "required": ["machineStates", "currentMachineState"]
3534     }
3535     }
3536
3537     example: |
3538     {
3539     "rt": "oic.r.operational.state",
3540     "id": "unique_example_id",
3541     "machineStates": "pause, standby, ready, active",
3542     "currentMachineState": "active",
3543     "jobStates": "pre-wash, wash, rinse, spin, dry, air-dry, wrinkle-
3544 prevent",
3545     "currentJobState": "rinse",
3546     "runningTime": "PT15M20S",
3547     "remainingTime": "PT10M40S",
3548     "progressPercentage": 75
3549     }
3550
3551     post:
3552     description: |
3553     Sets the desired operational or job state.
3554

```

```

3555 body:
3556   application/json:
3557     schema: |
3558       {
3559         "id": "http://openinterconnect.org/schemas/oic.r.operational.state#",
3560         "$schema": "http://json-schema.org/draft-04/schema#",
3561         "title": "Operational State",
3562         "definitions": {
3563           "oic.r.operational.state": {
3564             "type": "object",
3565             "properties": {
3566               "currentMachineState": {
3567                 "type": "string",
3568                 "description": "Current state of operation of the device."
3569               },
3570               "currentJobState": {
3571                 "type": "string",
3572                 "description": "Currently active jobState"
3573               }
3574             }
3575           },
3576         },
3577         "type": "object",
3578         "allOf": [
3579           {"$ref": "oic.core.json#/definitions/oic.core"},
3580           {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
3581           {"$ref": "#/definitions/oic.r.operational.state"}
3582         ]
3583       }
3584
3585     example: |
3586       {
3587         "id": "unique_example_id",
3588         "currentMachineState": "pause",
3589         "currentJobState": "wash"
3590       }
3591
3592 responses :
3593   200:
3594     body:
3595       application/json:
3596         schema: |
3597           {
3598             "id": "http://openinterconnect.org/schemas/oic.r.operational.state#",
3599             "$schema": "http://json-schema.org/draft-04/schema#",
3600             "title": "Operational State",
3601             "definitions": {
3602               "oic.r.operational.state": {
3603                 "type": "object",
3604                 "properties": {
3605                   "currentMachineState": {
3606                     "type": "string",
3607                     "description": "Current state of operation of the device."
3608                   },
3609                   "currentJobState": {
3610                     "type": "string",
3611                     "description": "Currently active jobState"
3612                   }
3613                 }
3614             },
3615             "type": "object",
3616             "allOf": [
3617               {"$ref": "oic.core.json#/definitions/oic.core"},
3618               {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
3619               {"$ref": "#/definitions/oic.r.operational.state"}
3620             ]
3621           }

```

```

3621     ]
3622   }
3623
3624   example: |
3625     {
3626       "id": "unique_example_id",
3627       "currentMachineState": "pause",
3628       "currentJobState": "wash"
3629     }
3630
3631   403:
3632     description: |
3633       This response is generated by the OIC Server when the client sends:
3634       An update with an value for currentMachineState that is not found in machineStates.
3635       An update with an value for currentJobState that is not found in jobStates.
3636       The server responds with the current resource representation.
3637
3638   body:
3639     application/json:
3640       schema: |
3641         {
3642           "id": "http://openinterconnect.org/schemas/oic.r.operational.state#",
3643           "$schema": "http://json-schema.org/draft-04/schema#",
3644           "title": "Operational State",
3645           "definitions": {
3646             "oic.r.operational.state": {
3647               "type": "object",
3648               "properties": {
3649                 "machineStates": {
3650                   "type": "string",
3651                   "description": "ReadOnly, Comma separated list of the possible operational
3652 states."
3653                 },
3654                 "currentMachineState": {
3655                   "type": "string",
3656                   "description": "Current state of operation of the device."
3657                 },
3658                 "jobStates": {
3659                   "type": "string",
3660                   "description": "ReadOnly, Comma separate list of the possible job states."
3661                 },
3662                 "currentJobState": {
3663                   "type": "string",
3664                   "description": "Currently active jobState"
3665                 },
3666                 "runningTime": {
3667                   "type": "string",
3668                   "description": "ReadOnly, Elapsed time in the current operational state"
3669                 },
3670                 "remainingTime": {
3671                   "type": "string",
3672                   "description": "ReadOnly, Time till completion of the current operational
3673 state"
3674                 },
3675                 "progressPercentage": {
3676                   "type": "integer",
3677                   "description": "ReadOnly, Percentage completeness of the current jobState"
3678                 }
3679               }
3680             }
3681           },
3682           "type": "object",
3683           "allOf": [
3684             {"$ref": "oic.core.json#/definitions/oic.core"},
3685             {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
3686             {"$ref": "#/definitions/oic.r.operational.state"}
3687           ],

```



```

3688         "required": ["machineStates", "currentMachineState"]
3689     }
3690
3691     example: |
3692     {
3693         "id":                "unique_example_id",
3694         "machineStates":    "pause, standby, ready, active",
3695         "currentMachineState": "active",
3696         "jobStates":        "pre-wash, wash, rinse, spin, dry, air-dry, wrinkle-
3697 prevent",
3698         "currentJobState":  "rinse",
3699         "runningTime":      "PT15M20S",
3700         "remainingTime":    "PT10M40S",
3701         "progressPercentage": 75
3702     }
3703

```

### 3704 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

### 3705 6.18.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/OperationalStateResURI		get	post		

### 3706 6.19 Ramp Time

#### 3707 6.19.1 Introduction

3708 This resource that describes the Ramp Time of a dimming function. This specifies the actual  
3709 speed of changing between 2 dimming values. Time is specified in milliseconds [ms]. If the range  
3710 value is not specified then the maximum value is 100 ms. The RampTime of 0ms indicates the  
3711 minimal delay possible by the implementation.

#### 3712 6.19.2 Example URI

3713 /RampTimeResURI

#### 3714 6.19.3 Resource Type

3715 The resource type (rt) is defined as: oic.r.light.rampTime.

#### 3716 6.19.4 RAML Definition

```

3717 #%RAML 0.8
3718 title: OICRampTime
3719 version: v1.0-20150727
3720 traits:
3721   - interface :
3722     queryParameters:

```

```

3723     if:
3724         enum: ["oic.if.a"]
3725
3726 /RampTimeResURI:
3727     description: |
3728         This resource that describes the Ramp Time of a dimming function.
3729         This specifies the actual speed of changing between 2 dimming values.
3730         Time is specified in milliseconds [ms].
3731         If the range value is not specified then the maximum value is 100 ms.
3732         The RampTime of 0ms indicates the minimal delay possible by the implementation.
3733
3734     is : ['interface']
3735     get:
3736         description: |
3737             Retrieves the current RampTime.
3738
3739     responses :
3740         200:
3741             body:
3742                 application/json:
3743                     schema: |
3744                         {
3745                             "id": "http://openinterconnect.org/schemas/oic.r.light.rampTime#",
3746                             "$schema": "http://json-schema.org/draft-04/schema#",
3747                             "title": "Ramp Time",
3748                             "definitions": {
3749                                 "oic.r.light.rampTime": {
3750                                     "type": "object",
3751                                     "properties": {
3752                                         "rampTime": {
3753                                             "type": "integer",
3754                                             "description": "Actual speed of changing between 2 dimming values"
3755                                         },
3756                                         "range": {
3757                                             "type": "string",
3758                                             "description": "ReadOnly, Min and Max of possible values"
3759                                         }
3760                                     }
3761                                 }
3762                             },
3763                             "type": "object",
3764                             "allOf": [
3765                                 {"$ref": "oic.core.json#/definitions/oic.core"},
3766                                 {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
3767                                 {"$ref": "#/definitions/oic.r.light.rampTime"}
3768                             ],
3769                             "required": ["rampTime"]
3770                         }
3771
3772     example: |
3773         {
3774             "rt":          "oic.r.light.rampTime",
3775             "id":          "unique_example_id",
3776             "rampTime": 0,
3777             "range":      "0,100"
3778         }
3779
3780     post:
3781         description: |
3782             Sets the current RampTime.
3783
3784     body:

```

```

3785 application/json:
3786   schema: |
3787     {
3788       "id": "http://openinterconnect.org/schemas/oic.r.light.rampTime#",
3789       "$schema": "http://json-schema.org/draft-04/schema#",
3790       "title": "Ramp Time",
3791       "definitions": {
3792         "oic.r.light.rampTime": {
3793           "type": "object",
3794           "properties": {
3795             "rampTime": {
3796               "type": "integer",
3797               "description": "Actual speed of changing between 2 dimming values"
3798             },
3799             "range": {
3800               "type": "string",
3801               "description": "ReadOnly, Min and Max of possible values"
3802             }
3803           }
3804         }
3805       },
3806       "type": "object",
3807       "allOf": [
3808         {"$ref": "oic.core.json#/definitions/oic.core"},
3809         {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
3810         {"$ref": "#/definitions/oic.r.light.rampTime"}
3811       ],
3812       "required": ["rampTime"]
3813     }
3814
3815   example: |
3816     {
3817       "id": "unique_example_id",
3818       "rampTime": 50
3819     }
3820
3821 responses :
3822   200:
3823     body:
3824       application/json:
3825         schema: |
3826           {
3827             "id": "http://openinterconnect.org/schemas/oic.r.light.rampTime#",
3828             "$schema": "http://json-schema.org/draft-04/schema#",
3829             "title": "Ramp Time",
3830             "definitions": {
3831               "oic.r.light.rampTime": {
3832                 "type": "object",
3833                 "properties": {
3834                   "rampTime": {
3835                     "type": "integer",
3836                     "description": "Actual speed of changing between 2 dimming values"
3837                   },
3838                   "range": {
3839                     "type": "string",
3840                     "description": "ReadOnly, Min and Max of possible values"
3841                   }
3842                 }
3843               }
3844             },
3845             "type": "object",
3846             "allOf": [
3847               {"$ref": "oic.core.json#/definitions/oic.core"},
3848               {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
3849               {"$ref": "#/definitions/oic.r.light.rampTime"}
3850             ],

```

```

3851         "required": ["rampTime"]
3852     }
3853
3854     example: |
3855         {
3856             "id":          "unique_example_id",
3857             "rampTime": 50
3858         }
3859
3860     403:
3861         description: |
3862             This response is generated by the OIC Server when the client sends:
3863             An update with an out of range property value for rampTime.
3864             The server responds with the current resource representation.
3865
3866     body:
3867         application/json:
3868             schema: |
3869                 {
3870                     "id": "http://openinterconnect.org/schemas/oic.r.light.rampTime#",
3871                     "$schema": "http://json-schema.org/draft-04/schema#",
3872                     "title": "Ramp Time",
3873                     "definitions": {
3874                         "oic.r.light.rampTime": {
3875                             "type": "object",
3876                             "properties": {
3877                                 "rampTime": {
3878                                     "type": "integer",
3879                                     "description": "Actual speed of changing between 2 dimming values"
3880                                 },
3881                                 "range": {
3882                                     "type": "string",
3883                                     "description": "ReadOnly, Min and Max of possible values"
3884                                 }
3885                             }
3886                         }
3887                     },
3888                     "type": "object",
3889                     "allOf": [
3890                         {"$ref": "oic.core.json#/definitions/oic.core"},
3891                         {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
3892                         {"$ref": "#/definitions/oic.r.light.rampTime"}
3893                     ],
3894                     "required": ["rampTime"]
3895                 }
3896
3897             example: |
3898                 {
3899                     "id":          "unique_example_id",
3900                     "rampTime": 40
3901                 }
3902
3903

```

### 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		

3905 **6.20 Refrigeration**

3906 **6.20.1 Introduction**

3907 This resource describes a refrigeration function. This is not a Refrigerator device. The filter state  
3908 is a read-only value providing the percentage life time remaining for the water filter. RapidFreeze  
3909 is a boolean that controls the rapid freeze capability if present. RapidCool is a boolean that  
3910 controls the rapid cool capability if present. Defrost is a boolean that controls the defrost cycle if  
3911 present.

3912 **6.20.2 Example URI**

3913 /RefrigerationResURI

3914 **6.20.3 Resource Type**

3915 The resource type (rt) is defined as: oic.r.refrigeration.

3916 **6.20.4 RAML Definition**

```
3917 #%RAML 0.8
3918 title: OICRefrigeration
3919 version: v1.0-20150727
3920 traits:
3921   - interface :
3922       queryParameters:
3923         if:
3924           enum: ["oic.if.a"]
3925
3926 /RefrigerationResURI:
3927   description: |
3928     This resource describes a refrigeration function.
3929     This is not a Refrigerator device.
3930     The filter state is a read-only value providing the percentage life time remaining for the
3931     water filter.
3932     RapidFreeze is a boolean that controls the rapid freeze capability if present.
3933     RapidCool is a boolean that controls the rapid cool capability if present.
3934     Defrost is a boolean that controls the defrost cycle if present.
3935
3936   is : ['interface']
3937   get:
3938     description: |
3939       Retrieves the current Refrigeration function status.
3940
3941   responses :
3942     200:
3943       body:
3944         application/json:
3945           schema: |
3946             {
3947               "id": "http://openinterconnect.org/schemas/oic.r.refrigeration#",
3948               "$schema": "http://json-schema.org/draft-04/schema#",
3949               "title": "Refrigeration",
3950               "definitions": {
3951                 "oic.r.refrigeration": {
3952                   "type": "object",
3953                   "properties": {
3954                     "filter": {
3955                       "type": "integer",
3956                       "description": "ReadOnly, Percentage life time remaining for the water
3957 filter"
3958                     },
3959                   "supportedFunctions": {
```

```

3960         "type": "string",
3961         "description": "ReadOnly, comma separated value set of supported functions"
3962     },
3963     "activeFunction": {
3964         "type": "string",
3965         "description": "Comma separated value set of active functions from the set
of supportedFunctions"
3966     }
3967 }
3968 }
3969 }
3970 },
3971 "type": "object",
3972 "allOf": [
3973     {"$ref": "oic.core.json#/definitions/oic.core"},
3974     {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
3975     {"$ref": "#/definitions/oic.r.refrigeration"}
3976 ],
3977 "required": ["supportedFunctions", "activeFunction"]
3978 }
3979
3980 example: |
3981 {
3982     "rt":         "oic.r.refrigeration",
3983     "id":         "unique_example_id",
3984     "filter":    75,
3985     "rapidFreeze": false,
3986     "rapidCool": false,
3987     "defrost":   true
3988 }
3989
3990 post:
3991 description: |
3992     Activates the desired Refrigeration functions.
3993     Supported values are rapidFreeze, rapidCool and defrost.
3994
3995 body:
3996 application/json:
3997 schema: |
3998 {
3999     "id": "http://openinterconnect.org/schemas/oic.r.refrigeration#",
4000     "$schema": "http://json-schema.org/draft-04/schema#",
4001     "title": "Refrigeration",
4002     "definitions": {
4003         "oic.r.refrigeration": {
4004             "type": "object",
4005             "properties": {
4006                 "rapidFreeze": {
4007                     "type": "boolean",
4008                     "description": "Indicates whether the unit has a rapid freeze capability
active."
4009                 },
4010                 "rapidCool": {
4011                     "type": "boolean",
4012                     "description": "Indicates whether the unit has a rapid cool capability active"
4013                 },
4014                 "defrost": {
4015                     "type": "boolean",
4016                     "description": "Indicates whether a defrost cycle is currently active"
4017                 }
4018             }
4019         }
4020     },
4021     "type": "object",
4022     "allOf": [
4023         {"$ref": "oic.core.json#/definitions/oic.core"},
4024         {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
4025         {"$ref": "#/definitions/oic.r.refrigeration"}
4026     ]

```

```

4027         ],
4028         "required": ["defrost"]
4029     }
4030
4031     example: |
4032         {
4033             "id":         "unique_example_id",
4034             "defrost":   false
4035         }
4036
4037     responses :
4038         200:
4039             description: |
4040                 Indicates that the Refrigeration function was changed.
4041                 The new status is provided in the response.
4042
4043             body:
4044                 application/json:
4045                     schema: |
4046                         {
4047                             "id": "http://openinterconnect.org/schemas/oic.r.refrigeration#",
4048                             "$schema": "http://json-schema.org/draft-04/schema#",
4049                             "title": "Refrigeration",
4050                             "definitions": {
4051                                 "oic.r.refrigeration": {
4052                                     "type": "object",
4053                                     "properties": {
4054                                         "rapidFreeze": {
4055                                             "type": "boolean",
4056                                             "description": "Indicates whether the unit has a rapid freeze capability
4057 active."
4058                                         },
4059                                         "rapidCool": {
4060                                             "type": "boolean",
4061                                             "description": "Indicates whether the unit has a rapid cool capability
4062 active"
4063                                         },
4064                                         "defrost": {
4065                                             "type": "boolean",
4066                                             "description": "Indicates whether a defrost cycle is currently active"
4067                                         }
4068                                     }
4069                                 }
4070                             },
4071                             "type": "object",
4072                             "allOf": [
4073                                 {"$ref": "oic.core.json#/definitions/oic.core"},
4074                                 {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
4075                                 {"$ref": "#/definitions/oic.r.refrigeration"}
4076                             ],
4077                             "required": ["defrost"]
4078                         }
4079
4080             example: |
4081                 {
4082                     "id":         "unique_example_id",
4083                     "defrost":   false
4084                 }
4085

```

#### 4086 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

4087 **6.20.6 CRUDN behavior**

Resource	Create	Read	Update	Delete	Notify
/RefrigerationResURI		get	post		

4088 **6.21 Temperature**

4089 **6.21.1 Introduction**

4090 This resource describes a sensed or actuated Temperature value. The temperature describes the  
4091 current value measured. If the units attribute is missing the default is Celsius [C]. The range is a  
4092 comma separated min,max values for this temperature on this device. If no range is provided the  
4093 default is +/- MAXINT.

4094 **6.21.2 Example URI**

4095 /TemperatureResURI

4096 **6.21.3 Resource Type**

4097 The resource type (rt) is defined as: oic.r.temperature.

4098 **6.21.4 RAML Definition**

```

4099 #%RAML 0.8
4100 title: OICTemperature
4101 version: v1.0-20150727
4102 traits:
4103   - interface :
4104     queryParameters:
4105       if:
4106         enum: ["oic.if.a", "oic.if.s"]
4107
4108 /TemperatureResURI:
4109   description: |
4110     This resource describes a sensed or actuated Temperature value.
4111     The temperature describes the current value measured.
4112     If the units attribute is missing the default is Celsius [C].
4113     The range is a comma separated min,max values for this temperature on this device.
4114     If no range is provided the default is +/- MAXINT.
4115
4116   is : ['interface']
4117   get:
4118     description: |
4119       Retrieves the current temperature value.
4120       A client can specify the units for the requested temperature by use of a query parameter.
4121
4122     queryParameters:
4123       units:
4124         enum: CFK
4125     responses :
4126       200:
4127         body:
4128           application/json:
4129             schema: |

```



```

4130     {
4131         "id": "http://openinterconnect.org/schemas/oic.r.temperature#",
4132         "$schema": "http://json-schema.org/draft-04/schema#",
4133         "title": "Temperature",
4134         "definitions": {
4135             "oic.r.temperature": {
4136                 "type": "object",
4137                 "properties": {
4138                     "temperature": {
4139                         "type": "number",
4140                         "description": "Current temperature setting or measurement"
4141                     },
4142                     "units": {
4143                         "enum": ["C","F","K"],
4144                         "description": "ReadOnly, Units for the temperature value"
4145                     },
4146                     "range": {
4147                         "type": "string",
4148                         "description": "ReadOnly, Comma separated min,max values for this
4149 temperature on this device"
4150                     }
4151                 }
4152             }
4153         },
4154         "type": "object",
4155         "allOf": [
4156             {"$ref": "oic.core.json#/definitions/oic.core"},
4157             {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
4158             {"$ref": "#/definitions/oic.r.temperature"}
4159         ],
4160         "required": ["temperature"]
4161     }
4162 
```

```

4163     example: |
4164     {
4165         "rt":           "oic.r.temperature",
4166         "id":           "unique_example_id",
4167         "temperature": 20.0,
4168         "units":        "C",
4169         "range":        "0,100"
4170     }
4171 
```

```

4172     403:
4173     description: |
4174         This response is generated by the OIC Server when the client sends:
4175         A retrieve with q queryParameter indicating a unit that the server does not support.
4176         The server responds with the units enumeration illustrating the error.
4177 
```

```

4178     body:
4179     application/json:
4180     schema: |
4181     {
4182         "id": "http://openinterconnect.org/schemas/oic.r.temperature#",
4183         "$schema": "http://json-schema.org/draft-04/schema#",
4184         "title": "Temperature",
4185         "definitions": {
4186             "oic.r.temperature": {
4187                 "type": "object",
4188                 "properties": {
4189                     "units": {
4190                         "enum": ["C","F","K"],
4191                         "description": "ReadOnly, Units for the temperature value"
4192                     },
4193                     "range": {
4194                         "type": "string",
4195                         "description": "ReadOnly, Comma separated min,max values for this
4196 temperature on this device"

```

```

4197         }
4198     }
4199 }
4200 },
4201 "type": "object",
4202 "allOf": [
4203     {"$ref": "oic.core.json#/definitions/oic.core"},
4204     {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
4205     {"$ref": "#/definitions/oic.r.temperature"}
4206 ],
4207 }
4208
4209 example: |
4210 {
4211     "id": "unique_example_id",
4212     "units": "C"
4213 }
4214
4215 post:
4216 description: |
4217     Sets the desired temperature value.
4218     If the units are omitted the current value for units known by the server is used.
4219
4220 body:
4221 application/json:
4222 schema: |
4223     {
4224         "id": "http://openinterconnect.org/schemas/oic.r.temperature#",
4225         "$schema": "http://json-schema.org/draft-04/schema#",
4226         "title": "Temperature",
4227         "definitions": {
4228             "oic.r.temperature": {
4229                 "type": "object",
4230                 "properties": {
4231                     "temperature": {
4232                         "type": "number",
4233                         "description": "Current temperature setting or measurement"
4234                     },
4235                     "units": {
4236                         "enum": ["C", "F", "K"],
4237                         "description": "ReadOnly, Units for the temperature value"
4238                     },
4239                     "range": {
4240                         "type": "string",
4241                         "description": "ReadOnly, Comma separated min,max values for this temperature
4242 on this device"
4243                     }
4244                 }
4245             }
4246         },
4247         "type": "object",
4248         "allOf": [
4249             {"$ref": "oic.core.json#/definitions/oic.core"},
4250             {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
4251             {"$ref": "#/definitions/oic.r.temperature"}
4252         ],
4253         "required": ["temperature"]
4254     }
4255
4256 example: |
4257 {
4258     "id": "unique_example_id",
4259     "temperature": 18
4260 }
4261

```

```

4262     responses :
4263         200:
4264             body:
4265                 application/json:
4266                     schema: |
4267                         {
4268                             "id": "http://openinterconnect.org/schemas/oic.r.temperature#",
4269                             "$schema": "http://json-schema.org/draft-04/schema#",
4270                             "title": "Temperature",
4271                             "definitions": {
4272                                 "oic.r.temperature": {
4273                                     "type": "object",
4274                                     "properties": {
4275                                         "temperature": {
4276                                             "type": "number",
4277                                             "description": "Current temperature setting or measurement"
4278                                         },
4279                                         "units": {
4280                                             "enum": ["C","F","K"],
4281                                             "description": "ReadOnly, Units for the temperature value"
4282                                         },
4283                                         "range": {
4284                                             "type": "string",
4285                                             "description": "ReadOnly, Comma separated min,max values for this
4286 temperature on this device"
4287                                         }
4288                                     }
4289                                 },
4290                             "type": "object",
4291                             "allOf": [
4292                                 {"$ref": "oic.core.json#/definitions/oic.core"},
4293                                 {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
4294                                 {"$ref": "#/definitions/oic.r.temperature"}
4295                             ],
4296                             "required": ["temperature"]
4297                         }
4298                     }
4299
4300                 example: |
4301                     {
4302                         "id": "unique_example_id",
4303                         "temperature": 18
4304                     }
4305
4306         403:
4307             description: |
4308                 This response is generated by the OIC Server when the client sends:
4309                 An update with an out of range property value for temperature.
4310                 The server responds with the range property illustrating the error.
4311
4312             body:
4313                 application/json:
4314                     schema: |
4315                         {
4316                             "id": "http://openinterconnect.org/schemas/oic.r.temperature#",
4317                             "$schema": "http://json-schema.org/draft-04/schema#",
4318                             "title": "Temperature",
4319                             "definitions": {
4320                                 "oic.r.temperature": {
4321                                     "type": "object",
4322                                     "properties": {
4323                                         "units": {
4324                                             "enum": ["C","F","K"],
4325                                             "description": "ReadOnly, Units for the temperature value"
4326                                         },

```

```

4327         "range": {
4328             "type": "string",
4329             "description": "ReadOnly, Comma separated min,max values for this
4330 temperature on this device"
4331         }
4332     }
4333 },
4334 },
4335 "type": "object",
4336 "allOf": [
4337     {"$ref": "oic.core.json#/definitions/oic.core"},
4338     {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
4339     {"$ref": "#/definitions/oic.r.temperature"}
4340 ]
4341 }
4342
4343 example: |
4344 {
4345     "id":      "unique_example_id",
4346     "range":  "0,100"
4347 }
4348

```

### 4349 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

### 4350 6.21.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/TemperatureResURI		get	post		

## 4351 6.22 Time Period

### 4352 6.22.1 Introduction

4353 This resource describes the time period over which any additionally provided information is  
4354 derived or bounded. The start and stop times are ISO8601 encoded strings.

### 4355 6.22.2 Example URI

4356 /TimePeriodResURI

### 4357 6.22.3 Resource Type

4358 The resource type (rt) is defined as: oic.r.time.period.

### 4359 6.22.4 RAML Definition

```

4360 #%RAML 0.8
4361 title: OICTimePeriod
4362 version: v1.0-20150727
4363 traits:
4364   - interface :
4365       queryParameters:
4366           if:
4367               enum: ["oic.if.a"]
4368
4369 /TimePeriodResURI:
4370     description: |

```

```

4371     This resource describes the time period over which any additionally provided
4372     information is derived or bounded.
4373     The start and stop times are ISO8601 encoded strings.
4374
4375     is : ['interface']
4376
4377     get:
4378         description: |
4379             Defines a time period for information retrieval, action or other behaviour.
4380
4381     responses :
4382         200:
4383             body:
4384                 application/json:
4385                     schema: |
4386                         {
4387                             "id": "http://openinterconnect.org/schemas/oic.r.time.period#",
4388                             "$schema": "http://json-schema.org/draft-04/schema#",
4389                             "title": "Time Period",
4390                             "definitions": {
4391                                 "oic.r.time.period": {
4392                                     "type": "object",
4393                                     "properties": {
4394                                         "startTime": {
4395                                             "type": "string",
4396                                             "description": "Start time for the time period"
4397                                         },
4398                                         "stopTime": {
4399                                             "type": "string",
4400                                             "description": "Stop time for the time period"
4401                                         }
4402                                     }
4403                                 },
4404                                 "type": "object",
4405                                 "allOf": [
4406                                     {"$ref": "oic.core.json#/definitions/oic.core"},
4407                                     {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
4408                                     {"$ref": "#/definitions/oic.r.time.period"}
4409                                 ],
4410                                 "required": ["startTime"]
4411                             }
4412
4413                     example: |
4414                         {
4415                             "rt": "oic.r.time.period",
4416                             "id": "unique_example_id",
4417                             "startTime": "2015-01-09T14:30Z",
4418                             "stopTime": "2015-01-09T14:45Z"
4419                         }
4420
4421     post:
4422         description: |
4423             Sets or updates a time period for information retrieval, action or other behavior.
4424
4425         body:
4426             application/json:
4427                 schema: |
4428                     {
4429                         "id": "http://openinterconnect.org/schemas/oic.r.time.period#",
4430                         "$schema": "http://json-schema.org/draft-04/schema#",
4431                         "title": "Time Period",
4432                         "definitions": {
4433                             "oic.r.time.period": {

```

```

4434         "type": "object",
4435         "properties": {
4436             "startTime": {
4437                 "type": "string",
4438                 "description": "Start time for the time period"
4439             },
4440             "stopTime": {
4441                 "type": "string",
4442                 "description": "Stop time for the time period"
4443             }
4444         }
4445     },
4446     "type": "object",
4447     "allOf": [
4448         {"$ref": "oic.core.json#/definitions/oic.core"},
4449         {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
4450         {"$ref": "#/definitions/oic.r.time.period"}
4451     ],
4452     "required": ["startTime"]
4453 }
4454
4455
4456 example: |
4457     {
4458         "id": "unique_example_id",
4459         "startTime": "2015-01-09T14:30Z",
4460         "stopTime": "2015-01-09T14:45Z"
4461     }
4462
4463 responses :
4464     200:
4465         body:
4466             application/json:
4467                 schema: |
4468                     {
4469                         "id": "http://openinterconnect.org/schemas/oic.r.time.period#",
4470                         "$schema": "http://json-schema.org/draft-04/schema#",
4471                         "title": "Time Period",
4472                         "definitions": {
4473                             "oic.r.time.period": {
4474                                 "type": "object",
4475                                 "properties": {
4476                                     "startTime": {
4477                                         "type": "string",
4478                                         "description": "Start time for the time period"
4479                                     },
4480                                     "stopTime": {
4481                                         "type": "string",
4482                                         "description": "Stop time for the time period"
4483                                     }
4484                                 }
4485                             }
4486                         },
4487                         "type": "object",
4488                         "allOf": [
4489                             {"$ref": "oic.core.json#/definitions/oic.core"},
4490                             {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
4491                             {"$ref": "#/definitions/oic.r.time.period"}
4492                         ],
4493                         "required": ["startTime"]
4494                     }
4495
4496 example: |
4497     {
4498         "id": "unique_example_id",
4499         "startTime": "2015-01-09T14:30Z",

```

```

4500         "stopTime": "2015-01-09T14:45Z"
4501     }
4502

```

### 4503 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

### 4504 6.22.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/TimePeriodResURI		get	post		

## 4505 6.23 Activity Count

### 4506 6.23.1 Introduction

4507 This resource specifies an activity count. The resource can be readonly (oic.if.s interface) in  
 4508 which instance it represents a count. The resource can be readwrite (oic.if.a interface) in which  
 4509 instance it represents a goal or target for a count. The count property is an integer representing  
 4510 either the current count or goal value.

### 4511 6.23.2 Example URI

4512 /ActivityCountResURI

### 4513 6.23.3 Resource Type

4514 The resource type (rt) is defined as: oic.r.sensor.activity.count.

### 4515 6.23.4 RAML Definition

```

4516 #%RAML 0.8
4517 title: OICActivityCount
4518 version: v1.0-20150727
4519 traits:
4520   - interface :
4521     queryParameters:
4522       if:
4523         enum: ["oic.if.s", "oic.if.a"]
4524
4525 /ActivityCountResURI:
4526   description: |
4527     This resource specifies an activity count.
4528     The resource can be readonly (oic.if.s interface) in which instance it represents a count.
4529     The resource can be readwrite (oic.if.a interface) in which instance it represents a goal or
4530     target for a count.
4531     The count property is an integer representing either the current count or goal value.
4532
4533   is : ['interface']
4534   get:
4535     description: |
4536       Retrieves the current activity count.
4537
4538   responses :
4539     200:
4540       body:
4541         application/json:
4542           schema: |
4543             {
4544               "id": "http://openinterconnect.org/schemas/oic.r.sensor.activity.count.json#",
4545               "$schema": "http://json-schema.org/draft-04/schema#",

```

```

4546         "title": "Activity Count Sensor",
4547         "definitions": {
4548             "oic.r.sensor.activity.count": {
4549                 "properties": {
4550                     "count": {
4551                         "type": "integer",
4552                         "description": "Current or Target count."
4553                     }
4554                 }
4555             }
4556         },
4557         "type": "object",
4558         "allOf": [
4559             {"$ref": "oic.core.json#/definitions/oic.core"},
4560             {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
4561             {"$ref": "#/definitions/oic.r.sensor.activity.count"}
4562         ],
4563         "required": ["count"]
4564     }
4565
4566     example: |
4567         {
4568             "rt":      "oic.r.sensor.activity.count",
4569             "id":      "unique_example_id",
4570             "count":  2500
4571         }
4572
4573     post:
4574         description: |
4575             Sets the count target
4576
4577     body:
4578         application/json:
4579             schema: |
4580                 {
4581                     "id": "http://openinterconnect.org/schemas/oic.r.sensor.activity.count.json#",
4582                     "$schema": "http://json-schema.org/draft-04/schema#",
4583                     "title": "Activity Count Sensor",
4584                     "definitions": {
4585                         "oic.r.sensor.activity.count": {
4586                             "properties": {
4587                                 "count": {
4588                                     "type": "integer",
4589                                     "description": "Current or Target count."
4590                                 }
4591                             }
4592                         }
4593                     },
4594                     "type": "object",
4595                     "allOf": [
4596                         {"$ref": "oic.core.json#/definitions/oic.core"},
4597                         {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
4598                         {"$ref": "#/definitions/oic.r.sensor.activity.count"}
4599                     ],
4600                     "required": ["count"]
4601                 }
4602
4603             example: |
4604                 {
4605                     "id":      "unique_example_id",
4606                     "count":  5000
4607                 }
4608
4609     responses :
4610         200:

```



```

4611     body:
4612         application/json:
4613             schema: |
4614                 {
4615                     "id": "http://openinterconnect.org/schemas/oic.r.sensor.activity.count.json#",
4616                     "$schema": "http://json-schema.org/draft-04/schema#",
4617                     "title": "Activity Count Sensor",
4618                     "definitions": {
4619                         "oic.r.sensor.activity.count": {
4620                             "properties": {
4621                                 "count": {
4622                                     "type": "integer",
4623                                     "description": "Current or Target count."
4624                                 }
4625                             }
4626                         },
4627                     },
4628                     "type": "object",
4629                     "allOf": [
4630                         {"$ref": "oic.core.json#/definitions/oic.core"},
4631                         {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
4632                         {"$ref": "#/definitions/oic.r.sensor.activity.count"}
4633                     ],
4634                     "required": ["count"]
4635                 }
4636
4637             example: |
4638                 {
4639                     "id": "unique_example_id",
4640                     "count": 5000
4641                 }
4642

```

4643 **6.23.5 Property Definition**

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

4644 **6.23.6 CRUDN behavior**

Resource	Create	Read	Update	Delete	Notify
/ActivityCountResURI		get	post		

4645 **6.24 Atmospheric Pressure Sensor**

4646 **6.24.1 Introduction**

4647 This resource provides a measurement of Mean Sea Level Pressure experienced at the  
4648 measuring point expressed in millibars. The value is float which describes the atmospheric  
4649 pressure in millibars.

4650 **6.24.2 Example URI**

4651 /AtmosphericPressureResURI

4652 **6.24.3 Resource Type**

4653 The resource type (rt) is defined as: oic.r.sensor.atmosphericPressure.

4654 **6.24.4 RAML Definition**

```

4655 #%RAML 0.8
4656 title: OICAtmosphericPressureSensor
4657 version: v1.0-20150727
4658 traits:
4659     - interface :
4660         queryParameters:
4661             if:

```

```

4662         enum: ["oic.if.s"]
4663
4664 /AtmosphericPressureResURI:
4665     description: |
4666         This resource provides a measurement of Mean Sea Level Pressure experienced at the measuring
4667         point expressed in millibars.
4668         The value is float which describes the atmospheric pressure in millibars.
4669
4670     is : ['interface']
4671     get:
4672         responses :
4673             200:
4674                 body:
4675                     application/json:
4676                         schema: |
4677                             {
4678                                 "id": "http://openinterconnect.org/schemas/oic.r.sensor.atmosphericPressure.json#",
4679                                 "$schema": "http://json-schema.org/draft-04/schema#",
4680                                 "title": "Atmospheric Pressure Sensor",
4681                                 "definitions": {
4682                                     "oic.r.sensor.atmosphericPressure": {
4683                                         "properties": {
4684                                             "atmosphericPressure": {
4685                                                 "type": "number",
4686                                                 "description": "ReadOnly, Current atmospheric pressure in mbar."
4687                                             }
4688                                         }
4689                                     }
4690                                 },
4691                                 "type": "object",
4692                                 "allOf": [
4693                                     {"$ref": "oic.core.json#/definitions/oic.core"},
4694                                     {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
4695                                     {"$ref": "#/definitions/oic.r.sensor.atmosphericPressure"}
4696                                 ],
4697                                 "required": ["atmosphericPressure"]
4698                             }
4699
4700                         example: |
4701                             {
4702                                 "rt": "oic.r.sensor.atmosphericPressure",
4703                                 "id": "unique_example_id",
4704                                 "atmosphericPressure": 1000
4705                             }
4706

```

#### 4707 6.24.5 Property Definition

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

#### 4708 6.24.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/AtmosphericPressureResURI		get			

### 4709 6.25 Audio Controls

#### 4710 6.25.1 Introduction

4711 This resource defines basic audio control functions. The volume is an integer containing a  
4712 percentage [0,100]. A volume of 0 (zero) means no sound produced. A volume of 100 means

4713 maximum sound production. The mute control is implemented as a boolean. A mute value of true  
4714 means that the device is muted (no audio). A mute value of false means that the device is not  
4715 muted (audio).

## 4716 6.25.2 Example URI

4717 /AudioResURI

## 4718 6.25.3 Resource Type

4719 The resource type (rt) is defined as: oic.r.audio.

## 4720 6.25.4 RAML Definition

```
4721 #%RAML 0.8
4722 title: OICAudio
4723 version: v1.0-20150727
4724 traits:
4725   - interface :
4726     queryParameters:
4727       if:
4728         enum: ["oic.if.a"]
4729
4730 /AudioResURI:
4731   description: |
4732     This resource defines basic audio control functions.
4733     The volume is an integer containing a percentage [0,100].
4734     A volume of 0 (zero) means no sound produced.
4735     A volume of 100 means maximum sound production.
4736     The mute control is implemented as a boolean.
4737     A mute value of true means that the device is muted (no audio).
4738     A mute value of false means that the device is not muted (audio).
4739
4740   is : ['interface']
4741   get:
4742     responses :
4743       200:
4744         body:
4745           application/json:
4746             schema: |
4747               {
4748                 "id": "http://openinterconnect.org/schemas/oic.r.audio#",
4749                 "$schema": "http://json-schema.org/draft-04/schema#",
4750                 "definitions": {
4751                   "oic.r.audio": {
4752                     "type": "object",
4753                     "properties": {
4754                       "volume": {
4755                         "type": "integer",
4756                         "description": "Volume setting of an audio rendering device."
4757                       },
4758                       "mute": {
4759                         "type": "boolean",
4760                         "description": "Mute setting of an audio rendering device"
4761                       }
4762                     }
4763                   }
4764                 },
4765                 "type": "object",
4766                 "allOf": [
4767                   {"$ref": "oic.core.json#/definitions/oic.core"},
4768                   {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
4769                   {"$ref": "#/definitions/oic.r.audio"}
4770                 ],
4771                 "required": ["volume","mute"]
```

```

4772     }
4773
4774     example: |
4775     {
4776         "rt":      "oic.r.audio",
4777         "id":      "unique_example_id",
4778         "volume": 50,
4779         "mute":   false
4780     }
4781
4782     post:
4783     body:
4784     application/json:
4785     schema: |
4786     {
4787         "id": "http://openinterconnect.org/schemas/oic.r.audio#",
4788         "$schema": "http://json-schema.org/draft-04/schema#",
4789         "definitions": {
4790             "oic.r.audio": {
4791                 "type": "object",
4792                 "properties": {
4793                     "volume": {
4794                         "type": "integer",
4795                         "description": "Volume setting of an audio rendering device."
4796                     },
4797                     "mute": {
4798                         "type": "boolean",
4799                         "description": "Mute setting of an audio rendering device"
4800                     }
4801                 }
4802             }
4803         },
4804         "type": "object",
4805         "allOf": [
4806             {"$ref": "oic.core.json#/definitions/oic.core"},
4807             {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
4808             {"$ref": "#/definitions/oic.r.audio"}
4809         ],
4810         "required": ["volume", "mute"]
4811     }
4812
4813     example: |
4814     {
4815         "id":      "unique_example_id",
4816         "volume": 75,
4817         "mute":   false
4818     }
4819
4820     responses :
4821     200:
4822     body:
4823     application/json:
4824     schema: |
4825     {
4826         "id": "http://openinterconnect.org/schemas/oic.r.audio#",
4827         "$schema": "http://json-schema.org/draft-04/schema#",
4828         "definitions": {
4829             "oic.r.audio": {
4830                 "type": "object",
4831                 "properties": {
4832                     "volume": {
4833                         "type": "integer",
4834                         "description": "Volume setting of an audio rendering device."
4835                     },

```

```

4836         "mute": {
4837             "type": "boolean",
4838             "description": "Mute setting of an audio rendering device"
4839         }
4840     }
4841 }
4842 },
4843 "type": "object",
4844 "allof": [
4845     {"$ref": "oic.core.json#/definitions/oic.core"},
4846     {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
4847     {"$ref": "#/definitions/oic.r.audio"}
4848 ],
4849 "required": ["volume", "mute"]
4850 }
4851
4852 example: |
4853 {
4854     "id": "unique_example_id",
4855     "volume": 75,
4856     "mute": false
4857 }
4858

```

### 4859 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

### 4860 6.25.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/AudioResURI		get	post		

## 4861 6.26 Auto Focus

### 4862 6.26.1 Introduction

4863 This resource describes an auto focus on/off feature (on/off). The value is a boolean. An  
4864 AutoFocus value of 'true' means that the switch is on. An AutoFocus value of 'false' means that  
4865 the switch is off. Note that when PTZ is used the autofocus works only the selected area.

### 4866 6.26.2 Example URI

4867 /AutoFocusResURI

### 4868 6.26.3 Resource Type

4869 The resource type (rt) is defined as: oic.r.autofocus.

### 4870 6.26.4 RAML Definition

```

4871 #%RAML 0.8
4872 title: OICAutoFocus
4873 version: v1.0-20150727
4874 traits:
4875   - interface :
4876       queryParameters:
4877         if:
4878           enum: ["oic.if.a"]
4879
4880 /AutoFocusResURI:

```

```

4881 description: |
4882     This resource describes an auto focus on/off feature (on/off).
4883     The value is a boolean.
4884     An AutoFocus value of 'true' means that the switch is on.
4885     An AutoFocus value of 'false' means that the switch is off.
4886     Note that when PTZ is used the autofocus works only the selected area.
4887
4888 is : ['interface']
4889 get:
4890     responses :
4891         200:
4892             body:
4893                 application/json:
4894                     schema: |
4895                         {
4896                             "id": "http://openinterconnect.org/schemas/oic.r.autofocus#",
4897                             "$schema": "http://json-schema.org/draft-04/schema#",
4898                             "title": "Auto Focus",
4899                             "definitions": {
4900                                 "oic.r.autofocus": {
4901                                     "type": "object",
4902                                     "properties": {
4903                                         "autoFocus": {
4904                                             "type": "boolean",
4905                                             "description": "Status of the Auto Focus"
4906                                         }
4907                                     }
4908                                 },
4909                             },
4910                             "type": "object",
4911                             "allOf": [
4912                                 { "$ref": "oic.core.json#/definitions/oic.core" },
4913                                 { "$ref": "oic.baseResource.json#/definitions/oic.r.baseResource" },
4914                                 { "$ref": "#/definitions/oic.r.autofocus" }
4915                             ],
4916                             "required": [ "autoFocus" ]
4917                         }
4918
4919                     example: |
4920                         {
4921                             "rt": "oic.r.autofocus",
4922                             "id": "unique_example_id",
4923                             "autoFocus": false
4924                         }
4925
4926 post:
4927     body:
4928         application/json:
4929             schema: |
4930                 {
4931                     "id": "http://openinterconnect.org/schemas/oic.r.autofocus#",
4932                     "$schema": "http://json-schema.org/draft-04/schema#",
4933                     "title": "Auto Focus",
4934                     "definitions": {
4935                         "oic.r.autofocus": {
4936                             "type": "object",
4937                             "properties": {
4938                                 "autoFocus": {
4939                                     "type": "boolean",
4940                                     "description": "Status of the Auto Focus"
4941                                 }
4942                             }
4943                         }
4944                     },
4945                     "type": "object",

```

```

4946     "allOf": [
4947         {"$ref": "oic.core.json#/definitions/oic.core"},
4948         {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
4949         {"$ref": "#/definitions/oic.r.autofocus"}
4950     ],
4951     "required": [ "autoFocus" ]
4952 }
4953
4954 example: |
4955 {
4956     "id":          "unique_example_id",
4957     "autoFocus":  true
4958 }
4959
4960 responses :
4961 200:
4962   body:
4963     application/json:
4964       schema: |
4965         {
4966           "id": "http://openinterconnect.org/schemas/oic.r.autofocus#",
4967           "$schema": "http://json-schema.org/draft-04/schema#",
4968           "title": "Auto Focus",
4969           "definitions": {
4970             "oic.r.autofocus": {
4971               "type": "object",
4972               "properties": {
4973                 "autoFocus": {
4974                   "type": "boolean",
4975                   "description": "Status of the Auto Focus"
4976                 }
4977             }
4978           },
4979           "type": "object",
4980           "allOf": [
4981             {"$ref": "oic.core.json#/definitions/oic.core"},
4982             {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
4983             {"$ref": "#/definitions/oic.r.autofocus"}
4984           ],
4985           "required": [ "autoFocus" ]
4986         }
4987
4988       example: |
4989         {
4990           "id":          "unique_example_id",
4991           "autoFocus":  true
4992         }
4993
4994

```

### 4995 6.26.5 Property Definition

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

### 4996 6.26.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/AutoFocusResURI		get	post		

## 4997 6.27 Automatic Document Feeder

### 4998 6.27.1 Introduction

4999 This resource describes the state of an automatic document feeder, typically used with a scanner.  
5000 The states are read only. The adfStates is a comma separated list of the possible operational

5001 states. adfProcessing – the OK state, other states are errors or require ‘user attention’. The  
5002 currentAdfState is the current value of the ADF state on the device.

### 5003 6.27.2 Example URI

5004 /AutomaticDocumentFeederResURI

### 5005 6.27.3 Resource Type

5006 The resource type (rt) is defined as: oic.r.automaticDocumentFeeder.

### 5007 6.27.4 RAML Definition

```
5008 #%RAML 0.8
5009 title: OICAutomaticDocumentFeeder
5010 version: v1.0-20150727
5011 traits:
5012   - interface :
5013     queryParameters:
5014       if:
5015         enum: ["oic.if.s"]
5016
5017 /AutomaticDocumentFeederResURI:
5018   description: |
5019     This resource describes the state of an automatic document feeder, typically used with a
5020 scanner.
5021     The states are read only.
5022     The adfStates is a comma separated list of the possible operational states.
5023     adfProcessing - the OK state, other states are errors or require ‘user attention’.
5024     The currentAdfState is the current value of the ADF state on the device.
5025
5026   is : ['interface']
5027   get:
5028     description: |
5029       Retrieves the current automatic document feeder state.
5030
5031   responses :
5032     200:
5033       body:
5034         application/json:
5035           schema: |
5036             {
5037               "id": "http://openinterconnect.org/schemas/oic.r.automaticDocumentFeeder#",
5038               "$schema": "http://json-schema.org/draft-04/schema#",
5039               "title": "Automatic Document Feeder",
5040               "definitions": {
5041                 "oic.r.automaticDocumentFeeder": {
5042                   "type": "object",
5043                   "properties": {
5044                     "adfStates": {
5045                       "type": "string",
5046                       "description": "ReadOnly, Comma separated list of the possible adf states."
5047                     },
5048                     "currentAdfState": {
5049                       "type": "string",
5050                       "description": "ReadOnly, Current adf state."
5051                     }
5052                   }
5053                 }
5054             },
5055           "type": "object",
5056           "allOf": [
5057             {"$ref": "oic.core.json#/definitions/oic.core"},
5058             {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
```



```

5059         {"$ref": "#/definitions/oic.r.automaticDocumentFeeder"}
5060     },
5061     "required": ["adfStates", "currentAdfState"]
5062 }
5063
5064     example: |
5065     {
5066         "rt":             "oic.r.automaticDocumentFeeder",
5067         "id":             "unique_example_id",
5068         "adfStates":     "adfProcessing, adfEmpty, adfJam, adfLoaded, adfMispick,
5069 adfHatchOpen, adfDuplexPageTooShort, adfDuplexPageTooLong, adfMultipickDetected, adfInputTrayFailed,
5070 adfInputTrayOverloaded",
5071         "currentAdfState": "adfProcessing"
5072     }
5073

```

## 5074 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.

## 5075 6.27.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/AutomaticDocumentFeederResURI		get			

## 5076 6.28 Button Switch

### 5077 6.28.1 Introduction

5078 This resource describes the operation of a button style switch. The value is a boolean. A value of  
5079 True means that the button is being pushed/pressed. A value of False means that the button is  
5080 not being pushed/pressed.

### 5081 6.28.2 Example URI

5082 /ButtonResURI

### 5083 6.28.3 Resource Type

5084 The resource type (rt) is defined as: oic.r.button.

### 5085 6.28.4 RAML Definition

```

5086 #%RAML 0.8
5087 title: OICButton
5088 version: v1.0-20150727
5089 traits:
5090   - interface :
5091       queryParameters:
5092           if:
5093               enum: ["oic.if.s"]
5094
5095 /ButtonResURI:
5096   description: |
5097       This resource describes the operation of a button style switch.
5098       The value is a boolean.
5099       A value of True means that the button is being pushed/pressed.
5100       A value of False means that the button is not being pushed/pressed.
5101
5102   is : ['interface']
5103   get:
5104       responses :

```

```

5105     200:
5106         body:
5107             application/json:
5108                 schema: |
5109                     {
5110                         "id": "http://openinterconnect.org/schemas/oic.r.button.json#",
5111                         "$schema": "http://json-schema.org/draft-04/schema#",
5112                         "title": "Button Switch",
5113                         "definitions": {
5114                             "oic.r.button": {
5115                                 "properties": {
5116                                     "value": {
5117                                         "type": "boolean",
5118                                         "description": "ReadOnly, Status of the button"
5119                                     }
5120                                 }
5121                             }
5122                         },
5123                         "type": "object",
5124                         "allOf": [
5125                             {"$ref": "oic.core.json#/definitions/oic.core"},
5126                             {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
5127                             {"$ref": "#/definitions/oic.r.button"}
5128                         ],
5129                         "required": ["value"]
5130                     }
5131
5132                 example: |
5133                     {
5134                         "rt": "oic.r.button",
5135                         "id": "unique_example_id",
5136                         "value": true
5137                     }
5138

```

### 5139 6.28.5 Property Definition

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

### 5140 6.28.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/ButtonResURI		get			

## 5141 6.29 Carbon Dioxide Sensor

### 5142 6.29.1 Introduction

5143 This resource describes whether carbon dioxide has been sensed or not. The value is a boolean.  
5144 A value of True means that carbon dioxide has been detected. A value of False means that  
5145 carbon dioxide has not been detected.

### 5146 6.29.2 Example URI

5147 /CarbonDioxideResURI

### 5148 6.29.3 Resource Type

5149 The resource type (rt) is defined as: oic.r.sensor.carbonDioxide.

### 5150 6.29.4 RAML Definition

```

5151 #%RAML 0.8
5152 title: OICCarbonDioxideSensor
5153 version: v1.0-20157527
5154 traits:
5155     - interface :

```

```

5156     queryParameters:
5157         if:
5158             enum: ["oic.if.s"]
5159
5160 /CarbonDioxideResURI:
5161     description: |
5162         This resource describes whether carbon dioxide has been sensed or not.
5163         The value is a boolean.
5164         A value of True means that carbon dioxide has been detected.
5165         A value of False means that carbon dioxide has not been detected.
5166
5167     is : ['interface']
5168     get:
5169         responses :
5170             200:
5171                 body:
5172                     application/json:
5173                         schema: |
5174                             {
5175                                 "id": "http://openinterconnect.org/schemas/oic.r.sensor.carbonDioxide.json#",
5176                                 "$schema": "http://json-schema.org/draft-04/schema#",
5177                                 "title": "Carbon Dioxide Sensor",
5178                                 "definitions": {
5179                                     "oic.r.sensor.carbonDioxide": {
5180                                         "allOf": [
5181                                             {"$ref": "oic.r.sensor.json#/definitions/oic.r.sensor"}
5182                                         ]
5183                                     }
5184                                 },
5185                                 "type": "object",
5186                                 "allOf": [
5187                                     {"$ref": "oic.core.json#/definitions/oic.core"},
5188                                     {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
5189                                     {"$ref": "#/definitions/oic.r.sensor.carbonDioxide"}
5190                                 ],
5191                                 "required": ["value"]
5192                             }
5193
5194                         example: |
5195                             {
5196                                 "rt": "oic.r.sensor.carbonDioxide",
5197                                 "id": "unique_example_id",
5198                                 "value": true
5199                             }
5200

```

### 5201 6.29.5 Property Definition

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

### 5202 6.29.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/CarbonDioxideResURI		get			

## 5203 6.30 Carbon Monoxide Sensor

### 5204 6.30.1 Introduction

5205 This resource describes whether carbon monoxide has been sensed or not. The value is a  
5206 boolean. A value of True means that carbon monoxide has been detected. A value of False  
5207 means that carbon monoxide has not been detected.

### 5208 6.30.2 Example URI

5209 /CarbonMonoxideResURI

### 5210 6.30.3 Resource Type

5211 The resource type (rt) is defined as: oic.r.sensor.carbonMonoxide.

### 5212 6.30.4 RAML Definition

5213 `##RAML 0.8`

5214 `title: OICCarbonMonoxideSensor`

5215 `version: v1.0-20150727`

5216 `traits:`

5217 `- interface :`

5218  `queryParameters:`

5219  `if:`

5220  `enum: ["oic.if.s"]`

5221

5222 `/CarbonMonoxideResURI:`

5223  `description: |`

5224  `This resource describes whether carbon monoxide has been sensed or not.`

5225  `The value is a boolean.`

5226  `A value of True means that carbon monoxide has been detected.`

5227  `A value of False means that carbon monoxide has not been detected.`

5228

5229  `is : ['interface']`

5230  `get:`

5231  `responses :`

5232  `200:`

5233  `body:`

5234  `application/json:`

5235  `schema: |`

5236  `{`

5237  `"id": "http://openinterconnect.org/schemas/oic.r.sensor.carbonMonoxide.json#",`

5238  `"$schema": "http://json-schema.org/draft-04/schema#",`

5239  `"title": "Carbon Monoxide Sensor",`

5240  `"definitions": {`

5241  `"oic.r.sensor.carbonMonoxide": {`

5242  `"allOf": [`

5243  `{"$ref": "oic.r.sensor.json#/definitions/oic.r.sensor"}`

5244  `]`

5245  `}`

5246  `},`

5247  `"type": "object",`

5248  `"allOf": [`

5249  `{"$ref": "oic.core.json#/definitions/oic.core"},`

5250  `{"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},`

5251  `{"$ref": "#/definitions/oic.r.sensor.carbonMonoxide"}`

5252  `],`

5253  `"required": ["value"]`

5254  `}`

5255

5256  `example: |`

5257  `{`

5258  `"rt": "oic.r.sensor.carbonMonoxide",`

```

5259         "id":      "unique_example_id",
5260         "value":   true
5261     }
5262

```

### 5263 6.30.5 Property Definition

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

### 5264 6.30.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/CarbonMonoxideResURI		get			

### 5265 6.31 Auto White Balance

#### 5266 6.31.1 Introduction

5267 This resource describes an auto balance on/off feature (on/off). The value is a boolean. An  
 5268 AutoWhiteBalance value of 'true' means that the switch is on. An AutoWhiteBalance value of  
 5269 'false' means that the switch is off.

#### 5270 6.31.2 Example URI

5271 /AutoWhiteBalanceResURI

#### 5272 6.31.3 Resource Type

5273 The resource type (rt) is defined as: oic.r.colour.autowhitebalance.

#### 5274 6.31.4 RAML Definition

```

5275 #%RAML 0.8
5276 title: OICAutoWhiteBalance
5277 version: v1.0-20150727
5278 traits:
5279   - interface :
5280     queryParameters:
5281       if:
5282         enum: ["oic.if.a"]
5283
5284 /AutoWhiteBalanceResURI:
5285   description: |
5286     This resource describes an auto balance on/off feature (on/off).
5287     The value is a boolean.
5288     An AutoWhiteBalance value of 'true' means that the switch is on.
5289     An AutoWhiteBalance value of 'false' means that the switch is off.
5290
5291   is : ['interface']
5292   get:
5293     responses :
5294       200:
5295         body:
5296           application/json:
5297             schema: |
5298               {
5299                 "id": "http://openinterconnect.org/schemas/oic.r.colour.autowhitebalance#",
5300                 "$schema": "http://json-schema.org/draft-04/schema#",
5301                 "title": "Auto White Balance",
5302                 "definitions": {
5303                   "oic.r.colour.autowhitebalance": {
5304                     "type": "object",

```

```

5305         "properties": {
5306             "autoWhiteBalance": {
5307                 "type": "boolean",
5308                 "description": "Status of the Auto White balance"
5309             }
5310         }
5311     },
5312 },
5313 "type": "object",
5314 "allOf": [
5315     {"$ref": "oic.core.json#/definitions/oic.core"},
5316     {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
5317     {"$ref": "#/definitions/oic.r.colour.autowhitebalance"}
5318 ],
5319 "required": [ "autoWhiteBalance" ]
5320 }
5321
5322 example: |
5323     {
5324         "rt":             "oic.r.colour.autowhitebalance",
5325         "id":             "unique_example_id",
5326         "autoWhiteBalance": false
5327     }
5328
5329 post:
5330 body:
5331     application/json:
5332     schema: |
5333         {
5334             "id": "http://openinterconnect.org/schemas/oic.r.colour.autowhitebalance#",
5335             "$schema": "http://json-schema.org/draft-04/schema#",
5336             "title": "Auto White Balance",
5337             "definitions": {
5338                 "oic.r.colour.autowhitebalance": {
5339                     "type": "object",
5340                     "properties": {
5341                         "autoWhiteBalance": {
5342                             "type": "boolean",
5343                             "description": "Status of the Auto White balance"
5344                         }
5345                     }
5346                 },
5347             },
5348             "type": "object",
5349             "allOf": [
5350                 {"$ref": "oic.core.json#/definitions/oic.core"},
5351                 {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
5352                 {"$ref": "#/definitions/oic.r.colour.autowhitebalance"}
5353             ],
5354             "required": [ "autoWhiteBalance" ]
5355         }
5356
5357 example: |
5358     {
5359         "id":             "unique_example_id",
5360         "autoWhiteBalance": true
5361     }
5362
5363 responses :
5364     200:
5365     body:
5366     application/json:
5367     schema: |

```

```

5368     {
5369         "id": "http://openinterconnect.org/schemas/oic.r.colour.autowhitebalance#",
5370         "$schema": "http://json-schema.org/draft-04/schema#",
5371         "title": "Auto White Balance",
5372         "definitions": {
5373             "oic.r.colour.autowhitebalance": {
5374                 "type": "object",
5375                 "properties": {
5376                     "autoWhiteBalance": {
5377                         "type": "boolean",
5378                         "description": "Status of the Auto White balance"
5379                     }
5380                 }
5381             }
5382         },
5383         "type": "object",
5384         "allOf": [
5385             {"$ref": "oic.core.json#/definitions/oic.core"},
5386             {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
5387             {"$ref": "#/definitions/oic.r.colour.autowhitebalance"}
5388         ],
5389         "required": [ "autoWhiteBalance" ]
5390     }
5391
5392     example: |
5393         {
5394             "id": "unique_example_id",
5395             "autoWhiteBalance": true
5396         }
5397

```

### 5398 6.31.5 Property Definition

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

### 5399 6.31.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/AutoWhiteBalanceResURI		get	post		

## 5400 6.32 Colour Saturation

### 5401 6.32.1 Introduction

5402 This resource describes a Colour saturation value. The value is an integer. A coloursaturation  
5403 has a range of [0,100]. A coloursaturation value of 0 means producing black and white images. A  
5404 coloursaturation value of 50 means producing device specific normal colour images. A  
5405 coloursaturation value of 100 means producing device very full colour images.

### 5406 6.32.2 Example URI

5407 /ColourSaturationResURI

### 5408 6.32.3 Resource Type

5409 The resource type (rt) is defined as: oic.r.colour.saturation.

### 5410 6.32.4 RAML Definition

```

5411 #%RAML 0.8
5412 title: OICColourSaturation
5413 version: v1.0-20150727
5414 traits:
5415   - interface :
5416       queryParameters:
5417         if:
5418             enum: ["oic.if.a"]

```

```

5419
5420 /ColourSaturationResURI:
5421 description: |
5422     This resource describes a Colour saturation value.
5423     The value is an integer.
5424     A coloursaturation has a range of [0,100].
5425     A coloursaturation value of 0 means producing black and white images.
5426     A coloursaturation value of 50 means producing device specific normal colour images.
5427     A coloursaturation value of 100 means producing device very full colour images.
5428
5429 is : ['interface']
5430 get:
5431     responses :
5432         200:
5433             body:
5434                 application/json:
5435                     schema: |
5436                         {
5437                             "id": "http://openinterconnect.org/schemas/oic.r.colour.saturation#",
5438                             "$schema": "http://json-schema.org/draft-04/schema#",
5439                             "title": "Colour Saturation",
5440                             "definitions": {
5441                                 "oic.r.colour.saturation": {
5442                                     "type": "object",
5443                                     "properties": {
5444                                         "colourSaturation": {
5445                                             "type": "integer",
5446                                             "description": "The colour saturation value",
5447                                             "minimum": 0,
5448                                             "maximum": 100
5449                                         }
5450                                     }
5451                                 }
5452                             },
5453                             "type": "object",
5454                             "allOf": [
5455                                 {"$ref": "oic.core.json#/definitions/oic.core"},
5456                                 {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
5457                                 {"$ref": "#/definitions/oic.r.colour.saturation"}
5458                             ],
5459                             "required": [ "colourSaturation" ]
5460                         }
5461
5462                     example: |
5463                         {
5464                             "rt": "oic.r.colour.saturation",
5465                             "id": "unique_example_id",
5466                             "colourSaturation": 50
5467                         }
5468
5469 post:
5470     body:
5471         application/json:
5472             schema: |
5473                 {
5474                     "id": "http://openinterconnect.org/schemas/oic.r.colour.saturation#",
5475                     "$schema": "http://json-schema.org/draft-04/schema#",
5476                     "title": "Colour Saturation",
5477                     "definitions": {
5478                         "oic.r.colour.saturation": {
5479                             "type": "object",
5480                             "properties": {
5481                                 "colourSaturation": {
5482                                     "type": "integer",

```



```

5483         "description": "The colour saturation value",
5484         "minimum": 0,
5485         "maximum": 100
5486     }
5487 }
5488 }
5489 },
5490 "type": "object",
5491 "allOf": [
5492     {"$ref": "oic.core.json#/definitions/oic.core"},
5493     {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
5494     {"$ref": "#/definitions/oic.r.colour.saturation"}
5495 ],
5496 "required": [ "colourSaturation" ]
5497 }
5498
5499 example: |
5500 {
5501     "id": "unique_example_id",
5502     "colourSaturation": 60
5503 }
5504
5505 responses :
5506 200:
5507   body:
5508     application/json:
5509       schema: |
5510         {
5511           "id": "http://openinterconnect.org/schemas/oic.r.colour.saturation#",
5512           "$schema": "http://json-schema.org/draft-04/schema#",
5513           "title": "Colour Saturation",
5514           "definitions": {
5515             "oic.r.colour.saturation": {
5516               "type": "object",
5517               "properties": {
5518                 "colourSaturation": {
5519                   "type": "integer",
5520                   "description": "The colour saturation value",
5521                   "minimum": 0,
5522                   "maximum": 100
5523                 }
5524             }
5525           },
5526           "type": "object",
5527           "allOf": [
5528             {"$ref": "oic.core.json#/definitions/oic.core"},
5529             {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
5530             {"$ref": "#/definitions/oic.r.colour.saturation"}
5531           ],
5532           "required": [ "colourSaturation" ]
5533         }
5534
5535 example: |
5536 {
5537     "id": "unique_example_id",
5538     "colourSaturation": 60
5539 }
5540
5541
5542

```

### 6.32.5 Property Definition

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

maximum				
---------	--	--	--	--

5543 **6.32.6 CRUDN behavior**

Resource	Create	Read	Update	Delete	Notify
/ColourSaturationResURI		get	post		

5544 **6.33 Contact Sensor**

5545 **6.33.1 Introduction**

5546 This resource describes whether a contact sensor has been tripped or not. Typical use case is in  
 5547 Security Systems detecting window or door open. The value is a boolean. A value of True means  
 5548 that contact has been broken (open). A value of False means that contact is in place (closed).

5549 **6.33.2 Example URI**

5550 /ContactResURI

5551 **6.33.3 Resource Type**

5552 The resource type (rt) is defined as: oic.r.sensor.contact.

5553 **6.33.4 RAML Definition**

```

5554 #%RAML 0.8
5555 title: OICContactSensor
5556 version: v1.0-20150727
5557 traits:
5558   - interface :
5559     queryParameters:
5560       if:
5561         enum: ["oic.if.s"]
5562
5563 /ContactResURI:
5564   description: |
5565     This resource describes whether a contact sensor has been tripped or not.
5566     Typical use case is in Security Systems detecting window or door open.
5567     The value is a boolean.
5568     A value of True means that contact has been broken (open).
5569     A value of False means that contact is in place (closed).
5570
5571   is : ['interface']
5572   get:
5573     responses :
5574       200:
5575         body:
5576           application/json:
5577             schema: |
5578               {
5579                 "id": "http://openinterconnect.org/schemas/oic.r.sensor.contact.json",
5580                 "$schema": "http://json-schema.org/draft-04/schema#",
5581                 "title": "Contact Sensor",
5582                 "definitions": {
5583                   "oic.r.sensor.contact": {
5584                     "allOf": [
5585                       {"$ref": "oic.r.sensor.json#/definitions/oic.r.sensor"}
5586                     ]
5587                   }
5588                 },
5589                 "type": "object",
5590                 "allOf": [
5591                   {"$ref": "oic.core.json#/definitions/oic.core"},
5592                   {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
5593                   {"$ref": "#/definitions/oic.r.sensor.contact"}
5594                 ]
5595               }

```

```

5594         ],
5595         "required": ["value"]
5596     }
5597
5598     example: |
5599         {
5600             "rt":      "oic.r.sensor.contact",
5601             "id":      "unique_example_id",
5602             "value":   true
5603         }
5604

```

### 5605 6.33.5 Property Definition

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

### 5606 6.33.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/ContactResURI		get			

## 5607 6.34 Demand Response Load Control (DRLC).

### 5608 6.34.1 Introduction

5609 This resource describes any to be applied or currently being applied DRLC signal. The DRType  
5610 is the ApplianceLoadReductionType defined in Zigbee/HA Smart Energy Profile 2.0. Start is a  
5611 string containing an ISO8601 encoded start time. The duration value is in minutes. Override  
5612 indicates whether the consumer has overridden the request (true) or not (false).

### 5613 6.34.2 Example URI

5614 /DRLCResURI

### 5615 6.34.3 Resource Type

5616 The resource type (rt) is defined as: oic.r.energy.drlc.

### 5617 6.34.4 RAML Definition

```

5618 #%RAML 0.8
5619 title: OICDRLC
5620 version: v1.0-20150727
5621 traits:
5622   - interface :
5623       queryParameters:
5624           if:
5625               enum: ["oic.if.lb"]
5626
5627 /DRLCResURI:
5628     description: |
5629         This resource describes any to be applied or currently being applied DRLC signal.
5630         The DRType is the ApplianceLoadReductionType defined in Zigbee/HA Smart Energy Profile 2.0.
5631         Start is a string containing an ISO8601 encoded start time.
5632         The duration value is in minutes.
5633         Override indicates whether the consumer has overridden the request (true) or not (false).
5634
5635     is : ['interface']
5636     get:
5637         description: |
5638             Provides the current DRLC action that is being applied.
5639

```

```

5640     responses :
5641         200:
5642             body:
5643                 application/json:
5644                     schema: |
5645                         {
5646                             "id": "http://openinterconnect.org/schemas/oic.r.energy.drlc#",
5647                             "$schema": "http://json-schema.org/draft-04/schema#",
5648                             "definitions": {
5649                                 "oic.r.energy.drlc": {
5650                                     "type": "object",
5651                                     "properties": {
5652                                         "DRType": {
5653                                             "type": "integer",
5654                                             "description": "The to be applied demand-response type"
5655                                         },
5656                                         "start": {
5657                                             "type": "string",
5658                                             "description": "The start time for the application of DR"
5659                                         },
5660                                         "duration": {
5661                                             "type": "integer",
5662                                             "description": "The duration of the to be applied DR type"
5663                                         },
5664                                         "override": {
5665                                             "type": "boolean",
5666                                             "description": "Whether the consumer has overridden the application of DR"
5667                                         }
5668                                     }
5669                                 }
5670                             },
5671                             "type": "object",
5672                             "allOf": [
5673                                 {"$ref": "oic.core.json#/definitions/oic.core"},
5674                                 {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
5675                                 {"$ref": "#/definitions/oic.r.energy.drlc"}
5676                             ],
5677                             "required": ["DRType"]
5678                         }
5679
5680                     example: |
5681                         {
5682                             "rt": "oic.r.energy.drlc",
5683                             "id": "unique_example_id",
5684                             "DRType": 1,
5685                             "start": "2015-01-09T16:45Z",
5686                             "duration": 10,
5687                             "override": false
5688                         }
5689
5690         put:
5691             description: |
5692                 Provides the DRLC action to be applied to the device or updates an existing action.
5693
5694             body:
5695                 application/json:
5696                     schema: |
5697                         {
5698                             "id": "http://openinterconnect.org/schemas/oic.r.energy.drlc#",
5699                             "$schema": "http://json-schema.org/draft-04/schema#",
5700                             "definitions": {
5701                                 "oic.r.energy.drlc": {
5702                                     "type": "object",
5703                                     "properties": {
5704                                         "DRType": {

```

```

5705         "type": "integer",
5706         "description": "The to be applied demand-response type"
5707     },
5708     "start": {
5709         "type": "string",
5710         "description": "The start time for the application of DR"
5711     },
5712     "duration": {
5713         "type": "integer",
5714         "description": "The duration of the to be applied DR type"
5715     },
5716     "override": {
5717         "type": "boolean",
5718         "description": "Whether the consumer has overridden the application of DR"
5719     }
5720 }
5721 }
5722 },
5723 "type": "object",
5724 "allOf": [
5725     {"$ref": "oic.core.json#/definitions/oic.core"},
5726     {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
5727     {"$ref": "#/definitions/oic.r.energy.drlc"}
5728 ],
5729 "required": ["DRType"]
5730 }
5731
5732 example: |
5733 {
5734     "rt":         "oic.r.energy.drlc",
5735     "id":         "unique_example_id",
5736     "DRType":    1,
5737     "start":     "2015-01-09T16:45Z",
5738     "duration": 10
5739 }
5740
5741 responses :
5742 200:
5743     description: |
5744         Indicates that the target DRLC resource was changed.
5745         The new resource attributes are provided in the response.
5746
5747     body:
5748     application/json:
5749         schema: |
5750             {
5751                 "id": "http://openinterconnect.org/schemas/oic.r.energy.drlc#",
5752                 "$schema": "http://json-schema.org/draft-04/schema#",
5753                 "definitions": {
5754                     "oic.r.energy.drlc": {
5755                         "type": "object",
5756                         "properties": {
5757                             "DRType": {
5758                                 "type": "integer",
5759                                 "description": "The to be applied demand-response type"
5760                             },
5761                             "start": {
5762                                 "type": "string",
5763                                 "description": "The start time for the application of DR"
5764                             },
5765                             "duration": {
5766                                 "type": "integer",
5767                                 "description": "The duration of the to be applied DR type"
5768                             },
5769                             "override": {
5770                                 "type": "boolean",
5771                                 "description": "Whether the consumer has overridden the application of DR"

```

```

5772     }
5773     }
5774   }
5775 },
5776 "type": "object",
5777 "allOf": [
5778   {"$ref": "oic.core.json#/definitions/oic.core"},
5779   {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
5780   {"$ref": "#/definitions/oic.r.energy.drlc"}
5781 ],
5782 "required": ["DRType"]
5783 }
5784
5785 example: |
5786 {
5787   "DRType": 1,
5788   "id": "unique_example_id",
5789   "start": "2015-01-09T17:00Z",
5790   "duration": 15,
5791   "override": false
5792 }
5793
5794 201:
5795 description: |
5796   Indicates successful creation of the DRLC resource with the attributes provided.
5797   The response includes the URI of the created resource.
5798
5799 body:
5800 application/json:
5801 schema: |
5802 {
5803   "id": "http://openinterconnect.org/schemas#",
5804   "$schema": "http://json-schema.org/draft-04/schema#",
5805   "type": "object",
5806   "definitions": {
5807     "oic.create": {
5808       "type": "object",
5809       "properties": {
5810         "ResURI": { "type": "string" }
5811       }
5812     }
5813   },
5814   "type": "object",
5815   "$ref": "#/definitions/oic.create"
5816 }
5817
5818 example: |
5819 {
5820   "ResURI": "/MyDevice/MyDRLCURI"
5821 }
5822

```

### 5823 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 overridden the application of DR

### 5824 6.34.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
----------	--------	------	--------	--------	--------

/DRLCResURI	put	get			
-------------	-----	-----	--	--	--

5825 **6.35 Energy Overload/Circuit Breaker**

5826 **6.35.1 Introduction**

5827 This resource describes whether an energy overload detector/circuit breaker has been tripped.  
 5828 The value is a boolean. A value of True means that energy overload has been tripped. A value of  
 5829 False means that energy overload has not been tripped.

5830 **6.35.2 Example URI**

5831 /EnergyOverloadResURI

5832 **6.35.3 Resource Type**

5833 The resource type (rt) is defined as: oic.r.energy.overload.

5834 **6.35.4 RAML Definition**

5835 `##RAML 0.8`

5836 `title: OICEnergyOverload`  
 5837 `version: v1.0-20150727`

5838 `traits:`  
 5839 `- interface :`  
 5840  `queryParameters:`  
 5841  `if:`  
 5842  `enum: ["oic.if.s"]`

5843

5844 `/EnergyOverloadResURI:`

5845  `description: |`  
 5846  `This resource describes whether an energy overload detector/circuit breaker`  
 5847  `has been tripped.`  
 5848  `The value is a boolean.`  
 5849  `A value of True means that energy overload has been tripped.`  
 5850  `A value of False means that energy overload has not been tripped.`  
 5851

5852  `is : ['interface']`

5853  `get:`  
 5854  `responses :`

5855  `200:`

5856  `body:`  
 5857  `application/json:`

5858  `schema: |`  
 5859  `{`  
 5860  `"id": "http://openinterconnect.org/schemas/oic.r.energy.overload.json#",`  
 5861  `"$schema": "http://json-schema.org/draft-04/schema#",`  
 5862  `"title": "Energy Overload Sensor",`  
 5863  `"definitions": {`  
 5864  `"oic.r.energy.overload": {`  
 5865  `"allOf": [`  
 5866  `{"$ref": "oic.r.sensor.json#/definitions/oic.r.sensor"}`  
 5867  `]`  
 5868  `}`  
 5869  `},`  
 5870  `"type": "object",`  
 5871  `"allOf": [`  
 5872  `{"$ref": "oic.core.json#/definitions/oic.core"},`  
 5873  `{"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},`  
 5874  `{"$ref": "#/definitions/oic.r.energy.overload"}`  
 5875  `],`  
 5876  `"required": ["value"]`  
 5877  `}`  
 5878

5879  `example: |`

```

5880     {
5881         "rt": "oic.r.energy.overload",
5882         "id": "unique_example_id",
5883         "value": true
5884     }
5885 
```

### 5886 6.35.5 Property Definition

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

### 5887 6.35.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/EnergyOverloadResURI		get			

## 5888 6.36 Generic Sensor

### 5889 6.36.1 Introduction

5890 This resource describes whether some value or property or entity has been sensed or not. The  
5891 value is a boolean. A value of True means that the target has been sensed. A value of False  
5892 means that the target has not been sensed.

### 5893 6.36.2 Example URI

5894 /GenericSensorResURI

### 5895 6.36.3 Resource Type

5896 The resource type (rt) is defined as: oic.r.sensor.

### 5897 6.36.4 RAML Definition

```

5898 #%RAML 0.8
5899 title: OICGenericSensor
5900 version: v1.0-20150727
5901 traits:
5902   - interface :
5903       queryParameters:
5904           if:
5905               enum: ["oic.if.s"]
5906
5907 /GenericSensorResURI:
5908     description: |
5909       This resource describes whether some value or property or entity has been sensed or not.
5910       The value is a boolean.
5911       A value of True means that the target has been sensed.
5912       A value of False means that the target has not been sensed.
5913
5914     is : ['interface']
5915     get:
5916       responses :
5917         200:
5918           body:
5919             application/json:
5920               schema: |
5921                 {
5922                   "id": "http://openinterconnect.org/schemas/oic.r.sensor.json#",
5923                   "$schema": "http://json-schema.org/draft-04/schema#",
5924                   "title": "Generic Sensor",
5925                   "definitions": {
```



```

5926         "oic.r.sensor": {
5927             "type": "object",
5928             "properties": {
5929                 "value": {
5930                     "type": "boolean",
5931                     "description": "ReadOnly, true = sensed, false = not sensed."
5932                 }
5933             }
5934         },
5935     },
5936     "type": "object",
5937     "allOf": [
5938         {"$ref": "oic.core.json#/definitions/oic.core"},
5939         {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
5940         {"$ref": "#/definitions/oic.r.sensor"}
5941     ]
5942 }
5943
5944 example: |
5945 {
5946     "rt": "oic.r.sensor",
5947     "id": "unique_example_id",
5948     "value": true
5949 }
5950

```

5951 **6.36.5 Property Definition**

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

5952 **6.36.6 CRUDN behavior**

Resource	Create	Read	Update	Delete	Notify
/GenericSensorResURI		get			

5953 **6.37 Glass Break Sensor**

5954 **6.37.1 Introduction**

5955 This resource describes a glass break sensor. The value is a boolean. A value of True means  
5956 that glass break has been sensed. A value of False means that glass break not been sensed.

5957 **6.37.2 Example URI**

5958 /GlassBreakResURI

5959 **6.37.3 Resource Type**

5960 The resource type (rt) is defined as: oic.r.sensor.glassBreak.

5961 **6.37.4 RAML Definition**

```

5962 #%RAML 0.8
5963 title: OICGlassBreakSensor
5964 version: v1.0-20150727
5965 traits:
5966   - interface :
5967       queryParameters:
5968           if:
5969               enum: ["oic.if.s"]
5970
5971 /GlassBreakResURI:
5972     description: |

```

```

5973     This resource describes a glass break sensor.
5974     The value is a boolean.
5975     A value of True means that glass break has been sensed.
5976     A value of False means that glass break not been sensed.
5977
5978     is : ['interface']
5979
5979     get:
5980         responses :
5981             200:
5982                 body:
5983                     application/json:
5984                         schema: |
5985                             {
5986                                 "id": "http://openinterconnect.org/schemas/oic.r.sensor.glassBreak.json#",
5987                                 "$schema": "http://json-schema.org/draft-04/schema#",
5988                                 "title": "Glass Break Sensor",
5989                                 "definitions": {
5990                                     "oic.r.sensor.glassBreak": {
5991                                         "allOf": [
5992                                             {"$ref": "oic.r.sensor.json#/definitions/oic.r.sensor"}
5993                                         ]
5994                                     }
5995                                 },
5996                                 "type": "object",
5997                                 "allOf": [
5998                                     {"$ref": "oic.core.json#/definitions/oic.core"},
5999                                     {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
6000                                     {"$ref": "#/definitions/oic.r.sensor.glassBreak"}
6001                                 ],
6002                                 "required": ["value"]
6003                             }
6004
6005                         example: |
6006                             {
6007                                 "rt": "oic.r.sensor.glassBreak",
6008                                 "id": "unique_example_id",
6009                                 "value": true
6010                             }
6011

```

### 6012 6.37.5 Property Definition

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

### 6013 6.37.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/GlassBreakResURI		get			

## 6014 6.38 Heart Rate Zone

### 6015 6.38.1 Introduction

6016 This resource describes a measured heart rate by the current Zone using the Zoladz method The  
6017 Zoladz method defines Zones based on maximum heart rate; Zone 1 is the lowest, Zone 5 is the  
6018 highest. The heartRateZone is an enumeration containing one of: "Zone1", "Zone2", "Zone3",  
6019 "Zone4", "Zone5".

### 6020 6.38.2 Example URI

6021 /HeartRateZoneResURI

6022 **6.38.3 Resource Type**

6023 The resource type (rt) is defined as: oic.r.sensor.heart.zone.

6024 **6.38.4 RAML Definition**

```
6025 #%RAML 0.8
6026 title: OICHeartRateZone
6027 version: v1.0-20150727
6028 traits:
6029   - interface :
6030     queryParameters:
6031       if:
6032         enum: ["oic.if.s"]
6033
6034 /HeartRateZoneResURI:
6035   description: |
6036     This resource describes a measured heart rate by the current Zone using the Zoladz method
6037     The Zoladz method defines Zones based on maximum heart rate; Zone 1 is the lowest, Zone 5 is
6038     the highest.
6039     The heartRateZone is an enumeration containing one of: "Zone1", "Zone2", "Zone3", "Zone4",
6040     "Zone5".
6041
6042   is : ['interface']
6043   get:
6044     responses :
6045       200:
6046         body:
6047           application/json:
6048             schema: |
6049               {
6050                 "id": "http://openinterconnect.org/schemas/oic.r.sensor.heart.zone.json#",
6051                 "$schema": "http://json-schema.org/draft-04/schema#",
6052                 "title": "Heart Rate Zone",
6053                 "definitions": {
6054                   "oic.r.sensor.heart.zone": {
6055                     "properties": {
6056                       "heartRateZone": {
6057                         "enum": ["Zone1","Zone2","Zone3", "Zone4", "Zone5"],
6058                         "description": "ReadOnly, current heart rate zone based on the Zoladz
6059 system."
6060                       }
6061                     }
6062                   }
6063                 },
6064                 "type": "object",
6065                 "allOf": [
6066                   {"$ref": "oic.core.json#/definitions/oic.core"},
6067                   {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
6068                   {"$ref": "#/definitions/oic.r.sensor.heart.zone"}
6069                 ],
6070                 "required": ["heartRateZone"]
6071               }
6072
6073   example: |
6074     {
6075       "rt": "oic.r.sensor.heart.zone",
6076       "id": "unique_example_id",
6077       "heartRateZone": "Zone3"
6078     }
6079
```

6080 **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.

6081 **6.38.6 CRUDN behavior**

Resource	Create	Read	Update	Delete	Notify
/HeartRateZoneResURI		get			

6082 **6.39 Illuminance Sensor**

6083 **6.39.1 Introduction**

6084 This resource describes an illuminance sensor illuminance is a float and represents the sensed  
6085 luminous flux per unit area in lux.

6086 **6.39.2 Example URI**

6087 /IlluminanceSensorResURI

6088 **6.39.3 Resource Type**

6089 The resource type (rt) is defined as: oic.r.sensor.illuminance.

6090 **6.39.4 RAML Definition**

```

6091 #RAML 0.8
6092 title: OICilluminanceSensor
6093 version: v1.0-20150727
6094 traits:
6095   - interface :
6096     queryParameters:
6097       if:
6098         enum: ["oic.if.s"]
6099
6100 /IlluminanceSensorResURI:
6101   description: |
6102     This resource describes an illuminance sensor
6103     illuminance is a float and represents the sensed luminous flux per unit area in lux.
6104
6105   is : ['interface']
6106   get:
6107     responses :
6108       200:
6109         body:
6110           application/json:
6111             schema: |
6112               {
6113                 "id": "http://openinterconnect.org/schemas/oic.r.sensor.illuminance.json#",
6114                 "$schema": "http://json-schema.org/draft-04/schema#",
6115                 "title": "Illuminance Sensor",
6116                 "definitions": {
6117                   "oic.r.sensor.illuminance": {
6118                     "properties": {
6119                       "illuminance": {
6120                         "type": "number",
6121                         "description": "ReadOnly, sensed luminous flux per unit area in lux."
6122                       }
6123                     }
6124                   }
6125                 },
6126                 "type": "object",

```

```

6127         "allOf": [
6128             { "$ref": "oic.core.json#/definitions/oic.core" },
6129             { "$ref": "oic.baseResource.json#/definitions/oic.r.baseResource" },
6130             { "$ref": "#/definitions/oic.r.sensor.illuminance" }
6131         ],
6132         "required": ["illuminance"]
6133     }
6134
6135     example: |
6136         {
6137             "rt":          "oic.r.sensor.illuminance",
6138             "id":          "unique_example_id",
6139             "illuminance": 450
6140         }
6141

```

### 6142 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.

### 6143 6.39.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/IlluminanceSensorResURI		get			

## 6144 6.40 Magnetic Field Direction Sensor

### 6145 6.40.1 Introduction

6146 This resource describes the direction of the Earth's magnetic field at the observer's current point  
6147 in space. Typical use case includes measurement of compass readings on a personal device.  
6148 The value is a CSV of Hx, Hy, Hz. Each of Hx, Hy and Hz are expressed in A/m (Amperes per  
6149 metre)

### 6150 6.40.2 Example URI

6151 /MagneticFieldDirectionResURI

### 6152 6.40.3 Resource Type

6153 The resource type (rt) is defined as: oic.r.sensor.magneticFieldDirection.

### 6154 6.40.4 RAML Definition

```

6155 #%RAML 0.8
6156 title: OICMagneticFieldDirection
6157 version: v1.0-20150727
6158 traits:
6159   - interface :
6160       queryParameters:
6161           if:
6162               enum: ["oic.if.s"]
6163
6164 /MagneticFieldDirectionResURI:
6165     description: |
6166         This resource describes the direction of the Earth's magnetic field at the observer's current
6167         point in space.
6168         Typical use case includes measurement of compass readings on a personal device.
6169         The value is a CSV of Hx, Hy, Hz.
6170         Each of Hx, Hy and Hz are expressed in A/m (Amperes per metre)
6171
6172     is : ['interface']
6173     get:

```

```

6174     responses :
6175         200:
6176             body:
6177                 application/json:
6178                 schema: |
6179                     {
6180                         "id":
6181 "http://openinterconnect.org/schemas/oic.r.sensor.magneticFieldDirection.json#",
6182                         "$schema": "http://json-schema.org/draft-04/schema#",
6183                         "title": "Magnetic Field Direction Sensor",
6184                         "definitions": {
6185                             "oic.r.sensor.magneticFieldDirection": {
6186                                 "properties": {
6187                                     "value": {
6188                                         "type": "string",
6189                                         "description": "ReadOnly, CSV containing Hx, Hy, Hz."
6190                                     }
6191                                 }
6192                             }
6193                         },
6194                         "type": "object",
6195                         "allOf": [
6196                             {"$ref": "oic.core.json#/definitions/oic.core"},
6197                             {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
6198                             {"$ref": "#/definitions/oic.r.sensor.magneticFieldDirection"}
6199                         ],
6200                         "required": ["value"]
6201                     }
6202
6203                 example: |
6204                     {
6205                         "rt": "oic.r.sensor.magneticFieldDirection",
6206                         "id": "unique_example_id",
6207                         "value": "100,15,90"
6208                     }
6209

```

#### 6210 6.40.5 Property Definition

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

#### 6211 6.40.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/MagneticFieldDirectionResURI		get			

### 6212 6.41 Media

#### 6213 6.41.1 Introduction

6214 This resource specifies the media that an OIC Server (Camera) supports. The resource is an  
6215 array of media elements. Each element contains: A URL at which the specified media type  
6216 can be accessed. A string array containing the definition of the media using SDP. Each  
6217 entry in the sdp array is an SDP line. Each line shall follow the SDP description syntax as  
6218 defined in the SDP specification. The SDP specification can be found at  
6219 <http://tools.ietf.org/html/rfc4566>. The mime subtype video/H264 indicates video resource and the  
6220 mime subtype video/jpeg indicates still image resource.

#### 6221 6.41.2 Example URI

6222 /MediaResURI

#### 6223 6.41.3 Resource Type

6224 The resource type (rt) is defined as: oic.r.media.

#### 6225 6.41.4 RAML Definition

```
6226 #%RAML 0.8
6227 title: OICMedia
6228 version: v1.0-20150727
6229 traits:
6230   - interface :
6231     queryParameters:
6232       if:
6233         enum: ["oic.if.s"]
6234
6235 /MediaResURI:
6236   description: |
6237     This resource specifies the media that an OIC Server (Camera) supports.
6238     The resource is an array of media elements
6239     Each element contains:
6240       A URL at which the specified media type can be accessed.
6241       A string array containing the definition of the media using SDP.
6242       Each entry in the sdp array is an SDP line.
6243       Each line shall follow the SDP description syntax as defined in the SDP specification.
6244       The SDP specification can be found at http://tools.ietf.org/html/rfc4566.
6245       The mime subtype video/H264 indicates video resource and the mime subtype video/jpeg indicates
6246       still image resource.
6247
6248   is : ['interface']
6249   get:
6250     description: |
6251       Retrieves the current media resource.
6252
6253   responses :
6254     200:
6255       body:
6256         application/json:
6257           schema: |
6258             {
6259               "id": "http://openinterconnect.org/schemas/oic.r.media#",
6260               "$schema": "http://json-schema.org/draft-04/schema#",
6261               "title": "Media",
6262               "definitions": {
6263                 "oic.r.media": {
6264                   "media": {
6265                     "type": "array",
6266                     "items": {
6267                       "type": "object",
6268                       "properties": {
6269                         "url": {
6270                           "type": "string",
6271                           "description": "url for the media instance"
6272                         },
6273                         "sdp": {
6274                           "type": "array",
6275                           "description": "Array of strings, one per SDP line",
6276                           "items": {
6277                             "type": "string",
6278                             "description": "SDP media or attribute line"
6279                           }
6280                         }
6281                       }
6282                     }
6283                   }
6284                 }
6285               },
6286               "type": "object",
```

```

6287     "allOf": [
6288         { "$ref": "oic.core.json#/definitions/oic.core" },
6289         { "$ref": "oic.baseResource.json#/definitions/oic.r.baseResource" },
6290         { "$ref": "#/definitions/oic.r.media" }
6291     ],
6292     "required": ["media"]
6293 }
6294
6295 example: |
6296 {
6297     "rt": "oic.r.media",
6298     "id": "unique_example_id",
6299     "media": [
6300         {
6301             "url": "some example url",
6302             "sdp": [
6303                 "m=video 1 RTP/AVP 96",
6304                 "a=rtpmap:96 H264/9000",
6305                 "a=fmtp:96 profile-level-id=42A028;packetization-mode=1"
6306             ]
6307         },
6308         {
6309             "url": "some other example1 url",
6310             "sdp": [
6311                 "m=audio 2 RTP/AVP 97",
6312                 "a=rtpmap:97 MP4A-LATM/90000"
6313             ]
6314         },
6315         {
6316             "url": "some other example2 url",
6317             "sdp": [
6318                 "m=video 3 RTP/AVP 98",
6319                 "a=rtpmap:98 jpeg/90000",
6320                 "a=fmtp:98 sampling=YCbCr-4:2:0;width=256;height=256"
6321             ]
6322         }
6323     ]
6324 }
6325

```

### 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 of oic.r.media.input and one with a resource type of oic.r.media.output



6339 **6.42.2 Example URI**

6340 /mediaSourceListResURI

6341 **6.42.3 Resource Type**

6342 The resource type (rt) is defined as: oic.r.mediaSourceList.

6343 **6.42.4 RAML Definition**

6344 #%RAML 0.8

6345 title: OICMediaSourceList

6346 version: v1.0-20151019

6347 traits:

6348 - interface :

6349 queryParameters:

6350 if:

6351 enum: ["oic.if.a"]

6352

6353 /mediaSourceListResURI:

6354 description: |

6355 This resource provides the list of media sources available on the device.

6356 The sources are an array of mediaSource(s) as separately defined.

6357 The basic resource type oic.r.mediaSourceList does not provide any indications whether the  
6358 source is input or output.

6359 Hence, two specializations of this resource exist.

6360 When a device exposes input sources then an instance of this resource with a resource type of  
6361 oic.r.media.input is exposed.

6362 When a device exposes output sources then an instance of this resource with a resource type of  
6363 oic.r.media.output is exposed.

6364 A device that exposes both input and output media sources then exposes two instances of this  
6365 resource,

6366 one with a resource type of oic.r.media.input and one with a resource type of

6367 oic.r.media.output

6368

6369 is : ['interface']

6370 get:

6371 responses :

6372 200:

6373 body:

6374 application/json:

6375 schema: |

```
6376 {
6377   "$schema": "http://json-schemas.org/draft-04/schema#",
6378   "id": "http://openinterconnect.org/schemas/oic.r.mediaSourceList.json#",
6379   "title": "Media Source List",
6380   "definitions": {
6381     "oic.r.mediaSourceList": {
6382       "properties": {
6383         "sources": {
6384           "type": "array",
6385           "items": {
6386             "oneOf": [
6387               { "$ref": "oic.r.mediaSource.json#" }
6388             ]
6389           }
6390         }
6391       }
6392     },
6393     "type": "object",
6394     "allOf": [
6395       { "$ref": "oic.core.json#/definitions/oic.core" },
6396       { "$ref": "oic.baseResource.json#/definitions/oic.r.baseResource" },
6397       { "$ref": "#/definitions/oic.r.mediaSourceList" }
6398     ]
6399   }
6400 }
```

```

6399         ],
6400         "required": ["sources"]
6401     }
6402
6403     example: |
6404         {
6405             "rt": "oic.r.mediaSourceList",
6406             "id": "unique_example_id",
6407             "sources": [
6408                 {
6409                     "sourceName": "HDMI-CEC",
6410                     "sourceNumber": "1",
6411                     "sourceType": "audioPlusVideo",
6412                     "status": true
6413                 },
6414                 {
6415                     "sourceName": "dualRCA",
6416                     "sourceNumber": "1",
6417                     "sourceType": "audioOnly",
6418                     "status": false
6419                 }
6420             ]
6421         }
6422
6423     post:
6424         description: |
6425             Changes the status of the source(s).
6426             Allows changes of the sourceName and the status.
6427
6428     body:
6429         application/json:
6430             schema: |
6431                 {
6432                     "$schema": "http://json-schemas.org/draft-04/schema#",
6433                     "id": "http://openinterconnect.org/schemas/oic.r.mediaSourceList.json#",
6434                     "title": "Media Source List",
6435                     "definitions": {
6436                         "oic.r.mediaSourceList": {
6437                             "properties": {
6438                                 "sources": {
6439                                     "type": "array",
6440                                     "items": {
6441                                         "oneOf": [
6442                                             { "$ref": "oic.r.mediaSource.json#" }
6443                                         ]
6444                                     }
6445                                 }
6446                             }
6447                         }
6448                     },
6449                     "type": "object",
6450                     "allOf": [
6451                         { "$ref": "oic.core.json#/definitions/oic.core" },
6452                         { "$ref": "oic.baseResource.json#/definitions/oic.r.baseResource" },
6453                         { "$ref": "#/definitions/oic.r.mediaSourceList" }
6454                     ],
6455                     "required": ["sources"]
6456                 }
6457
6458     example: |
6459         {
6460             "id": "unique_example_id",
6461             "sources": [
6462                 {
6463                     "sourceName": "my new name",
6464                     "sourceNumber": "1",

```

```

6465         "status":      true
6466     }
6467 ]
6468 }
6469
6470 responses :
6471     200:
6472     body:
6473     application/json:
6474     schema: |
6475     {
6476     "$schema": "http://json-schemas.org/draft-04/schema#",
6477     "id": "http://openinterconnect.org/schemas/oic.r.mediaSourceList.json#",
6478     "title": "Media Source List",
6479     "definitions": {
6480     "oic.r.mediaSourceList": {
6481     "properties": {
6482     "sources": {
6483     "type": "array",
6484     "items": {
6485     "oneOf": [
6486     { "$ref": "oic.r.mediaSource.json#" }
6487     ]
6488     }
6489     }
6490     }
6491     },
6492     },
6493     "type": "object",
6494     "allOf": [
6495     { "$ref": "oic.core.json#/definitions/oic.core" },
6496     { "$ref": "oic.baseResource.json#/definitions/oic.r.baseResource" },
6497     { "$ref": "#/definitions/oic.r.mediaSourceList" }
6498     ],
6499     "required": ["sources"]
6500     }
6501
6502     example: |
6503     {
6504     "id":      "unique_example_id",
6505     "sources": [
6506     {
6507     "sourceName":  "my new name",
6508     "sourceNumber": "1",
6509     "status":      true
6510     }
6511     ]
6512     }
6513

```

### 6514 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

6515 **6.42.6 CRUDN behavior**

Resource	Create	Read	Update	Delete	Notify
/mediaSourceListResURI		get	post		

6516 **6.42.7 Referenced JSON schemas**

6517 **6.42.7.1 oic.r.mediaSource.json**

```

6518 {
6519   "$schema": "http://json-schema.org/draft-04/schema#",
6520   "id": "http://openinterconnect.org/schemas/oic.r.mediaSource.json#",
6521   "title": "Media Source",
6522   "definitions": {
6523     "oic.r.mediaSource": {
6524       "properties": {
6525         "sourceName": {
6526           "type": "string",
6527           "description": "Specifies a pre-defined media input or output"
6528         },
6529         "sourceNumber": {
6530           "type": [ "integer", "string" ],
6531           "description": "ReadOnly, Numeric identifier to specify the instance"
6532         },
6533         "sourceType": {
6534           "enum": [ "audioOnly", "videoOnly", "audioPlusVideo" ],
6535           "description": "ReadOnly, Specifies the type of the source"
6536         },
6537         "status": {
6538           "type": "boolean",
6539           "description": "Specifies if the specific source instance is selected or not"
6540         }
6541       }
6542     },
6543   },
6544   "type": "object",
6545   "allOf": [
6546     { "$ref": "oic.core.json#/definitions/oic.core" },
6547     { "$ref": "oic.baseResource.json#/definitions/oic.r.baseResource" },
6548     { "$ref": "#/definitions/oic.r.mediaSource" }
6549   ],
6550   "required": [ "sourceName", "status" ]
6551 }
6552

```

6553 **6.43 Motion Sensor**

6554 **6.43.1 Introduction**

6555 This resource describes whether motion has been sensed or not. The value is a boolean. A value  
 6556 of True means that motion has been sensed. A value of False means that motion not been  
 6557 sensed.

6558 **6.43.2 Example URI**

6559 /MotionResURI

6560 **6.43.3 Resource Type**

6561 The resource type (rt) is defined as: oic.r.sensor.motion.

6562 **6.43.4 RAML Definition**

```

6563 #%RAML 0.8
6564 title: OICMotionSensor
6565 version: v1.0-20150727
6566 traits:
6567   - interface :
6568     queryParameters:
6569       if:
6570         enum: [ "oic.if.s" ]

```

```

6571
6572 /MotionResURI:
6573   description: |
6574     This resource describes whether motion has been sensed or not.
6575     The value is a boolean.
6576     A value of True means that motion has been sensed.
6577     A value of False means that motion not been sensed.
6578
6579   is : ['interface']
6580   get:
6581     responses :
6582       200:
6583         body:
6584           application/json:
6585             schema: |
6586               {
6587                 "id": "http://openinterconnect.org/schemas/oic.r.sensor.motion.json#",
6588                 "$schema": "http://json-schema.org/draft-04/schema#",
6589                 "title": "Motion Sensor",
6590                 "definitions": {
6591                   "oic.r.sensor.motion": {
6592                     "allOf": [
6593                       {"$ref": "oic.r.sensor.json#/definitions/oic.r.sensor"}
6594                     ]
6595                   }
6596                 },
6597                 "type": "object",
6598                 "allOf": [
6599                   {"$ref": "oic.core.json#/definitions/oic.core"},
6600                   {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
6601                   {"$ref": "#/definitions/oic.r.sensor.motion"}
6602                 ],
6603                 "required": ["value"]
6604               }
6605
6606           example: |
6607             {
6608               "rt": "oic.r.sensor.motion",
6609               "id": "unique_example_id",
6610               "value": true
6611             }
6612

```

### 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

6621 **6.44.3 Resource Type**

6622 The resource type (rt) is defined as: oic.r.nightMode.

6623 **6.44.4 RAML Definition**

6624 `##RAML 0.8`

6625 `title: OICNightMode`  
6626 `version: v1.0-20150727`

6627 `traits:`

6628 `- interface :`  
6629  `queryParameters:`  
6630  `if:`  
6631  `enum: ["oic.if.a"]`

6632

6633 `/NightModeResURI:`

6634  `description: |`  
6635  `This resource describes a night mode on/off feature (on/off).`  
6636  `A nightMode value of 'true' means that the feature is on.`  
6637  `A nightMode value of 'false' means that the feature is off.`  
6638

6639  `is : ['interface']`

6640  `get:`

6641  `responses :`

6642  `200:`

6643  `body:`

6644  `application/json:`

6645  `schema: |`

6646  `{`  
6647  `"id": "http://openinterconnect.org/schemas/oic.r.nightMode.json#",`  
6648  `"$schema": "http://json-schema.org/draft-04/schema#",`  
6649  `"title": "Night Mode",`  
6650  `"definitions": {`  
6651  `"oic.r.nightMode": {`  
6652  `"type": "object",`  
6653  `"properties": {`  
6654  `"nightMode": {`  
6655  `"type": "boolean",`  
6656  `"description": "Status of the Night Mode"`  
6657  `}`  
6658  `}`  
6659  `}`  
6660  `},`  
6661  `"type": "object",`  
6662  `"allOf": [`  
6663  `{"$ref": "oic.core.json#/definitions/oic.core"},`  
6664  `{"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},`  
6665  `{"$ref": "#/definitions/oic.r.nightMode"}`  
6666  `],`  
6667  `"required": [ "nightMode" ]`  
6668  `}`  
6669

6670  `example: |`

6671  `{`  
6672  `"rt": "oic.r.nightMode",`  
6673  `"id": "unique_example_id",`  
6674  `"nightMode": false`  
6675  `}`  
6676

6677  `post:`

6678  `body:`

6679  `application/json:`

```

6680     schema: |
6681         {
6682             "id": "http://openinterconnect.org/schemas/oic.r.nightMode.json#",
6683             "$schema": "http://json-schema.org/draft-04/schema#",
6684             "title": "Night Mode",
6685             "definitions": {
6686                 "oic.r.nightMode": {
6687                     "type": "object",
6688                     "properties": {
6689                         "nightMode": {
6690                             "type": "boolean",
6691                             "description": "Status of the Night Mode"
6692                         }
6693                     }
6694                 },
6695             },
6696             "type": "object",
6697             "allOf": [
6698                 { "$ref": "oic.core.json#/definitions/oic.core" },
6699                 { "$ref": "oic.baseResource.json#/definitions/oic.r.baseResource" },
6700                 { "$ref": "#/definitions/oic.r.nightMode" }
6701             ],
6702             "required": [ "nightMode" ]
6703         }
6704
6705     example: |
6706         {
6707             "id": "unique_example_id",
6708             "nightMode": true
6709         }
6710
6711     responses :
6712         200:
6713             body:
6714                 application/json:
6715                     schema: |
6716                         {
6717                             "id": "http://openinterconnect.org/schemas/oic.r.nightMode.json#",
6718                             "$schema": "http://json-schema.org/draft-04/schema#",
6719                             "title": "Night Mode",
6720                             "definitions": {
6721                                 "oic.r.nightMode": {
6722                                     "type": "object",
6723                                     "properties": {
6724                                         "nightMode": {
6725                                             "type": "boolean",
6726                                             "description": "Status of the Night Mode"
6727                                         }
6728                                     }
6729                                 }
6730                             },
6731                             "type": "object",
6732                             "allOf": [
6733                                 { "$ref": "oic.core.json#/definitions/oic.core" },
6734                                 { "$ref": "oic.baseResource.json#/definitions/oic.r.baseResource" },
6735                                 { "$ref": "#/definitions/oic.r.nightMode" }
6736                             ],
6737                             "required": [ "nightMode" ]
6738                         }
6739
6740                     example: |
6741                         {
6742                             "id": "unique_example_id",
6743                             "nightMode": true

```

6744 }  
6745

#### 6746 6.44.5 Property Definition

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

#### 6747 6.44.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/NightModeResURI		get	post		

#### 6748 6.45 Presence Sensor

##### 6749 6.45.1 Introduction

6750 This resource describes whether presence has been sensed or not. The value is a boolean. A  
6751 value of True means that presence has been sensed. A value of False means that presence not  
6752 been sensed.

##### 6753 6.45.2 Example URI

6754 /PresenceResURI

##### 6755 6.45.3 Resource Type

6756 The resource type (rt) is defined as: oic.r.sensor.presence.

##### 6757 6.45.4 RAML Definition

6758 `##RAML 0.8`

6759 `title: OICPresenceSensor`  
6760 `version: v1.0-20150727`

6761 `traits:`

6762 `- interface :`

6763  `queryParameters:`

6764  `if:`

6765  `enum: ["oic.if.s"]`

6766

6767 `/PresenceResURI:`

6768  `description: |`

6769  `This resource describes whether presence has been sensed or not.`

6770  `The value is a boolean.`

6771  `A value of True means that presence has been sensed.`

6772  `A value of False means that presence not been sensed.`

6773

6774  `is : ['interface']`

6775  `get:`

6776  `responses :`

6777  `200:`

6778  `body:`

6779  `application/json:`

6780  `schema: |`

6781  `{`  
6782  `"id": "http://openinterconnect.org/schemas/oic.r.sensor.presence.json#",`  
6783  `"$schema": "http://json-schema.org/draft-04/schema#",`  
6784  `"title": "Presence Sensor",`  
6785  `"definitions": {`  
6786  `"oic.r.sensor.presence": {`  
6787  `"allOf": [`  
6788  `{"$ref": "oic.r.sensor.json#/definitions/oic.r.sensor"}`  
6789  `]`  
6790  `}`  
6791  `},`



```

6792         "type": "object",
6793         "allOf": [
6794             { "$ref": "oic.core.json#/definitions/oic.core" },
6795             { "$ref": "oic.baseResource.json#/definitions/oic.r.baseResource" },
6796             { "$ref": "#/definitions/oic.r.sensor.presence" }
6797         ],
6798         "required": ["value"]
6799     }
6800
6801     example: |
6802         {
6803             "rt": "oic.r.sensor.presence",
6804             "id": "unique_example_id",
6805             "value": true
6806         }
6807

```

### 6808 6.45.5 Property Definition

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

### 6809 6.45.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/PresenceResURI		get			

## 6810 6.46 Pan Tilt Zoom Movement

### 6811 6.46.1 Introduction

6812 This resource specifies the pan tilt and zoom capabilities of a device. The resource rt is dynamic  
6813 and reflects whether the values apply to physical movement of the device or digital/virtual  
6814 enhancements to the image. For physical movement the rt is 'oic.r.movement.ptz'. For  
6815 digital/virtual image enhancements the rt is 'oic.r.image.ptz'. The Pan and Tilt are specified in  
6816 degrees. The Zoom Factor is a value in the range 1-100 for linear (optical) zoom. The Zoom  
6817 Factor is a value in the range [1x, 2x, 4x, 8x, 16x, 32x] for digital zoom. If there is no zoom value  
6818 to set the Zoom Factor shall be '1x'. The value 0 degrees means neutral, this is the vendor  
6819 defined setting. Note that this resource also can be used to create an offset for physical  
6820 movement. When that is the case, the rt value is: oic.r.movement.offset.ptz Note that this  
6821 resource also can be used to create an offset for image movement. When that is the case, the rt  
6822 value is: oic.r.image.offset.ptz When the pan\_range value is omitted, then the range is [-180,180].  
6823 If pan is not supported then the range shall be [0,0] When the tilt\_range value is omitted, then  
6824 the range is [-180,180]. If tilt is not supported then the range shall be [0,0] Note: When the range  
6825 is specified as a float (e.g. 180.0, 180.0) then the pan and tilt values are also floats.

### 6826 6.46.2 Example URI

6827 /PanTiltZoomResURI

### 6828 6.46.3 Resource Type

6829 The resource type (rt) is defined as: oic.r.ptz.

### 6830 6.46.4 RAML Definition

```

6831 #%RAML 0.8
6832 title: OICPanTiltZoom
6833 version: v1.0-20150805
6834 traits:
6835   - interface :
6836       queryParameters:
6837         if:
6838           enum: ["oic.if.a"]

```

```

6839
6840 /PanTiltZoomResURI:
6841   description: |
6842     This resource specifies the pan tilt and zoom capabilities of a device.
6843     The resource rt is dynamic and reflects whether the values apply to
6844     physical movement of the device or digital/virtual enhancements to the image.
6845     For physical movement the rt is 'oic.r.movement.ptz'.
6846     For digital/virtual image enhancements the rt is 'oic.r.image.ptz'.
6847     The Pan and Tilt are specified in degrees.
6848     The Zoom Factor is a value in the range 1-100 for linear (optical) zoom.
6849     The Zoom Factor is a value in the range [1x, 2x, 4x, 8x, 16x, 32x] for digital zoom.
6850     If there is no zoom value to set the Zoom Factor shall be '1x'.
6851     The value 0 degrees means neutral, this is the vendor defined setting.
6852     Note that this resource also can be used to create an offset for physical movement.
6853     When that is the case, the rt value is: oic.r.movement.offset.ptz
6854     Note that this resource also can be used to create an offset for image movement.
6855     When that is the case, the rt value is: oic.r.image.offset.ptz
6856     When the pan_range value is omitted, then the range is [-180,180].
6857     If pan is not supported then the range shall be [0,0]
6858     When the tilt_range value is omitted, then the range is [-180,180].
6859     If tilt is not supported then the range shall be [0,0]
6860     Note: When the range is specified as a float (e.g
6861     180.0, 180.0) then the
6862     pan and tilt values are also floats.
6863
6864   is : ['interface']
6865   get:
6866     description: |
6867       Retrieves the current pan, tilt and zoom setting.
6868
6869   responses :
6870     200:
6871       body:
6872         application/json:
6873           schema: |
6874             {
6875               "id": "http://openinterconnect.org/schemas/oic.r.ptz#",
6876               "$schema": "http://json-schema.org/draft-04/schema#",
6877               "title": "Pan Tilt Zoom",
6878               "definitions": {
6879                 "oic.r.ptz": {
6880                   "type": "object",
6881                   "properties": {
6882                     "pan": {
6883                       "type": "number",
6884                       "description": "horizontal pan in degrees"
6885                     },
6886                     "tilt": {
6887                       "type": "number",
6888                       "description": "vertical tilt in degrees"
6889                     },
6890                     "pan_range": {
6891                       "type": "string",
6892                       "description": "ReadOnly, Min and Max values for the pan setting",
6893                       "format": "csv"
6894                     },
6895                     "tilt_range": {
6896                       "type": "string",
6897                       "description": "ReadOnly, Min and Max values for the tilt setting",
6898                       "format": "csv"
6899                     },
6900                     "zoomFactor": {
6901                       "type": "string",
6902                       "description": "The Zoomfactor value"
6903                     }
6904                   }
6905                 }
6906             }

```

```

6904         "zoomFactorRange": {
6905             "enum": ["linear", "1x", "2x", "4x", "8x", "16x", "32x"],
6906             "description": "ReadOnly, allowed Zoom Factor values. Linear equates to a
6907 1-100 min/max."
6908         }
6909     }
6910 },
6911 },
6912 "type": "object",
6913 "allOf": [
6914     {"$ref": "oic.core.json#/definitions/oic.core"},
6915     {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
6916     {"$ref": "#/definitions/oic.r.ptz"}
6917 ],
6918 "required": ["pan", "tilt", "zoomFactor"]
6919 }
6920
6921 example: |
6922 {
6923     "rt":         "oic.r.ptz",
6924     "id":         "unique_example_id",
6925     "pan":        0,
6926     "tilt":       0,
6927     "zoomFactor": "2x"
6928 }
6929
6930 post:
6931 description: |
6932     Sets the current pan, tilt and zoom value
6933
6934 body:
6935 application/json:
6936 schema: |
6937 {
6938     "id": "http://openinterconnect.org/schemas/oic.r.ptz#",
6939     "$schema": "http://json-schema.org/draft-04/schema#",
6940     "title": "Pan Tilt Zoom",
6941     "definitions": {
6942         "oic.r.ptz": {
6943             "type": "object",
6944             "properties": {
6945                 "pan": {
6946                     "type": "number",
6947                     "description": "horizontal pan in degrees"
6948                 },
6949                 "tilt": {
6950                     "type": "number",
6951                     "description": "vertical tilt in degrees"
6952                 },
6953                 "pan_range": {
6954                     "type": "string",
6955                     "description": "ReadOnly, Min and Max values for the pan setting",
6956                     "format": "csv"
6957                 },
6958                 "tilt_range": {
6959                     "type": "string",
6960                     "description": "ReadOnly, Min and Max values for the tilt setting",
6961                     "format": "csv"
6962                 },
6963                 "zoomFactor": {
6964                     "type": "string",
6965                     "description": "The Zoomfactor value"
6966                 },
6967                 "zoomFactorRange": {
6968                     "enum": ["linear", "1x", "2x", "4x", "8x", "16x", "32x"],
6969                     "description": "ReadOnly, allowed Zoom Factor values. Linear equates to a 1-100
6970 min/max."

```

```

6971     }
6972   }
6973 }
6974 },
6975 "type": "object",
6976 "allOf": [
6977   {"$ref": "oic.core.json#/definitions/oic.core"},
6978   {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
6979   {"$ref": "#/definitions/oic.r.ptz"}
6980 ],
6981 "required": ["pan", "tilt", "zoomFactor"]
6982 }
6983
6984 example: |
6985 {
6986   "id":          "unique_example_id",
6987   "pan":         10,
6988   "tilt":        -10,
6989   "zoomFactor": "4x"
6990 }
6991
6992 responses :
6993 200:
6994   body:
6995     application/json:
6996       schema: |
6997         {
6998           "id": "http://openinterconnect.org/schemas/oic.r.ptz#",
6999           "$schema": "http://json-schema.org/draft-04/schema#",
7000           "title": "Pan Tilt Zoom",
7001           "definitions": {
7002             "oic.r.ptz": {
7003               "type": "object",
7004               "properties": {
7005                 "pan": {
7006                   "type": "number",
7007                   "description": "horizontal pan in degrees"
7008                 },
7009                 "tilt": {
7010                   "type": "number",
7011                   "description": "vertical tilt in degrees"
7012                 },
7013                 "pan_range": {
7014                   "type": "string",
7015                   "description": "ReadOnly, Min and Max values for the pan setting",
7016                   "format": "csv"
7017                 },
7018                 "tilt_range": {
7019                   "type": "string",
7020                   "description": "ReadOnly, Min and Max values for the tilt setting",
7021                   "format": "csv"
7022                 },
7023                 "zoomFactor": {
7024                   "type": "string",
7025                   "description": "The Zoomfactor value"
7026                 },
7027                 "zoomFactorRange": {
7028                   "enum": ["linear", "1x", "2x", "4x", "8x", "16x", "32x"],
7029                   "description": "ReadOnly, allowed Zoom Factor values. Linear equates to a
7030 1-100 min/max."
7031                 }
7032               }
7033             }
7034           },
7035           "type": "object",
7036           "allOf": [
7037             {"$ref": "oic.core.json#/definitions/oic.core"},

```

```

7038         {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
7039         {"$ref": "#/definitions/oic.r.ptz"}
7040     ],
7041     "required": ["pan", "tilt", "zoomFactor"]
7042 }
7043
7044     example: |
7045         {
7046             "id":          "unique_example_id",
7047             "pan":         10,
7048             "tilt":        -10,
7049             "zoomFactor": "4x"
7050         }
7051

```

## 7052 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.

## 7053 6.46.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/PanTiltZoomResURI		get	post		

## 7054 6.47 Signal Strength

### 7055 6.47.1 Introduction

7056 This resource describes the strength of a signal by means of lqi and rssi. The lqi is a floating  
 7057 point number that represents Link Quality Indicator. The rssi is a floating point number that  
 7058 represents the received signal strength indicator.

### 7059 6.47.2 Example URI

7060 /SignalStrengthResURI

### 7061 6.47.3 Resource Type

7062 The resource type (rt) is defined as: oic.r.signalStrength.

### 7063 6.47.4 RAML Definition

```

7064 #%RAML 0.8
7065 title: OICSignalStrength
7066 version: v1.0-20150727
7067 traits:
7068   - interface :
7069       queryParameters:
7070         if:
7071           enum: ["oic.if.s"]
7072
7073 /SignalStrengthResURI:

```

```

7074 description: |
7075     This resource describes the strength of a signal by means of lqi and rssi.
7076     The lqi is a floating point number that represents Link Quality Indicator.
7077     The rssi is a floating point number that represents the received signal strength indicator.
7078
7079 is : ['interface']
7080
7080 get:
7081     responses :
7082         200:
7083             body:
7084                 application/json:
7085                     schema: |
7086                         {
7087                             "id": "http://openinterconnect.org/schemas/oic.r.signalStrength.json#",
7088                             "$schema": "http://json-schema.org/draft-04/schema#",
7089                             "title": "Signal Strength",
7090                             "definitions": {
7091                                 "oic.r.signalStrength": {
7092                                     "type": "object",
7093                                     "properties": {
7094                                         "lqi": {
7095                                             "type": "number",
7096                                             "description": "ReadOnly, current value of Link Quality Indicator"
7097                                         },
7098                                         "rssi": {
7099                                             "type": "number",
7100                                             "description": "ReadOnly, current value of Received Signal Strength
7101 Indicator"
7102                                         }
7103                                     }
7104                                 },
7105                             "type": "object",
7106                             "allOf": [
7107                                 {"$ref": "oic.core.json#/definitions/oic.core"},
7108                                 {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
7109                                 {"$ref": "#/definitions/oic.r.signalStrength"}
7110                             ],
7111                             "required": ["lqi","rssi"]
7112                         }
7113
7114
7115         example: |
7116             {
7117                 "rt": "oic.r.signalStrength",
7118                 "id": "unique_example_id",
7119                 "lqi": 10,
7120                 "rssi": 55
7121             }
7122

```

### 7123 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

### 7124 6.47.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/SignalStrengthResURI		get			

## 7125 **6.48 Speech Synthesis-TTS**

### 7126 **6.48.1 Introduction**

7127 This resource may be created on the OIC Server that is capable of rendering speech by an OIC  
7128 Client or may be created on the OIC Server by some resident application. The audio rendered is  
7129 at this stage local to the Server (i.e. not streamed). The utterance is an SSML document. The  
7130 supportedLanguages is a comma separate value list of the RFC 5646 defined language tags that  
7131 are supported. The supportedVoices is an SSML document fragment indicating the voices that  
7132 are supported.

### 7133 **6.48.2 Example URI**

7134 /SpeechTTSResURI

### 7135 **6.48.3 Resource Type**

7136 The resource type (rt) is defined as: oic.r.speech.tts.

### 7137 **6.48.4 RAML Definition**

7138 #%RAML 0.8

7139 title: *OICSpeechTTS*  
7140 version: *v1.0-20150727*

7141 traits:

7142 - interface :  
7143 queryParameters:  
7144 if:  
7145 enum: ["oic.if.a"]

7146

7147 /SpeechTTSResURI:

7148 description: |

7149 This resource may be created on the OIC Server that is capable of rendering speech by an OIC  
7150 Client  
7151 or may be created on the OIC Server by some resident application.  
7152 The audio rendered is at this stage local to the Server (i.e  
7153 not streamed).  
7154 The utterance is an SSML document.  
7155 The supportedLanguages is a comma separate value list of the RFC 5646 defined language tags  
7156 that are supported.  
7157 The supportedVoices is an SSML document fragment indicating the voices that are supported.  
7158

7159 is : ['interface']

7160 get:

7161 description: |

7162 Utterance in the example shall be a properly escaped (JSON rules) SSML document  
7163 An example is given below:  
7164 "<?xml version=\"1.0\" encoding=\"ISO-8859-1\"?>\n\r  
7165 < speak version=\"1.1\" xmlns=\"http://www.w3.org/2001/10/synthesis\" \n\r  
7166 \txmlns:xsi=\"http://www.w3.org/2001/XMLSchema-instance\" \n\r  
7167 \txsi:schemaLocation=\"http://www.w3.org/2001/10/synthesis \n\r  
7168 \thttp://www.w3.org/TR/speech-synthesis11/synthesis.xsd\" \n\r  
7169 \txml:lang=\"en-US\">\n\r  
7170 \n\r  
7171 \tThe title of the movie is:\n\r  
7172 \t\"La vita è bella\"\n\r  
7173 \t(Life is beautiful),\n\r  
7174 \twhich is directed by Roberto Benigni.\n\r  
7175 </speak\"  
7176

7177 responses :

7178 200:

7179 body:

7180 application/json:

```

7181     schema: |
7182         {
7183             "id": "http://openinterconnect.org/schemas/oic.r.speech.tts.json#",
7184             "$schema": "http://json-schema.org/draft-04/schema#",
7185             "title": "Speech Synthesis-TTS",
7186             "definitions": {
7187                 "oic.r.speech.tts": {
7188                     "type": "object",
7189                     "properties": {
7190                         "utterance": {
7191                             "type": "string",
7192                             "description": "SSML document including the speak body"
7193                         },
7194                         "supportedLanguages": {
7195                             "type": "string",
7196                             "description": "ReadOnly, comma separated list of supported language tags"
7197                         },
7198                         "supportedVoices": {
7199                             "type": "string",
7200                             "description": "ReadOnly, SSML document fragment indicating supported
7201 voices"
7202                         }
7203                     }
7204                 }
7205             },
7206             "type": "object",
7207             "allOf": [
7208                 {"$ref": "oic.core.json#/definitions/oic.core"},
7209                 {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
7210                 {"$ref": "#/definitions/oic.r.speech.tts"}
7211             ],
7212             "required": ["utterance"]
7213         }
7214
7215     example: |
7216         {
7217             "rt":             "oic.r.speech.tts",
7218             "id":             "unique_example_id",
7219             "utterance":      "SSML Document",
7220             "supportedLanguages": "en-US, en-GB, fr-CA",
7221             "supportedVoices": "<voice gender=\"female\" variant=\"2\"></voice>\n\r<voice
7222 name=\"Mike\"></voice>"
7223         }
7224
7225     post:
7226     description: |
7227         Changes the utterance being rendered.
7228         Example shows a change in language selected.
7229
7230     body:
7231     application/json:
7232     schema: |
7233         {
7234             "id": "http://openinterconnect.org/schemas/oic.r.speech.tts.json#",
7235             "$schema": "http://json-schema.org/draft-04/schema#",
7236             "title": "Speech Synthesis-TTS",
7237             "definitions": {
7238                 "oic.r.speech.tts": {
7239                     "type": "object",
7240                     "properties": {
7241                         "utterance": {
7242                             "type": "string",
7243                             "description": "SSML document including the speak body"
7244                         },
7245                         "supportedLanguages": {
7246                             "type": "string",
7247                             "description": "ReadOnly, comma separated list of supported language tags"

```



```

7248         },
7249         "supportedVoices": {
7250             "type": "string",
7251             "description": "ReadOnly, SSML document fragment indicating supported voices"
7252         }
7253     }
7254 }
7255 },
7256 "type": "object",
7257 "allOf": [
7258     {"$ref": "oic.core.json#/definitions/oic.core"},
7259     {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
7260     {"$ref": "#/definitions/oic.r.speech.tts"}
7261 ],
7262 "required": ["utterance"]
7263 }
7264
7265 example: |
7266 {
7267     "rt":          "oic.r.speech.tts",
7268     "id":          "unique_example_id",
7269     "utterance":  "SSML Document"
7270 }
7271
7272 responses :
7273 200:
7274     body:
7275     application/json:
7276         schema: |
7277             {
7278                 "id": "http://openinterconnect.org/schemas/oic.r.speech.tts.json#",
7279                 "$schema": "http://json-schema.org/draft-04/schema#",
7280                 "title": "Speech Synthesis-TTS",
7281                 "definitions": {
7282                     "oic.r.speech.tts": {
7283                         "type": "object",
7284                         "properties": {
7285                             "utterance": {
7286                                 "type": "string",
7287                                 "description": "SSML document including the speak body"
7288                             },
7289                             "supportedLanguages": {
7290                                 "type": "string",
7291                                 "description": "ReadOnly, comma separated list of supported language tags"
7292                             },
7293                             "supportedVoices": {
7294                                 "type": "string",
7295                                 "description": "ReadOnly, SSML document fragment indicating supported
7296 voices"
7297                             }
7298                         }
7299                     }
7300                 },
7301                 "type": "object",
7302                 "allOf": [
7303                     {"$ref": "oic.core.json#/definitions/oic.core"},
7304                     {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
7305                     {"$ref": "#/definitions/oic.r.speech.tts"}
7306                 ],
7307                 "required": ["utterance"]
7308             }
7309
7310         example: |
7311             {
7312                 "rt":          "oic.r.speech.tts",
7313                 "id":          "unique_example_id",

```

```

7314         "utterance": "SSML Document"
7315     }
7316 }

```

### 7317 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

### 7318 6.48.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/SpeechTTSResURI		get	post		

### 7319 6.49 Touch Sensor

#### 7320 6.49.1 Introduction

7321 This resource describes whether touch has been sensed or not. The value is a boolean. A value  
7322 of True means that touch has been sensed. A value of False means that touch not been sensed.

#### 7323 6.49.2 Example URI

7324 /TouchResURI

#### 7325 6.49.3 Resource Type

7326 The resource type (rt) is defined as: oic.r.sensor.touch.

#### 7327 6.49.4 RAML Definition

```

7328 #%RAML 0.8
7329 title: OICTouchSensor
7330 version: v1.0-20150727
7331 traits:
7332   - interface :
7333       queryParameters:
7334           if:
7335               enum: ["oic.if.s"]
7336
7337 /TouchResURI:
7338   description: |
7339     This resource describes whether touch has been sensed or not.
7340     The value is a boolean.
7341     A value of True means that touch has been sensed.
7342     A value of False means that touch not been sensed.
7343
7344   is : ['interface']
7345   get:
7346     responses :
7347       200:
7348         body:
7349           application/json:
7350             schema: |
7351               {
7352                 "id": "http://openinterconnect.org/schemas/oic.r.sensor.touch.json#",
7353                 "$schema": "http://json-schema.org/draft-04/schema#",
7354                 "title": "Touch Sensor",

```

```

7355     "definitions": {
7356       "oic.r.sensor.touch": {
7357         "allOf": [
7358           {"$ref": "oic.r.sensor.json#/definitions/oic.r.sensor"}
7359         ]
7360       }
7361     },
7362     "type": "object",
7363     "allOf": [
7364       {"$ref": "oic.core.json#/definitions/oic.core"},
7365       {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
7366       {"$ref": "#/definitions/oic.r.sensor.touch"}
7367     ],
7368     "required": ["value"]
7369   }
7370
7371   example: |
7372     {
7373       "rt": "oic.r.sensor.touch",
7374       "id": "unique_example_id",
7375       "value": true
7376     }
7377

```

### 7378 6.49.5 Property Definition

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

### 7379 6.49.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/TouchResURI		get			

## 7380 6.50 UV Radiation

### 7381 6.50.1 Introduction

7382 This resource specifies UV radiation measurement. The measurement is the current measured  
7383 UV Index

### 7384 6.50.2 Example URI

7385 /UVRadiationResURI

### 7386 6.50.3 Resource Type

7387 The resource type (rt) is defined as: oic.r.sensor.radiation.uv.

### 7388 6.50.4 RAML Definition

```

7389 #%RAML 0.8
7390 title: OICUVRadiation
7391 version: v1.0-20150805
7392 traits:
7393   - interface :
7394     queryParameters:
7395       if:
7396         enum: ["oic.if.s"]
7397
7398 /UVRadiationResURI:
7399   description: |
7400     This resource specifies UV radiation measurement.
7401     The measurement is the current measured UV Index
7402

```

```

7403     is : ['interface']
7404     get:
7405         description: |
7406             Retrieves the current UV Radiation value
7407
7408     responses :
7409         200:
7410             body:
7411                 application/json:
7412                     schema: |
7413                         {
7414                             "id": "http://openinterconnect.org/schemas/oic.r.sensor.radiation.uv#",
7415                             "$schema": "http://json-schema.org/draft-04/schema#",
7416                             "title": "UV Radiation",
7417                             "definitions": {
7418                                 "oic.r.sensor.radiation.uv": {
7419                                     "type": "object",
7420                                     "properties": {
7421                                         "measurement": {
7422                                             "type": "number",
7423                                             "description": "ReadOnly, the measured UV Index"
7424                                         }
7425                                     }
7426                                 }
7427                             },
7428                             "type": "object",
7429                             "allOf": [
7430                                 {"$ref": "oic.core.json#/definitions/oic.core"},
7431                                 {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
7432                                 {"$ref": "#/definitions/oic.r.sensor.radiation.uv"}
7433                             ],
7434                             "required": ["measurement"]
7435                         }
7436
7437                     example: |
7438                         {
7439                             "rt": "oic.r.sensor.radiation.uv",
7440                             "id": "unique_example_id",
7441                             "measurement": 3.5
7442                         }
7443

```

#### 7444 6.50.5 Property Definition

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

#### 7445 6.50.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/UVRadiationResURI		get			

### 7446 6.51 Water Sensor

#### 7447 6.51.1 Introduction

7448 This resource describes whether water has been sensed or not. The value is a boolean. A value  
7449 of True means that water has been sensed. A value of False means that water not been sensed.

#### 7450 6.51.2 Example URI

7451 /WaterResURI

#### 7452 6.51.3 Resource Type

7453 The resource type (rt) is defined as: oic.r.sensor.water.

7454 **6.51.4 RAML Definition**

```

7455 #RAML 0.8
7456 title: OICWaterSensor
7457 version: v1.0-20150727
7458 traits:
7459   - interface :
7460     queryParameters:
7461       if:
7462         enum: ["oic.if.a"]
7463
7464 /WaterResURI:
7465   description: |
7466     This resource describes whether water has been sensed or not.
7467     The value is a boolean.
7468     A value of True means that water has been sensed.
7469     A value of False means that water not been sensed.
7470
7471   is : ['interface']
7472   get:
7473     responses :
7474       200:
7475         body:
7476           application/json:
7477             schema: |
7478               {
7479                 "id": "http://openinterconnect.org/schemas/oic.r.sensor.water.json#",
7480                 "$schema": "http://json-schema.org/draft-04/schema#",
7481                 "title": "Water Sensor",
7482                 "definitions": {
7483                   "oic.r.sensor.water": {
7484                     "allOf": [
7485                       {"$ref": "oic.r.sensor.json#/definitions/oic.r.sensor"}
7486                     ]
7487                   }
7488                 },
7489                 "type": "object",
7490                 "allOf": [
7491                   {"$ref": "oic.core.json#/definitions/oic.core"},
7492                   {"$ref": "oic.baseResource.json#/definitions/oic.r.baseResource"},
7493                   {"$ref": "#/definitions/oic.r.sensor.water"}
7494                 ],
7495                 "required": ["value"]
7496               }
7497
7498             example: |
7499               {
7500                 "rt": "oic.r.sensor.water",
7501                 "id": "unique_example_id",
7502                 "value": true
7503               }
7504

```

7505 **6.51.5 Property Definition**

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

7506 **6.51.6 CRUDN behavior**

Resource	Create	Read	Update	Delete	Notify
----------	--------	------	--------	--------	--------

7507

/WaterResURI		get			
--------------	--	-----	--	--	--