

# OCF Core Specification

Part One

VERSION 1.0.0 | June 2017



**OPEN** CONNECTIVITY  
FOUNDATION™

CONTACT [admin@openconnectivity.org](mailto:admin@openconnectivity.org)

Copyright Open Connectivity Foundation, Inc. © 2017.  
All Rights Reserved.

## Legal Disclaimer

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

The OCF logo is a trademark of Open Connectivity Foundation, Inc. in the United States or other countries. \*Other names and brands may be claimed as the property of others.

Copyright © 2016-2017 Open Connectivity Foundation, Inc. All rights reserved.

Copying or other form of reproduction and/or distribution of these works are strictly prohibited.

## CONTENTS

20		
21		
22	1	Scope ..... 15
23	2	Normative references ..... 15
24	3	Terms, definitions, symbols and abbreviations ..... 18
25	3.1	Terms and definitions ..... 18
26	3.2	Symbols and abbreviations ..... 21
27	4	Document conventions and organization ..... 22
28	4.1	Conventions ..... 22
29	4.2	Notation ..... 22
30	4.3	Data types ..... 23
31	5	Architecture ..... 24
32	5.1	Overview ..... 24
33	5.2	Principle ..... 25
34	5.3	Functional block diagram ..... 26
35	5.4	Framework ..... 27
36	5.5	Example Scenario with roles ..... 27
37	5.6	Example Scenario: Bridging to Non- OCF ecosystem ..... 28
38	6	Identification and addressing ..... 29
39	6.1	Introduction ..... 29
40	6.2	Identification ..... 30
41	6.2.1	Resource identification and addressing ..... 30
42	6.3	Namespace ..... 31
43	6.4	Network addressing ..... 31
44	7	Resource model ..... 31
45	7.1	Introduction ..... 31
46	7.2	Resource ..... 32
47	7.3	Property ..... 33
48	7.3.1	Introduction ..... 33
49	7.3.2	Common Properties ..... 34
50	7.4	Resource Type ..... 35
51	7.4.1	Introduction ..... 35
52	7.4.2	Resource Type Property ..... 36
53	7.4.3	Resource Type definition ..... 36
54	7.4.4	Multi-value "rt" Resource ..... 38
55	7.5	Device Type ..... 38
56	7.6	Interface ..... 39
57	7.6.1	Introduction ..... 39
58	7.6.2	Interface Property ..... 39
59	7.6.3	Interface methods ..... 40
60	7.7	Resource representation ..... 52
61	7.8	Structure ..... 52
62	7.8.1	Introduction ..... 52

63	7.8.2	Resource Relationships .....	52
64	7.8.3	Collections .....	57
65	7.9	Third (3 <sup>rd</sup> ) party specified extensions .....	60
66	8	CRUDN .....	61
67	8.1	Overview .....	61
68	8.2	CREATE .....	62
69	8.2.1	CREATE request .....	63
70	8.2.2	Processing by the Server .....	63
71	8.2.3	CREATE response .....	63
72	8.3	RETRIEVE .....	63
73	8.3.1	RETRIEVE request .....	64
74	8.3.2	Processing by the Server .....	64
75	8.3.3	RETRIEVE response .....	64
76	8.4	UPDATE .....	64
77	8.4.1	UPDATE request .....	65
78	8.4.2	Processing by the Server .....	65
79	8.4.3	UPDATE response .....	65
80	8.5	DELETE .....	65
81	8.5.1	DELETE request .....	66
82	8.5.2	Processing by the Server .....	66
83	8.5.3	DELETE response .....	66
84	8.6	NOTIFY .....	66
85	9	Network and connectivity .....	67
86	9.1	Introduction .....	67
87	9.2	Architecture .....	67
88	9.3	IPv6 network layer requirements .....	68
89	9.3.1	Introduction .....	68
90	9.3.2	IPv6 node requirements .....	69
91	10	Endpoint .....	69
92	10.1	Endpoint definition .....	69
93	10.2	Endpoint information .....	70
94	10.2.1	Introduction .....	70
95	10.2.2	“ep” .....	70
96	10.2.3	“pri” .....	70
97	10.2.4	Endpoint information in “eps” Parameter .....	71
98	10.3	Endpoint discovery .....	71
99	10.3.1	Introduction .....	71
100	10.3.2	Implicit discovery .....	71
101	10.3.3	Explicit discovery with “/oic/res” response .....	71
102	10.4	CoAP based Endpoint discovery .....	75
103	11	Functional interactions .....	76
104	11.1	Introduction .....	76
105	11.2	Onboarding, Provisioning and Configuration .....	76
106	11.3	Resource discovery .....	78

107	11.3.1	Introduction .....	78
108	11.3.2	Resource based discovery: mechanisms .....	78
109	11.3.3	Resource based discovery: Information publication process .....	80
110	11.3.4	Resource based discovery: Finding information .....	81
111	11.3.5	Resource discovery using “/oic/res” .....	87
112	11.3.6	Resource directory (RD) based discovery .....	89
113	11.4	Notification.....	103
114	11.4.1	Overview .....	103
115	11.4.2	Observe .....	103
116	11.5	Device management.....	105
117	11.5.1	Overview .....	105
118	11.5.2	Diagnostics and maintenance .....	105
119	11.6	Scenes .....	106
120	11.6.1	Introduction .....	106
121	11.6.2	Scenes.....	106
122	11.6.3	Security considerations .....	110
123	11.7	Icons.....	110
124	11.7.1	Overview .....	110
125	11.7.2	Resource .....	111
126	11.8	Introspection.....	111
127	11.8.1	Overview .....	111
128	11.8.2	Usage of introspection.....	113
129	12	Messaging.....	114
130	12.1	Introduction .....	114
131	12.2	Mapping of CRUDN to CoAP.....	115
132	12.2.1	Overview .....	115
133	12.2.2	URIs.....	115
134	12.2.3	CoAP method with request and response .....	115
135	12.2.4	Content-Format negotiation.....	117
136	12.2.5	OCF-Content-Format-Version information.....	118
137	12.2.6	Content-Format policy .....	118
138	12.2.7	CRUDN to CoAP response codes .....	119
139	12.2.8	CoAP block transfer .....	119
140	12.3	CoAP serialization over TCP.....	120
141	12.4	Payload Encoding in CBOR.....	121
142	13	Security .....	121
143	Annex A (informative)	Operation Examples .....	123
144	A.1	Introduction.....	123
145	A.2	When at home: From smartphone turn on a single light.....	123
146	A.3	GroupAction execution.....	124
147	A.4	When garage door opens, turn on lights in hall; also notify smartphone.....	124
148	A.5	Device management.....	124
149	Annex B (informative)	OCF interaction scenarios and deployment models .....	126
150	B.1	OCF interaction scenarios.....	126

151	B.2	Deployment model.....	127
152	Annex C (informative)	Other Resource Models and OCF Mapping.....	129
153	C.1	Multiple resource models.....	129
154	C.2	OCF approach for support of multiple resource models.....	129
155	C.3	Resource model indication.....	130
156	C.4	An Example Profile (IPSO profile).....	130
157	C.4.1	Conceptual equivalence.....	130
158	Annex D (normative)	Resource Type definitions.....	133
159	D.1	List of Resource Type definitions.....	133
160	D.2	OCF Collection.....	134
161	D.2.1	Introduction.....	134
162	D.2.2	Example URI.....	134
163	D.2.3	Resource Type.....	134
164	D.2.4	RAML Definition.....	134
165	D.2.5	Property Definition.....	139
166	D.2.6	CRUDN behavior.....	140
167	D.2.7	Referenced JSON schemas.....	140
168	D.2.8	oic.oic-link-schema.json.....	140
169	D.3	Device Configuration.....	142
170	D.3.1	Introduction.....	142
171	D.3.2	Example URI.....	142
172	D.3.3	Resource Type.....	142
173	D.3.4	RAML Definition.....	142
174	D.3.5	Property Definition.....	147
175	D.3.6	CRUDN behavior.....	147
176	D.4	Platform Configuration.....	147
177	D.4.1	Introduction.....	147
178	D.4.2	Example URI.....	147
179	D.4.3	Resource Type.....	147
180	D.4.4	RAML Definition.....	147
181	D.4.5	Property Definition.....	150
182	D.4.6	CRUDN behavior.....	150
183	D.5	Device.....	150
184	D.5.1	Introduction.....	150
185	D.5.2	Wellknown URI.....	150
186	D.5.3	Resource Type.....	150
187	D.5.4	RAML Definition.....	151
188	D.5.5	Property Definition.....	153
189	D.5.6	CRUDN behavior.....	153
190	D.6	Maintenance.....	154
191	D.6.1	Introduction.....	154
192	D.6.2	Wellknown URI.....	154
193	D.6.3	Resource Type.....	154
194	D.6.4	RAML Definition.....	154

195	D.6.5	Property Definition .....	156
196	D.6.6	CRUDN behavior.....	156
197	D.7	Platform .....	157
198	D.7.1	Introduction .....	157
199	D.7.2	Wellknown URI .....	157
200	D.7.3	Resource Type .....	157
201	D.7.4	RAML Definition.....	157
202	D.7.5	Property Definition .....	159
203	D.7.6	CRUDN behavior.....	160
204	D.8	Ping.....	160
205	D.8.1	Introduction .....	160
206	D.8.2	Wellknown URI .....	160
207	D.8.3	Resource Type .....	160
208	D.8.4	RAML Definition.....	160
209	D.8.5	Property Definition .....	162
210	D.8.6	CRUDN behavior.....	162
211	D.9	Discoverable Resources Baseline Interface .....	162
212	D.9.1	Introduction .....	162
213	D.9.2	Wellknown URI .....	162
214	D.9.3	Resource Type .....	162
215	D.9.4	RAML Definition.....	162
216	D.9.5	Property Definition .....	164
217	D.9.6	CRUDN behavior.....	165
218	D.10	Discoverable Resources Link List interface.....	165
219	D.10.1	Introduction .....	165
220	D.10.2	Wellknown URI .....	165
221	D.10.3	Resource Type .....	165
222	D.10.4	RAML Definition.....	165
223	D.10.5	Property Definition .....	166
224	D.10.6	CRUDN behavior.....	167
225	D.10.7	Referenced JSON schemas .....	168
226	D.10.8	oic.oic-link-schema.json.....	168
227	D.11	Scenes (Top level).....	170
228	D.11.1	Introduction .....	170
229	D.11.2	Example URI.....	170
230	D.11.3	Resource Type .....	170
231	D.11.4	RAML Definition.....	170
232	D.11.5	Property Definition .....	172
233	D.11.6	CRUDN behavior.....	172
234	D.12	Scene Collections .....	172
235	D.12.1	Introduction .....	172
236	D.12.2	Example URI.....	173
237	D.12.3	Resource Type .....	173
238	D.12.4	RAML Definition.....	173

239	D.12.5	Property Definition .....	176
240	D.12.6	CRUDN behavior.....	177
241	D.13	Scene Member.....	177
242	D.13.1	Introduction .....	177
243	D.13.2	Example URI.....	177
244	D.13.3	Resource Type .....	177
245	D.13.4	RAML Definition.....	177
246	D.13.5	Property Definition .....	179
247	D.13.6	CRUDN behavior.....	179
248	D.14	Resource directory resource.....	179
249	D.14.1	Introduction .....	179
250	D.14.2	Wellknown URI.....	180
251	D.14.3	Resource Type .....	180
252	D.14.4	RAML Definition.....	180
253	D.14.5	Property Definition .....	185
254	D.14.6	CRUDN behavior.....	186
255	D.15	Icon.....	186
256	D.15.1	Introduction .....	186
257	D.15.2	Example URI.....	186
258	D.15.3	Resource Type .....	186
259	D.15.4	RAML Definition.....	186
260	D.15.5	Property Definition .....	187
261	D.15.6	CRUDN behavior.....	187
262	D.16	Introspection Resource .....	188
263	D.16.1	Introduction .....	188
264	D.16.2	Example URI.....	188
265	D.16.3	Resource Type .....	188
266	D.16.4	RAML Definition.....	188
267	D.16.5	Property Definition .....	189
268	D.16.6	CRUDN behavior.....	190
269	Annex E (informative)	Swagger2.0 definitions.....	191
270	E.1	Icon.....	191
271	E.1.1	Introduction .....	191
272	E.1.2	Example URI.....	191
273	E.1.3	Resource Type .....	191
274	E.1.4	Swagger2.0 Definition.....	191
275	E.1.5	Property Definition .....	193
276	E.1.6	CRUDN behavior.....	193
277	E.2	Introspection Resource .....	194
278	E.2.1	Introduction .....	194
279	E.2.2	Example URI.....	194
280	E.2.3	Resource Type .....	194
281	E.2.4	Swagger2.0 Definition.....	194
282	E.2.5	Property Definition .....	196



283	E.2.6	CRUDN behavior.....	197
284	E.3	OCF Collection.....	197
285	E.3.1	Introduction .....	197
286	E.3.2	Example URI.....	197
287	E.3.3	Resource Type .....	197
288	E.3.4	Swagger2.0 Definition.....	197
289	E.3.5	Property Definition .....	210
290	E.3.6	CRUDN behavior.....	213
291	E.4	Platform Configuration .....	213
292	E.4.1	Introduction .....	213
293	E.4.2	Example URI.....	213
294	E.4.3	Resource Type .....	213
295	E.4.4	Swagger2.0 Definition.....	213
296	E.4.5	Property Definition .....	217
297	E.4.6	CRUDN behavior.....	217
298	E.5	Device Configuration.....	217
299	E.5.1	Introduction .....	217
300	E.5.2	Example URI.....	217
301	E.5.3	Resource Type .....	217
302	E.5.4	Swagger2.0 Definition.....	217
303	E.5.5	Property Definition .....	223
304	E.5.6	CRUDN behavior.....	224
305	E.6	Device.....	224
306	E.6.1	Introduction .....	224
307	E.6.2	Wellknown URI.....	224
308	E.6.3	Resource Type .....	224
309	E.6.4	Swagger2.0 Definition.....	224
310	E.6.5	Property Definition .....	227
311	E.6.6	CRUDN behavior.....	228
312	E.7	Maintenance.....	228
313	E.7.1	Introduction .....	228
314	E.7.2	Wellknown URI.....	228
315	E.7.3	Resource Type .....	229
316	E.7.4	Swagger2.0 Definition.....	229
317	E.7.5	Property Definition .....	231
318	E.7.6	CRUDN behavior.....	231
319	E.8	Platform.....	231
320	E.8.1	Introduction .....	231
321	E.8.2	Wellknown URI.....	231
322	E.8.3	Resource Type .....	231
323	E.8.4	Swagger2.0 Definition.....	232
324	E.8.5	Property Definition .....	235
325	E.8.6	CRUDN behavior.....	236
326	E.9	Ping.....	236

327	E.9.1	Introduction .....	236
328	E.9.2	Wellknown URI .....	236
329	E.9.3	Resource Type .....	236
330	E.9.4	Swagger2.0 Definition .....	236
331	E.9.5	Property Definition .....	238
332	E.9.6	CRUDN behavior.....	239
333	E.10	Resource directory resource.....	239
334	E.10.1	Introduction .....	239
335	E.10.2	Wellknown URI .....	239
336	E.10.3	Resource Type .....	239
337	E.10.4	Swagger2.0 Definition .....	239
338	E.10.5	Property Definition .....	248
339	E.10.6	CRUDN behavior.....	249
340	E.11	Discoverable Resources.....	249
341	E.11.1	Introduction .....	249
342	E.11.2	Wellknown URI .....	249
343	E.11.3	Resource Type .....	249
344	E.11.4	Swagger2.0 Definition .....	249
345	E.11.5	Property Definition .....	256
346	E.11.6	CRUDN behavior.....	258
347	E.12	Scenes .....	258
348	E.12.1	Introduction .....	258
349	E.12.2	Example URI.....	258
350	E.12.3	Resource Type .....	258
351	E.12.4	Swagger2.0 Definition .....	258
352	E.12.5	Property Definition .....	272
353	E.12.6	CRUDN behavior.....	275
354			
355			

356		
357		Figures
358		
359	Figure 1: Architecture - concepts .....	25
360	Figure 2: Functional block diagram.....	26
361	Figure 3: Communication layering model.....	27
362	Figure 4: Example illustrating the Roles .....	28
363	Figure 5: Framework - Architecture Detail.....	28
364	Figure 6: Server bridging to Non- OCF device.....	29
365	Figure 7: Example of a Resource.....	33
366	Figure 8: Example - "Heater" Resource (for illustration only).....	50
367	Figure 9: Example - Actuator Interface.....	50
368	Figure 10: Example of a Link .....	52
369	Figure 11: Example of distinct Links.....	52
370	Figure 12: Example of use of anchor in Link .....	53
371	Figure 13: Example of "eps Parameter.....	56
372	Figure 14: List of Links in a Resource.....	57
373	Figure 15: Example showing Collection and Links .....	59
374	Figure 16. CREATE operation.....	63
375	Figure 17. RETRIEVE operation .....	64
376	Figure 18. UPDATE operation.....	65
377	Figure 19. DELETE operation .....	66
378	Figure 20. High Level Network & Connectivity Architecture.....	68
379	Figure 21: Example of "ep" .....	70
380	Figure 22: Example of Link with "eps" Parameter .....	71
381	Figure 23: Example of "/oic/res" with Endpoint information .....	75
382	Figure 24. Resource based discovery: Information publication process .....	81
383	Figure 25. Resource based discovery: Finding information .....	81
384	Figure 26. Indirect discovery of resource by resource directory.....	90
385	Figure 27. RD discovery and RD supported query of resources support.....	92
386	Figure 28. Resource Direction Deployment Scenarios.....	93
387	Figure 29. Example of POST request payload.....	97
388	Figure 30. Example of POST response payload.....	98
389	Figure 31. Example of DELETE request with "di" or "ins" query.....	99
390	Figure 32. Observe Mechanism.....	104
391	Figure 33 Generic scene resource structure .....	107
392	Figure 34 Interactions to check Scene support and setup of specific scenes .....	108
393	Figure 35 Client interactions on a specific scene.....	109
394	Figure 36 Interaction overview due to a Scene change.....	110

395	Figure 37 Interactions to check Introspection support and download the Introspection	
396	Device Data.....	114
397	Figure 38 Content-Format Policy .....	119
398	Figure 39. When at home: from smartphone turn on a single light.....	124
399	Figure 40. Device management (maintenance).....	125
400	Figure 41. Direct interaction between Server and Client .....	126
401	Figure 42. Interaction between Client and Server using another Server.....	126
402	Figure 43. Interaction between Client and Server using Intermediary.....	126
403	Figure 44. Interaction between Client and Server using support from multiple Servers and	
404	Intermediary.....	127
405	Figure 45. Example of Devices.....	127
406		
407		
408		

## Tables

409		
410		
411	Table 1. Additional OCF Types.....	<b>Error! Bookmark not defined.</b>
412	Table 2. Name Property Definition.....	35
413	Table 3. Resource Identity Property Definition.....	35
414	Table 4. Resource Type Common Property definition .....	36
415	Table 5. Example foobar Resource Type.....	37
416	Table 6. Example foobar properties .....	37
417	Table 7. Resource Interface Property definition .....	39
418	Table 8. OCF standard Interfaces .....	40
419	Table 9. Common Properties for Collections (in addition to Common Properties defined in	
420	section 7.3.2) .....	60
421	Table 10. 3rd party defined Resource elements .....	61
422	Table 11. Parameters of CRUDN messages .....	62
423	Table 12. “ep” value for Transport Protocol Suite .....	70
424	Table 13. List of Core Resources.....	76
425	Table 14. Configuration Resource.....	76
426	Table 15. “oic.wk.con” Resource Type definition.....	77
427	Table 16. “oic.wk.con.p” Resource Type definition.....	78
428	Table 17. Mandatory discovery Core Resources .....	82
429	Table 18. “oic.wk.res” Resource Type definition.....	83
430	Table 19. Protocol scheme registry.....	83
431	Table 20. “oic.wk.d” Resource Type definition.....	84
432	Table 21. “oic.wk.p” Resource Type definition.....	86
433	Table 22. “oic.wk.rd” Resource Type definition.....	90
434	Table 23. “oic.wk.rd” Properties.....	91
435	Table 24: Selection parameters.....	94
436	Table 25. Optional diagnostics and maintenance device management Core Resources ....	105
437	Table 26. “oic.wk.mnt” Resource Type definition.....	106
438	Table 27 list of Resource Types for Scenes.....	110
439	Table 28. Optional Icon Core Resource.....	111
440	Table 29. “oic.r.icon” Resource Type definition.....	111
441	Table 30. Introspection Resource.....	113
442	Table 31. “oic.wk.introspection” Resource Type definition.....	113
443	Table 32. CoAP request and response.....	115
444	Table 33. OCF Content-Formats.....	117
445	Table 34. OCF-Content-Format-Version and OCF-Accept-Content-Format-Version Option	
446	Numbers.....	118
447	Table 35. OCF-Accept-Content-Format-Version and the OCF-Content-Format-Version	
448	Representation .....	118

449	Table 36. Examples of OCF-Content-Format-Version and OCF-Accept-Content-Format-	
450	Version Representation .....	118
451	Table 37. Ping resource.....	120
452	Table 38. “oic.wk.ping” Resource Type definition .....	121
453	Table 39. oic.example.light Resource Type definition .....	123
454	Table 40. oic.example.garagedoor Resource Type definition.....	123
455	Table 41. Light control Resource Type definition.....	131
456	Table 42. Light control Resource Type definition.....	131
457	Table 43. Alphabetized list of core resources.....	133
458		
459		

## 1 Scope

The OCF specifications are divided into two sets of documents:

- Core Specification documents: The Core Specification documents specify the Framework, i.e., the OCF core architecture, interfaces, protocols and services to enable OCF profiles implementation for Internet of Things (IoT) usages and ecosystems.
- Vertical Profiles Specification documents: The Vertical Profiles Specification documents specify the OCF profiles to enable IoT usages for different market segments such as smart home, industrial, healthcare, and automotive. The Application Profiles Specification is built upon the interfaces and network security of the OCF core architecture defined in the Core Specification.

This document is the OCF Core specification which specifies the Framework and core architecture.

## 2 Normative references

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

ISO 8601, *Data elements and interchange formats – Information interchange – Representation of dates and times*, International Standards Organization, December 3, 2004

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

IETF RFC 1981, *Path MTU Discovery for IP version 6*, August 1996  
<https://tools.ietf.org/rfc/rfc1981.txt>

IETF RFC 2460, *Internet Protocol, version 6 (IPv6), December, 1998*  
<https://tools.ietf.org/rfc/rfc2460.txt>

IETF RFC 2616, *Hypertext Transfer Protocol – HTTP/1.1*, June 1999.  
<http://www.ietf.org/rfc/rfc2616.txt>

IETF RFC 3810, *Multicast Listener Discovery Version 2 (MLDv2) for IPv6*, June 2004  
<http://www.ietf.org/rfc/rfc3810.txt>

IETF RFC 3986, *Uniform Resource Identifier (URI): General Syntax*, January 2005.  
<http://www.ietf.org/rfc/rfc3986.txt>

IETF RFC 4122, *A Universally Unique Identifier (UUID) URN Namespace*, July 2005  
<http://www.ietf.org/rfc/rfc4122.txt>

IETF RFC 4287, *The Atom Syndication Format*, December 2005,  
<http://www.ietf.org/rfc/rfc4287.txt>

IETF RFC 4193, *Unique Local IPv6 Unicast Addresses*, October 2005  
<http://www.ietf.org/rfc/rfc4193.txt>

IETF RFC 4291, *IP Version 6 Addressing Architecture*, February 2006  
<http://www.ietf.org/rfc/rfc4291.txt>

IETF RFC 4443, *Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification*, March 2006  
<http://www.ietf.org/rfc/rfc4443.txt>

500 IETF RFC 4861, *Neighbor Discovery for IP version 6 (IPv6)*, September 2007  
501 <http://www.ietf.org/rfc/rfc4861.txt>

502 IETF RFC 4862, *IPv6 Stateless Address Autoconfiguration*, September 2007  
503 <http://www.ietf.org/rfc/rfc4862.txt>

504 IETF RFC 4941, *Privacy Extensions for Stateless Address Autoconfiguration in IPv6*, September  
505 2007  
506 <http://www.ietf.org/rfc/rfc4941.txt>

507 IETF RFC 4944, *Transmission of IPv6 Packets over IEEE 802.15.4 Networks*, September 2007  
508 <http://www.ietf.org/rfc/rfc4944.txt>

509 IETF RFC 5646, *Tags for Identifying Languages*, September 2009  
510 <http://www.ietf.org/rfc/rfc5646.txt>

511 IETF RFC 5988, *Web Linking: General Syntax*, October 2010  
512 <http://www.ietf.org/rfc/rfc5988.txt>

513 IETF RFC 6434, *IPv6 Node Requirements*, December 2011  
514 <http://www.ietf.org/rfc/rfc6434.txt>

515 IETF RFC 6455, *The WebSocket Protocol*, December 2011  
516 <https://www.ietf.org/rfc/rfc6455.txt>

517 IETF RFC 6573, *The Item and Collection Link Relations*, April 2012  
518 <http://www.ietf.org/rfc/rfc6573.txt>

519 IETF RFC 6690, *Constrained RESTful Environments (CoRE) Link Format*, August 2012  
520 <http://www.ietf.org/rfc/rfc6690.txt>

521 IETF RFC 6762, *Multicast DNS* February 2013  
522 <http://www.ietf.org/rfc/rfc6762.txt>

523 IETF RFC 6763, *DNS-Based Service Discovery*, February 2013  
524 <http://www.ietf.org/rfc/rfc6763.txt>

525 IETF RFC 6775, *Neighbor Discovery Optimization for IPv6 over Low-Power Wireless Personal  
526 Area Networks (6LoWPANs)*, November 2012  
527 <http://www.ietf.org/rfc/rfc6775.txt>

528 IETF RFC 7049, *Concise Binary Object Representation (CBOR)*, October 2013  
529 <http://www.ietf.org/rfc/rfc7049.txt>

530 IETF RFC 7084, *Basic Requirements for IPv6 Customer Edge Routers*, November 2013  
531 <http://www.ietf.org/rfc/rfc7084.txt>

532 IETF RFC 7159, *The JavaScript Object Notation (JSON) Data Interchange Format*, March 2014  
533 <http://tools.ietf.org/rfc/rfc7159.txt>

534 IETF RFC 7252, *The Constrained Application Protocol (CoAP)*, June 2014  
535 <http://tools.ietf.org/rfc/rfc7252.txt>

536 IETF RFC 7301, *Transport Layer Security (TLS) Application-Layer Protocol Negotiation  
537 Extension*, July 2014  
538 <https://tools.ietf.org/html/rfc7301>



539 IETF RFC 7428, *Transmission of IPv6 Packets over ITU-T G.9959 Networks*, February 2015  
540 <http://www.ietf.org/rfc/rfc7428.txt>

541 IETF RFC 7641, *Observing Resources in the Constrained Application Protocol*  
542 *(CoAP)*, September 2015  
543 <https://tools.ietf.org/html/rfc7641>

544 IETF RFC 7668, *IPv6 over BLUETOOTH(r) Low Energy*, October 2015  
545 <https://tools.ietf.org/html/rfc7668>

546 IETF RFC 7721, *Security and Privacy Considerations for IPv6 Address Generation Mechanisms*,  
547 March 2016  
548 <https://tools.ietf.org/html/rfc7721>

549 IETF RFC 7959, *Block-Wise Transfers in the Constrained Application Protocol (CoAP)*, August  
550 2016  
551 <https://tools.ietf.org/html/rfc7959>

552 IETF draft-ietf-core-coap-tcp-tls-07, *CoAP over TCP, TLS, and WebSockets*, June 10 2015  
553 <https://datatracker.ietf.org/doc/draft-ietf-core-coap-tcp-tls/>

554 ECMA-4-4, *The JSON Data Interchange Format*, October 2013.  
555 <http://www.ecma-international.org/publications/files/ECMA-ST/ECMA-404.pdf>

556 OCF Security, *Open Connectivity Foundation Security Capabilities*, Version 1.0,

557 IANA IPv6 Multicast Address Space Registry  
558 <http://www.iana.org/assignments/ipv6-multicast-addresses/ipv6-multicast-addresses.xhtml>

559 IANA Media Types Assignment, March 2017  
560 <http://www.iana.org/assignments/media-types/media-types.xhtml>

561

562

563 OpenAPI specification, *fka Swagger RESTful API Documentation Specification*  
564 <https://github.com/OAI/OpenAPI-Specification/blob/master/versions/2.0.md>

565 OCF Resource Type Definitions, *API Definition Language for OCF Resource Type Definitions*,  
566 Release OCF-v1.0.0  
567 <https://github.com/openconnectivityfoundation/core>

568 W3C XML character escaping, *Extensible Markup Language (XML) 1.0*, November 2008  
569 <http://www.w3.org/TR/2008/REC-xml-20081126/#syntax>

570 **3 Terms, definitions, symbols and abbreviations**

571 **3.1 Terms and definitions**

572 **3.1.1**  
573 **Client**  
574 a logical entity that accesses a Resource on a Server

575 **3.1.2**  
576 **Collection**  
577 a Resource that contains zero or more Links

578 **3.1.3**  
579 **Configuration Source**  
580 a cloud or service network or a local read-only file which contains and provides  
581 configuration related information to the Devices

582 **3.1.4**  
583 **Core Resources**  
584 those Resources that are defined in this specification

585 **3.1.5**  
586 **Default Interface**  
587 an Interface used to generate the response when an Interface is omitted in a request

588 **3.1.6**  
589 **Device**  
590 a logical entity that assumes one or more Roles (e.g., Client, Server)

591 Note 1 to entry: More than one Device can exist on a physical platform.

592 **3.1.7**  
593 **Device Type**  
594 a uniquely named definition indicating a minimum set of Resource Types that a Device supports

595 Note 1 to entry: A Device Type provides a hint about what the Device is, such as a light or a fan, for use during  
596 Resource discovery.

597 **3.1.8**  
598 **Endpoint**  
599 the source or destination of a request and response messages for a given Transport Protocol Suite

600 Note 1 to entry: Example of a Transport Protocol Suite would be CoAP over UDP over IPv6.

601 **3.1.9**  
602 **Entity**  
603 an aspect of the physical world that is exposed through a Device

604 Note 1 to entry: Example of an entity is an LED.

605 **3.1.10**  
606 **Framework**  
607 a set of related functionalities and interactions defined in this specification, which enable  
608 interoperability across a wide range of networked devices, including IoT

609 **3.1.11**  
610 **Interface**  
611 provides a view and permissible responses on a Resource

612 **3.1.12**  
613 **Introspection**  
614 mechanism to determine the capabilities of the hosted Resources of a Device

615 **3.1.13**  
616 **Introspection Device Data**  
617 data that describes the payloads per implemented method of the Resources that makes up the  
618 Device

619 Note 1 to entry: See section 11.8 for all requirements and exceptions

620 **3.1.14**  
621 **Links**  
622 extends typed web links according to IETF RFC 5988

623 **3.1.15**  
624 **Non-OCF Device**  
625 A device which does not comply with the OCF Device requirements

626 **3.1.16**  
627 **Notification**  
628 the mechanism to make a Client aware of resource state changes in a Resource

629 **3.1.17**  
630 **Observe**  
631 the act of monitoring a Resource by sending a RETRIEVE request which is cached by the Server  
632 hosting the Resource and reprocessed on every change to that Resource

633 **3.1.18**  
634 **Parameter**  
635 an element that provides metadata about a Resource referenced by the target URI of a Link

636 **3.1.19**  
637 **Partial UPDATE**  
638 an UPDATE request to a Resource that includes a subset of the Properties that are visible via the  
639 Interface being applied for the Resource Type

640 **3.1.20**  
641 **Platform**  
642 a physical device containing one or more Devices

643 **3.1.21**  
644 **Remote Access Endpoint (RAE) Client**  
645 a Client which supports XMPP functionality in order to access a Server from a remote location

646 **3.1.22**  
647 **Remote Access Endpoint (RAE) Server**  
648 a Server which supports XMPP and can publish its resource(s) to an XMPP server in the cloud,  
649 thus becoming remotely addressable and accessible

650 Note 1 to entry: An RAE Server also supports ICE/STUN/TURN.

651 **3.1.23**  
652 **Resource**  
653 represents an Entity modelled and exposed by the Framework

654 **3.1.24**  
655 **Resource Directory**  
656 a set of descriptions of Resources where the actual Resources are held on Servers external to the  
657 Device hosting the Resource Directory, allowing lookups to be performed for those resources

658 Note 1 to entry: This functionality can be used by sleeping Servers or Servers that choose not to listen/respond to  
659 multicast requests directly.

660 **3.1.25**  
661 **Resource Interface**  
662 a qualification of the permitted requests on a Resource

663 **3.1.26**  
664 **Resource Property**  
665 a significant aspect or parameter of a resource, including metadata, that is exposed through the  
666 Resource

667 **3.1.27**  
668 **Resource Type**  
669 a uniquely named definition of a class of Resource Properties and the interactions that are  
670 supported by that class

671 Note 1 to entry: Each Resource has a Property "rt" whose value is the unique name of the Resource Type.

672 **3.1.28**  
673 **Scene**  
674 a static entity that stores a set of defined Resource property values for a collection of Resources

675 Note 1 to entry: A Scene is a prescribed setting of a set of resources with each having a predetermined value for the  
676 property that has to change.

677 **3.1.29**  
678 **Scene Collection**  
679 a collection Resource that contains an enumeration of possible Scene Values and the current  
680 Scene Value

681 Note 1 to entry: The member values of the Scene collection Resource are Scene Members.

682 **3.1.30**  
683 **Scene Member**  
684 a Resource that contains mappings of Scene Values to values of a property in the resource

685 **3.1.31**  
686 **Scene Value**  
687 a Scene enumerator representing the state in which a Resource can be

688 **3.1.32**  
689 **Server**  
690 a Device with the role of providing resource state information and facilitating remote interaction  
691 with its resources

692 Note 1 to entry: A Server can be implemented to expose non-OCF Device resources to Clients (section 5.6)

693 **3.1.33**  
694 **Vertical Resource Type**  
695 a Resource Type in a vertical domain specification

696 Note 1 to entry: An example of a Vertical Resource Type would be “oic.r.switch.binary”.

## 697 **3.2 Symbols and abbreviations**

### 698 **3.2.1**

#### 699 **ACL**

700 Access Control List

701 Note 1 to entry: The details are defined in OCF Security.

### 702 **3.2.2**

#### 703 **CBOR**

704 Concise Binary Object Representation

### 705 **3.2.3**

#### 706 **CoAP**

707 Constrained Application Protocol

### 708 **3.2.4**

#### 709 **EXI**

710 Efficient XML Interchange

### 711 **3.2.5**

#### 712 **IRI**

713 Internationalized Resource Identifiers

### 714 **3.2.6**

#### 715 **ISP**

716 Internet Service Provider

### 717 **3.2.7**

#### 718 **JSON**

719 JavaScript Object Notation

### 720 **3.2.8**

#### 721 **mDNS**

722 Multicast Domain Name Service

### 723 **3.2.9**

#### 724 **MTU**

725 Maximum Transmission Unit

### 726 **3.2.10**

#### 727 **NAT**

728 Network Address Translation

### 729 **3.2.11**

#### 730 **OCF**

731 Open Connectivity Foundation

732 the organization that created this specification

### 733 **3.2.12**

#### 734 **RAML**

735 RESTful API Modeling Language

### 736 **3.2.13**

#### 737 **REST**

738 Representational State Transfer

739 **3.2.14**  
740 **RESTfull**  
741 REST-compliant Web services

742 **3.2.15**  
743 **URI**  
744 Uniform Resource Identifier

745 **3.2.16**  
746 **URN**  
747 Uniform Resource Name

748 **3.2.17**  
749 **UTC**  
750 Coordinated Universal Time

751 **3.2.18**  
752 **UUID**  
753 Universal Unique Identifier

754 **3.2.19**  
755 **XML**  
756 Extensible Markup Language

## 757 **4 Document conventions and organization**

### 758 **4.1 Conventions**

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

### 763 **4.2 Notation**

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

766 Required (or shall or mandatory)(M).

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

770 Recommended (or should)(S).

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

777 Allowed (may or allowed)(O).

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

781 DEPRECATED.

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

Conditionally allowed (CA)

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

Conditionally required (CR)

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

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

Words that are emphasized are printed in *italic*.

### 4.3 Data types

Resources are defined using data types derived from JSON values as defined in ECMA-4-4. However, a Resource can overload a JSON defined value to specify a particular subset of the JSON value, using validation keywords defined in [JSON Schema Validation].

Among other validation keywords, section 7 of [JSON Schema Validation] defines a “format” keyword with a number of format attributes such as “uri” and “date-time”, and a “pattern” keyword with a regular expression that can be used to validate a string. This section defines patterns that are available for use in describing OCF Resources. The pattern names can be used in specification text where JSON format names can occur. The actual JSON schemas shall use the JSON type and pattern instead.

For all rows defined in Table 1 below, the JSON type is string.

**Table 1. Additional OCF Types**

Pattern Name	Pattern	Description
csv	<none>	A comma separated list of values encoded within a string. The value type in the csv is described by the property where the csv is used. For example a csv of integers.  Note: csv is considered deprecated and an array of strings should be used instead for new Resources.
date	^([0-9]{4})-(1[0-2] 0[1-9])-(3[0-1] 2[0-9] 1[0-9] 0[1-9])\$	As defined in ISO 8601. The format is [yyyy]-[mm]-[dd].
int64	^0 (-?[1-9][0-9]{0,18})\$	A string instance is valid against this attribute if it contains an integer in the range $[-(2^{63}), (2^{63}-1)]$

		Note: IETF RFC 7159 section 6 explains that JSON integers outside the range $[-(2^{53})+1, (2^{53})-1]$ are not interoperable and so JSON numbers cannot be used for 64-bit numbers.
language-tag	$^{[A-Za-z]\{1,8\}(-[A-Za-z0-9]\{1,8\})^*\$}$	An IETF language tag formatted according to IETF RFC 5646 section 2.1.
uint64	$^{0 ([1-9][0-9]\{0,19\})\$}$	A string instance is valid against this attribute if it contains an integer in the range $[0, (2^{64})-1]$  Also see note for int64
uuid	$^{[a-fA-F0-9]\{8\}-[a-fA-F0-9]\{4\}-[a-fA-F0-9]\{4\}-[a-fA-F0-9]\{4\}-[a-fA-F0-9]\{12\}\$}$	A UUID string representation formatted according to IETF RFC 4122 section 3.

Strings shall be encoded as UTF-8 unless otherwise specified.

In a JSON schema, “maxLength” for a string indicates the maximum number of characters not octets. However, “maxLength” shall also indicate the maximum number of octets. If no “maxLength” is defined for a string, then the maximum length shall be 64 octets.

## 5 Architecture

### 5.1 Overview

The architecture enables resource based interactions among IoT artefacts, i.e. physical devices or applications. The architecture leverages existing industry standards and technologies and provides solutions for establishing connections (either wireless or wired) and managing the flow of information among devices, regardless of their form factors, operating systems or service providers.

Specifically, the architecture provides:

- A communication and interoperability framework for multiple market segments (Consumer, Enterprise, Industrial, Automotive, Health, etc.), OSs, platforms, modes of communication, transports and use cases
- A common and consistent model for describing the environment and enabling information and semantic interoperability
- Common communication protocols for discovery and connectivity
- Common security and identification mechanisms
- Opportunity for innovation and product differentiation
- A scalable solution addressing different device capabilities, applicable to smart devices as well as the smallest connected things and wearable devices

The architecture is based on the Resource Oriented Architecture design principles and described in the sections 5.2 through 5.6 respectively. Section 5.2 presents the guiding principles for OCF operations. Section 5.3 defines the functional block diagram and Framework. Section 5.5 provides an example scenario with roles. Section 5.6 provides an example scenario of bridging to non- OCF ecosystem.



## 5.2 Principle

In the architecture, Entities in the physical world (e.g., temperature sensor, an electric light or a home appliance) are represented as resources. Interactions with an Entity are achieved through its resource representations (section 7.7) using operations that adhere to Representational State Transfer (REST) architectural style, i.e., RESTful interactions.

The architecture defines the overall structure of the Framework as an information system and the interrelationships of the Entities that make up OCF. Entities are exposed as Resources, with their unique identifiers (URIs) and support interfaces that enable RESTful operations on the Resources. Every RESTful operation has an initiator of the operation (the client) and a responder to the operation (the server). In the Framework, the notion of the client and server is realized through roles (section 5.5). Any Device can act as a Client and initiate a RESTful operation on any Device acting as a Server. Likewise, any Device that exposes Entities as Resources acts as a Server. Conformant to the REST architectural style, each RESTful operation contains all the information necessary to understand the context of the interaction and is driven using a small set of generic operations, i.e., CREATE, RETRIEVE, UPDATE, DELETE and NOTIFY (CRUDN) defined in section 8, which include representations of Resources.

Figure 1 depicts the architecture.

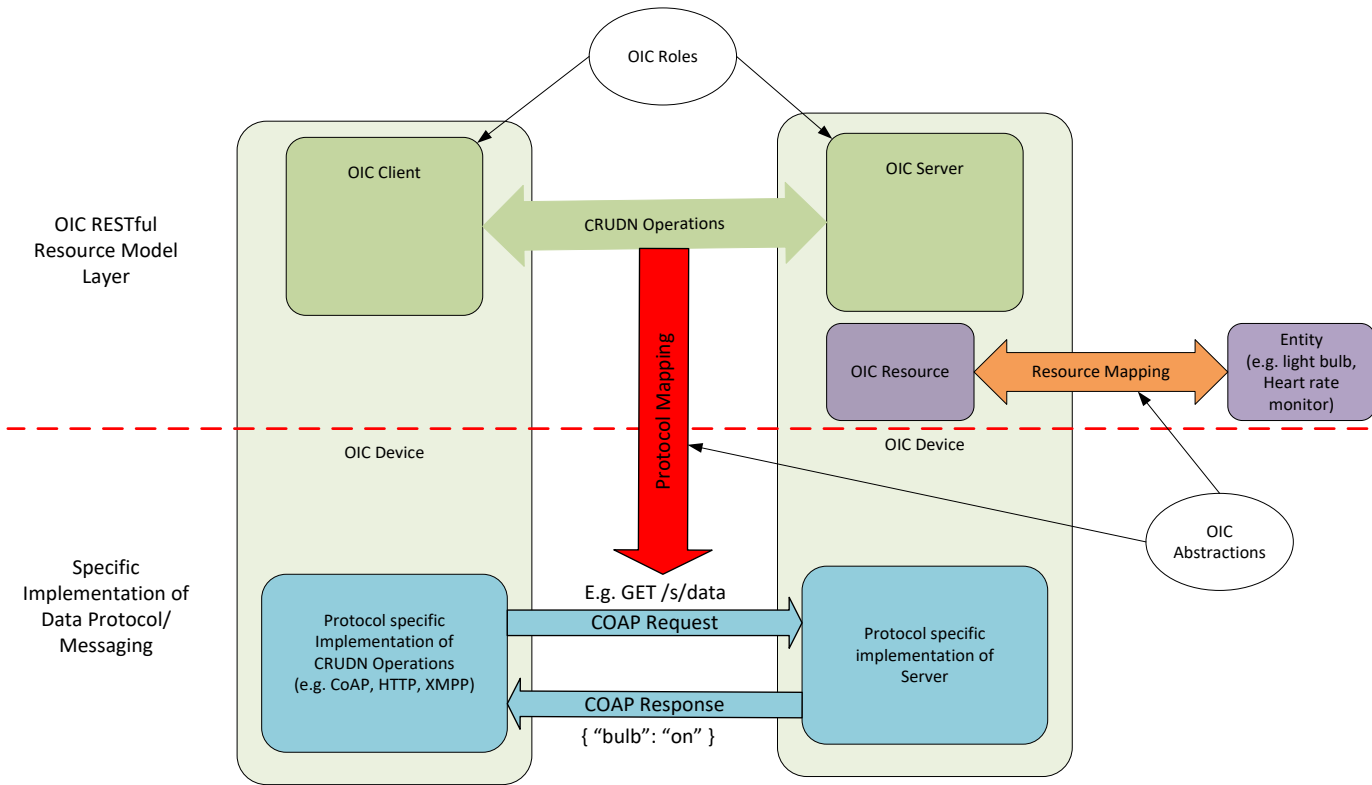


Figure 1: Architecture - concepts

The architecture is organized conceptually into three major aspects that provide overall separation of concern: resource model, RESTful operations and abstractions.

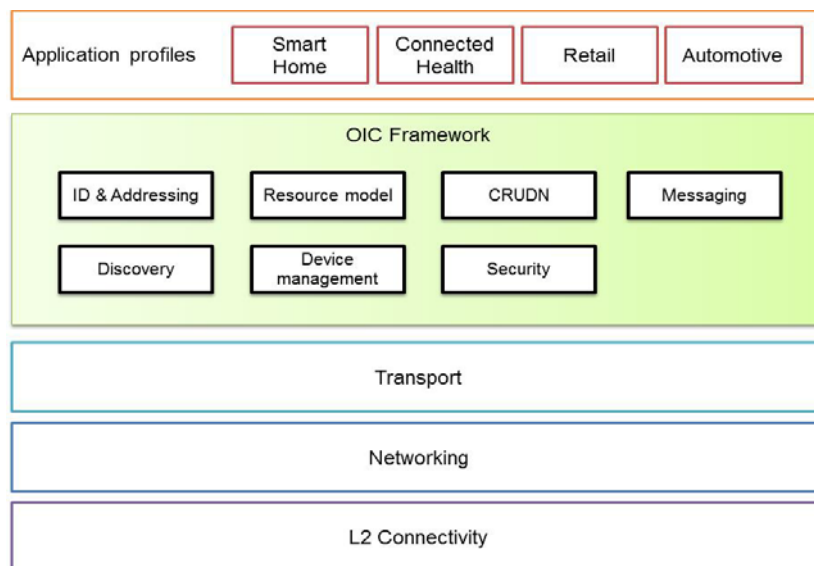
- Resource model: The resource model provides the abstractions and concepts required to logically model, and logically operate on the application and its environment. The core resource

model is common and agnostic to any specific application domain such as smart home, industrial or automotive. For example, the resource model defines a Resource which abstracts an Entity and the representation of a Resource maps the Entity's state. Other resource model concepts can be used to model other aspects, for example behaviour.

- **RESTful operations:** The generic CRUDN operations are defined using the RESTful paradigm to model the interactions with a Resource in a protocol and technology agnostic way. The specific communication or messaging protocols are part of the protocol abstraction and mapping of Resources to specific protocols is provided in section 11.8.
- **Abstraction:** The abstractions in the resource model and the RESTful operations are mapped to concrete elements using abstraction primitives. An entity handler is used to map an Entity to a Resource and connectivity abstraction primitives are used to map logical RESTful operations to data connectivity protocols or technologies. Entity handlers may also be used to map Resources to Entities that are reached over protocols that are not natively supported by OCF.

### 5.3 Functional block diagram

The functional block diagram encompasses all the functionalities required for operation. These functionalities are categorized as L2 connectivity, networking, transport, Framework, and application profiles. The functional blocks are depicted in Figure 2 and listed below.

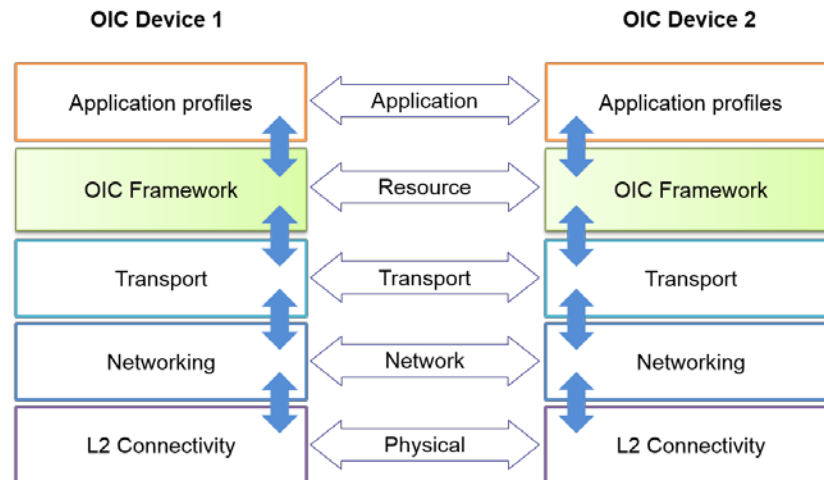


**Figure 2: Functional block diagram**

- **L2 connectivity:** Provides the functionalities required for establishing physical and data link layer connections (e.g., Wi-Fi™ or Bluetooth® connection) to the network.
- **Networking:** Provides functionalities required for Devices to exchange data among themselves over the network (e.g., Internet).
- **Transport:** Provides end-to-end flow transport with specific QoS constraints. Examples of a transport protocol include TCP and UDP or new Transport protocols under development in the IETF, e.g., Delay Tolerant Networking (DTN).
- **Framework:** Provides the core functionalities as defined in this specification. The functional block is the source of requests and responses that are the content of the communication between two Devices.

- **Application profile:** Provides market segment specific data model and functionalities, e.g., smart home data model and functions for the smart home market segment.

When two Devices communicate with each other, each functional block in a Device interacts with its counterpart in the peer Device as shown in Figure 3.



**Figure 3: Communication layering model**

## 5.4 Framework

Framework consists of functions which provide core functionalities for operation.

- 1) **Identification and addressing.** Defines the identifier and addressing capability. The Identification and addressing function is defined in section 6.
- 2) **Discovery.** Defines the process for discovering available
  - a) Devices (Endpoint Discovery in section 10) and
  - b) Resources (Resource discovery in section 11.3)
- 3) **Resource model.** Specifies the capability for representation of Entities in terms of resources and defines mechanisms for manipulating the resources. The resource model function is defined in section 7.
- 4) **CRUDN.** Provides a generic scheme for the interactions between a Client and Server as defined in section 8.
- 5) **Messaging.** Provides specific message protocols for RESTful operation, i.e. CRUDN. For example, CoAP is a primary messaging protocol. The messaging function is defined in section 11.8.
- 6) **Device management.** Specifies the discipline of managing the capabilities of a Device, and includes device provisioning and initial setup as well as device monitoring and diagnostics. The device management function is defined in section 11.5.
- 7) **Security.** Includes authentication, authorization, and access control mechanisms required for secure access to Entities. The security function is defined in section 13.

## 5.5 Example Scenario with roles

Interactions are defined between logical entities known as Roles. Three roles are defined: Client, Server and Intermediary.

Figure 4 illustrates an example of the Roles in a scenario where a smart phone sends a request message to a thermostat; the original request is sent over HTTP, but is translated into a CoAP

request message by a gateway in between, and then delivered to the thermostat. In this example, the smart phone takes the role of a Client, the gateway takes the role of an Intermediary and the thermostat takes the role of a Server.

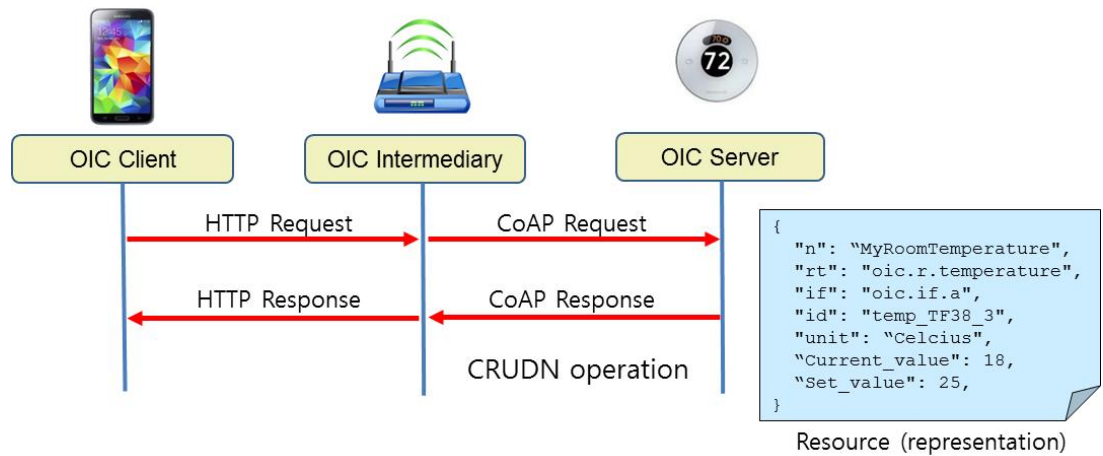


Figure 4: Example illustrating the Roles

### 5.6 Example Scenario: Bridging to Non- OCF ecosystem

The use case for this scenario is a display (like a wrist watch) that is used to monitor a heart rate sensor that implements a protocol that is not OCF supported.

Figure 5 provides a detailed logical view of the concepts described in Figure 1.

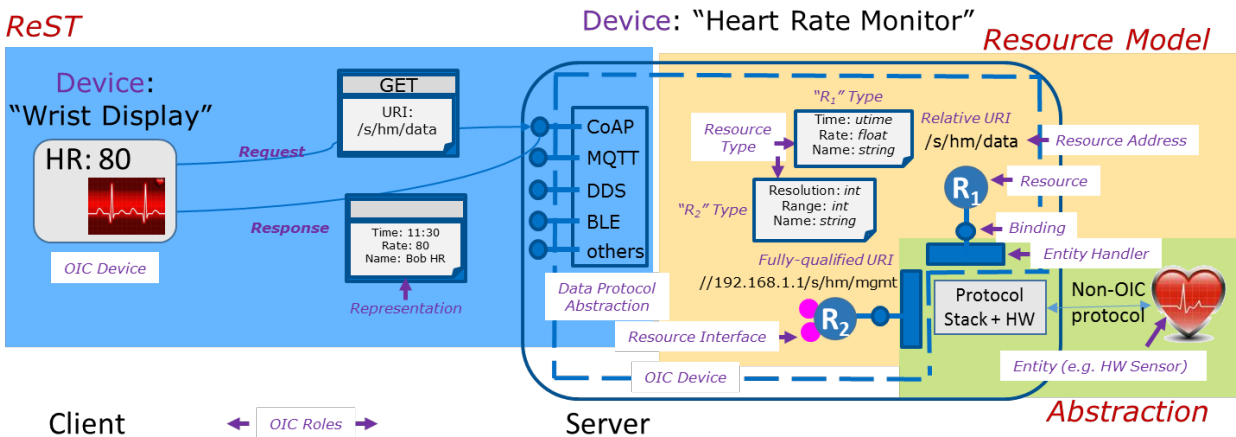
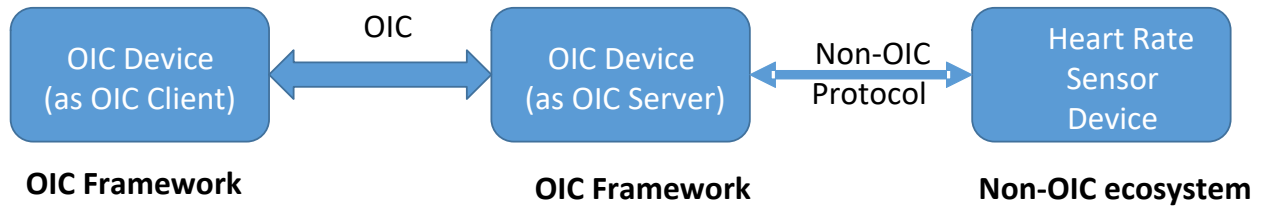


Figure 5: Framework - Architecture Detail

The details may be implemented in many ways, for example, by using a Server with an entity handler to interface directly to a non- OCF device as shown in Figure 6.



**Figure 6: Server bridging to Non- OCF device**

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

Once the resources are created and made discoverable, then the Display Device can discover these resources and operate on them using the mechanisms described in this specification. The requests to a resource on the Server are then interpreted by the entity handler and forwarded to the non- OCF device using the protocol supported by the non-OCF device. The returned information from the non- OCF device is then mapped to the appropriate response for that resource.

## 6 Identification and addressing

### 6.1 Introduction

Facilitating proper and efficient interactions between elements in the Framework, requires a means to identify, name and address these elements.

The *identifier* unambiguously identifies an element in a context or domain. The context or domain may be determined by the use or the application. The identifier is expected to be immutable over the lifecycle of that element and is unambiguous within a context or domain.

The *address* is used to define a place, way or means of reaching or accessing the element in order to interact with it. An address may be mutable based on the context.

The *name* is a handle that distinguishes the element from other elements in the framework. The name may be changed over the lifecycle of that element.

There may be methods or resolution schemes that allow determining any of these based on the knowledge of one or more of others (e.g., determine name from address or address from name).

Each of these aspects may be defined separately for multiple contexts (e.g., a context could be a layer in a stack). So an address may be a URL for addressing resource and an IP address for addressing at the connectivity layer. In some situations, both these addresses would be required. For example, to do RETRIEVE (section 8.3) operation on a particular resource representation, the client needs to know the address of the target resource and the address of the server through which the resource is exposed.

In a context or domain of use, a name or address could be used as identifier or vice versa. For example, a URL could be used as an identifier for a resource and designated as a URI.

The remainder of this section discusses the identifier, address and naming from the point of view of the resource model and the interactions to be supported by the resource model. Examples of interactions are the RESTful interactions, i.e. CRUDN operation (section 8) on a resource. Also the mapping of these to transport protocols, e.g., CoAP is described.

## 6.2 Identification

An identifier is unambiguous within the context or domain of use. There are many schemes that may be used to generate an identifier that has the required properties. The identifier may be context-specific in that the identifier is expected to be and guaranteed to be unambiguous only within that context or domain. Identifier may also be context-independent where these identifiers are guaranteed to be unambiguous across all contexts and domains both spatially and temporally. The context-specific identifiers could be defined by simple schemes like monotonic enumeration or may be defined by overloading an address or name, for example an IP address may be an identifier within the private domain behind a gateway in a smart home. On the other hand, context-independent identifiers require a stronger scheme that derives universally unique identities, for example any one of the versions of Universally Unique Identifiers (UUIDs). Context independent identifier may also be generated using hierarchy of domains where the root of the hierarchy is identified with a UUID and sub-domains may generate context independent identifier by concatenating context-specific identifiers for that domain to the context-independent identifier of their parent.

### 6.2.1 Resource identification and addressing

A resource may be identified using a URI and addressed by the same URI if the URI is a URL. In some cases a resource may need an identifier that is different from a URI; in this case, the resource may have a property whose value is the identifier. When the URI is in the form of a URL, then the URI may be used to address the resource.

An OCF URI is based on the general form of a URI as defined in IETF RFC 3986 as follows:

**<scheme>://<authority>/<path>?<query>**

Specifically the OCF URI is specified in the following form:

**ocf://<authority>/<path>?<query>**

A description of values that each component takes is given below.

The *scheme* for the URI is 'ocf'. The 'ocf' scheme represents the semantics, definitions and use as defined in this document. If a URI has the portion preceding the '/' (double slash) omitted, then the 'ocf' scheme shall be assumed.

Each transport binding is responsible for specifying how an OCF URI is converted to a transport protocol URI before sending over the network by the requestor. Similarly on the receiver side, each transport binding is responsible for specifying how an OCF URI is converted from a transport protocol URI before handing over to the resource model layer on the receiver.

The authority of an OCF URI shall be the Device ID ("di") value, as defined in [OCF Security], of the Server.

The *path* is a string that unambiguously identifies or references a resource within the context of the Server. In this version of the specification, a path shall not include pct-encoded non-ASCII characters or NUL characters. A *path* shall be preceded by a '/' (slash). The *path* may have '/' (slash) separated segments for human readability reasons. In the OCF context, the '/' (slash) separated segments are treated as a single string that directly references the resources (i.e. a flat structure) and not parsed as a hierarchy. On the Server, the path or some substring in the path may be shortened by using hashing or some other scheme provided the resulting reference is unique within the context of the host.

Once a path is generated, a Client accessing the resource or recipient of the URI should use that path as an opaque string and should not parse to infer a structure, organization or semantic.

1022 A query string shall contain a list of <name>=<value> segments (aka “name-value pair”) each  
1023 separated by a ‘&’ (ampersand). The query string will be mapped to the appropriate syntax of the  
1024 protocol used for messaging. (e.g., CoAP).

1025 A URI may be either

- 1026 • Fully qualified or
- 1027 • Relative

1028 *Generation of URI:*

1029 A URI may be defined by the Client which is the creator of that resource. Such a URI may be  
1030 relative or absolute (fully qualified). A relative URI shall be relative to the Device on which it is  
1031 hosted. Alternatively, a URI may be generated by the Server of that resource automatically based  
1032 on a pre-defined convention or organization of the resources, based on an interface, based on  
1033 some rules or with respect to different roots or bases.

1034 *Use of URI:*

1035 The absolute path reference of a URI is to be treated as an opaque string and a Client should not  
1036 infer any explicit or implied structure in the URI – the URI is simply an address. It is also  
1037 recommended that Devices hosting a resource treat the URI of each resource as an opaque string  
1038 that addresses only that resource. (e.g., URI's /a and /a/b are considered as distinct addresses  
1039 and resource b cannot be construed as a child of resource a).

### 1040 **6.3 Namespace:**

1041 The relative URI prefix “/oic/” is reserved as a namespace for URIs defined in OCF specifications  
1042 and shall not be used for URIs that are not defined in OCF specifications.

### 1043 **6.4 Network addressing**

1044 The following are the addresses used in this specification:

- 1045 • **IP address**

1046 An IP address is used when the device is using an IP configured interface.

1047 When a Device only has the identity information of its peer, a resolution mechanism is needed to  
1048 map the identifier to the corresponding address.

## 1049 **7 Resource model**

### 1050 **7.1 Introduction**

1051 The Resource Model defines concepts and mechanisms that provide consistency and core  
1052 interoperability between devices in the OCF ecosystems. The Resource Model concepts and  
1053 mechanisms are then mapped to the transport protocols to enable communication between the  
1054 devices – each transport provides the communication protocol interoperability. The Resource  
1055 Model, therefore, allows for interoperability to be defined independent of the transports.

1056 In addition, the concepts in the Resource Model support modelling of the primary artefacts and  
1057 their relationships to one and another and capture the semantic information required for  
1058 interoperability in a context. In this way, OCF goes beyond simple protocol interoperability to  
1059 capture the rich semantics required for true interoperability in Wearable and Internet of Things  
1060 ecosystems.

1061 The primary concepts in the Resource Model are: Entity, Resources, Uniform Resource Identifiers  
1062 (URI), Resource Types, Properties, Representations, Interfaces, Collections and Links. In addition,  
1063 the general mechanisms are CREATE, RETRIEVE, UPDATE, DELETE and NOTIFY. These  
1064 concepts and mechanisms may be composed in various ways to define the rich semantics and  
1065 interoperability needed for a diverse set of use cases that the OCF framework is applied to.

1066 In the OCF Resource Model framework, an Entity needs to be visible, interacted with or  
1067 manipulated, it is represented by an abstraction called a Resource. A Resource encapsulates and  
1068 represents the state of an Entity. A Resource is identified, addressed and named using URIs.

1069 Properties are "key=value" pairs and represent state of the Resource. A snapshot of these  
1070 Properties is the Representation of the Resource. A specific view of the Representation and the  
1071 mechanisms applicable in that view are specified as Interfaces. Interactions with a Resource are  
1072 done as Requests and Responses containing Representations.

1073 A resource instance is derived from a Resource Type. The uni-directional relationship between  
1074 one Resource and another Resource is defined as a Link. A Resource that has Properties and  
1075 Links is a Collection.

1076 A set of Properties can be used to define a state of a Resource. This state may be retrieved or  
1077 updated using appropriate Representations respectively in the response from and request to that  
1078 Resource.

1079 A Resource (and Resource Type) could represent and be used to expose a capability. Interactions  
1080 with that Resource can be used to exercise or use that capability. Such capabilities can be used  
1081 to define processes like discovery, management, advertisement etc. For example: "discovery of  
1082 resources on a device" can be defined as the retrieval of a representation of a specific resource  
1083 where a property or properties have values that describe or reference the resources on the device.

1084 The information for Request or Response with the Representation may be communicated "on the  
1085 wire" by serializing using a transfer protocol or encapsulated in the payload of the transport  
1086 protocol – the specific method is determined by the normative mapping of the Request or Response  
1087 to the transport protocol. See section 11.8 for transport protocols supported.

1088 The RAML definitions used in this document are normative. This also includes that all defined  
1089 JSON payloads shall comply with the indicated JSON schema. See Annex D for Resource Types  
1090 defined in this specification.

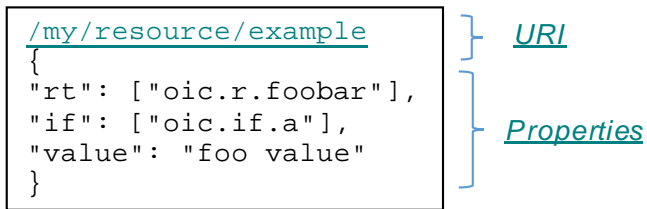
## 1091 **7.2 Resource**

1092 A Resource shall be defined by one or more Resource Type(s) – see Annex D for Resource Type.  
1093 A request to CREATE a Resource shall specify one or more Resource Types that define that  
1094 Resource.

1095 A Resource is hosted in a Device. A Resource shall have a URI as defined in section 6. The URI  
1096 may be assigned by the Authority at the creation of the Resource or may be pre-defined by the



1097 specification of the Resource Type.



1098

1099 **Figure 7: Example of a Resource**

1100

1101 Core Resources are the Resources defined in this specification to enable functional interactions  
1102 as defined in section 10 (e.g., Discovery, Device Management, etc). Among the Core Resources,  
1103 “/oic/res”, “/oic/p”, and “/oic/d” shall be supported on all Devices. Devices may support other Core  
1104 Resources depending on the functional interactions they support.

## 1105 **7.3 Property**

### 1106 **7.3.1 Introduction**

1107 A Property describes an aspect that is exposed through a Resource including meta-information  
1108 related to that resource.

1109 A Property shall have a name i.e. Property Name and a value i.e. Property Value. The Property is  
1110 expressed as a key-value pair where key is the Property Name and value the Property Value like  
1111 <Property Name> = <Property Value>. For example if the “temperature” Property has a Property  
1112 Name “temp” and a Property Value “30F”, then the Property is expressed as “temp=30F”. The  
1113 specific format of the Property depends on the encoding scheme. For example, in JSON, Property  
1114 is represented as "key": value (e.g., "temp": 30).

1115 In addition, the Property definition shall have a

- 1116 • **Value Type** – the Value Type defines the values that a Property Value may take. The Value  
1117 Type may be a simple data type (e.g. string, Boolean) as defined in section 3.4 or may be a  
1118 complex data type defined with a schema. The Value Type may define
  - 1119 ○ Value Rules define the rules for the set of values that the Property Value may take.  
1120 Such rules may define the range of values, the min-max, formulas, set of  
1121 enumerated values, patterns, conditional values and even dependencies on values  
1122 of other Properties. The rules may be used to validate the specific values in a  
1123 Property Value and flag errors.
- 1124 • **Mandatory** – specifies if the Property is mandatory or not for a given Resource Type.
- 1125 • **Access modes** – specifies whether the Property may be read, written or both. Updates are  
1126 equivalent to a write. “r” is used for read and “w” is used for write – both may be specified.  
1127 Write does not automatically imply read.

1128 The definition of a Property may include the following additional information – these items are  
1129 informative:

- 1130 • **Property Title** - a human-friendly name to designate the Property; usually not sent over the  
1131 wire
- 1132 • **Description** – descriptive text defining the purpose and expected use of this Property.

A Property may be used in the query part of an URI as one criterion for selection of a particular Resource. This is done by declaring the Property (i.e. <Property Name> = <desired Property Value>) as one of the segments of the query. In this version of the specification, only ASCII strings are permitted in query filters, and NUL characters are disallowed in query filters. This means that only property values with ASCII characters can be matched in a query filter. The Resource is selected when all the declared Properties in the query match the corresponding Properties in the full Representation of the target Resource. The full Representation is the snapshot that includes the union of all Properties in all Resource Types that define the target Resource. If the Property is declared in the “filter” segment of the query then the declared Property is matched to the Representation defined by the Interface to isolate certain parts of that Representation.

In general, a property is meaningful only within the resource to which it is associated. However a base set of properties that may be supported by all Resources, known as Common Properties, keep their semantics intact across Resources i.e. their “key=value” pair means the same in any Resource. Detailed tables with the above fields for all common properties are defined in section 7.3.2.

## **7.3.2 Common Properties**

### **7.3.2.1 Introduction**

The Common Properties defined in this section may be specified for all Resources. The following Properties are defined as Common Properties: “Resource Type”, “Resource Interface”, “Name”, and “Resource Identity”.

The name of a Common Property shall be unique and shall not be used by other properties. When defining a new Resource Type, its non-common properties shall not use the name of existing Common Properties (e.g., “rt”, “if”, “n”, “id”). When defining a new “Common Property”, it should be ensured that its name has not been used by any other properties. The uniqueness of a new Common Property name can be verified by checking all the Properties of all the existing OCF defined Resource Types. However, this may become cumbersome as the number of Resource Types grow. To prevent such name conflicts in the future, OCF may reserve a certain name space for common property. Potential approaches are (1) a specific prefix (e.g. “oic”) may be designated and the name preceded by the prefix (e.g. “oic.psize”) is only for Common Property; (2) the names consisting of one or two letters are reserved for Common Property and all other Properties shall have the name with the length larger than the 2 letters; (3) Common Properties may be nested under specific object to distinguish themselves.

The ability to UPDATE a Common Property (that supports write as an access mode) is restricted to the “oic.if.rw” (read-write) Interface; thus a Common Property shall be updatable using the read-write Interface if and only if the Property supports write access as defined by the Property definition and the associated schema for the read-write Interface.

The following Common Properties for all Resources are specified in section 7.3.2.2 through section 7.3.2.6 and summarized as follows:

- Resource Type (“rt”) – this Property is used to declare the Resource Type of that Resource. Since a Resource could be define by more than one Resource Type the Property Value of the Resource Type Property can be used to declare more than one Resource type. For example: “rt”: [“oic.wk.d”, “oic.d.airconditioner”] declares that the Resource containing this Property is defined by either the “oic.wk.d” Resource Type or the “oic.d.airconditioner” Resource Type. See section 7.3.2.3 for details.
- Interface (“if”) – this Property declares the Interfaces supported by the Resource. The Property Value of the Interface Property can be multi-valued and lists all the Interfaces supported. See section 7.3.2.4 for details.
- Name (“n”) – the Property declares “human-readable” name assigned to the Resource. See section 7.3.2.5.

- Resource Identity ("id"): its Property Value shall be a unique (across the scope of the host Server) instance identifier for a specific instance of the Resource. The encoding of this identifier is device and implementation dependent. See section 7.3.2.6 for details.

### 7.3.2.2 Property Name and Property Value definitions

The Property Name and Property Value as used in this specification:

- Property Name**— the key in "key=value" pair. Property Name is case sensitive and its data type is "string" but only ASCII characters are permitted, and embedded NUL characters are not permitted.
- Property Value** – the value in "key=value" pair. Property Value is case sensitive when its data type is "string". Any enum values shall be ASCII only.

### 7.3.2.3 Resource Type

Resource Type Property is specified in section 7.4.

### 7.3.2.4 Interface

Interface Property is specified in Section 7.5.

### 7.3.2.5 Name

A human friendly name for the Resource, i.e. a specific resource instance name (e.g., MyLivingRoomLight), The Name Property is as defined in Table 2

**Table 2. Name Property Definition**

Property title	Property name	Value type	Value rule	Unit	Access mode	Mandatory	Description
<b>Name</b>	n	string			R, W	no	Human understandable name for the resource.

The 'Name' Property is read-write unless otherwise restricted by the Resource Type (i.e. the Resource Type does not support UPDATE or does not support UPDATE using read-write).

### 7.3.2.6 Resource Identity

The Resource Identity Property shall be a unique (across the scope of the host Server) instance identifier for a specific instance of the Resource. The encoding of this identifier is device and implementation dependent. The Resource Identity Property is as defined in Table 3.

**Table 3. Resource Identity Property Definition**

Property title	Property name	Value type	Value rule	Unit	Access mode	Mandatory	Description
<b>Resource Identity</b>	id	string	Implementation Dependent		R	No	Unique identifier of the Resource (over all Resources in the Device)

## 7.4 Resource Type

### 7.4.1 Introduction

Resource Type is a class or category of Resources and a Resource is an instance of one or more Resource Types.

1213 The Resource Types of a Resource is declared using the Resource Type Common Property as  
1214 described in Section 7.3.2.3 or in a Link using the Resource Type Parameter.

1215 A Resource Type may either be pre-defined (Core Resource Types in this specification and Vertical  
1216 Resource Types in vertical domain specifications) or in custom definitions by manufacturers, end  
1217 users, or developers of Devices (vendor-defined Resource Types). Resource Types and their  
1218 definition details may be communicated out of band (i.e. in documentation) or be defined explicitly  
1219 using a meta-language which may be downloaded and used by APIs or applications. OCF has  
1220 adopted RAML and JSON Schema as the specification method for OCF's RESTful interfaces and  
1221 Resource definitions.

1222 Every Resource Type shall be identified with a Resource Type ID which shall be represented using  
1223 the requirements and ABNF governing the Resource Type attribute in IETF RFC 6690(Section 2  
1224 for ABNF and Section 3.1 for requirements) with the caveat that segments are separated by a "."  
1225 (period). The entire string represents the Resource Type ID. When defining the ID each segment  
1226 may represent any semantics that are appropriate to the Resource Type. For example, each  
1227 segment could represent a namespace. Once the ID has been defined, the ID should be used  
1228 opaquely and an implementations should not infer any information from the individual segments.  
1229 The string "oic", when used as the first segment in the definition of the Resource Type ID, is  
1230 reserved for OCF-defined Resource Types. All OCF defined Resource Types are to be registered  
1231 with the IANA Core Parameters registry as described also in IETF RFC 6690.

## 1232 7.4.2 Resource Type Property

1233 A Resource when instantiated or created shall have one or more Resource Types that are the  
1234 template for that Resource. The Resource Types that the Resource conforms to shall be declared  
1235 using the "rt" Common Property for the Resource. The Property Value for the "rt" Common Property  
1236 shall be the list of Resource Type IDs for the Resource Types used as templates (i.e., "rt"=<list of  
1237 Resource Type IDs>).

1238 **Table 4. Resource Type Common Property definition**

Property title	Property name	Value type	Value rule	Unit	Access mode	Mandatory	Description
Resource type	rt	array	Array of strings, conveying resource Type IDs		R	yes	The property name rt is as described in IETF RFC 6690

1239 Resource Types may be explicitly discovered or implicitly shared between the user (i.e. Client) and  
1240 the host (i.e. Server) of the Resource.

## 1241 7.4.3 Resource Type definition

1242 Resource Type is specified as follows:

- 1243 • **Pre-defined URI** (optional) – a pre-defined URI may be specified for a specific Resource Type  
1244 in an OCF specification. When a Resource Type has a pre-defined URI, all instances of that  
1245 Resource Type shall use only the pre-defined URI. An instance of a different Resource Type  
1246 shall not use the pre-defined URI.
- 1247 • **Resource Type Title (optional)** – a human friendly name to designate the Resource Type.
- 1248 • **Resource Type ID** – the value of "rt" property which identifies the Resource Type, (e.g.,  
1249 "oic.wk.p").
- 1250 • **Resource Interfaces** – list of the interfaces that may be supported by the Resource Type.

- **Resource Properties** – definition of all the properties that apply to the Resource Type. The Resource Type definition shall define whether a property is mandatory, conditional mandatory, or optional.
  - **Related Resource Types** (optional) – the specification of other Resource Types that may be referenced as part of the Resource Type, applicable to collections.
  - **Mime Types** (optional) – mime types supported by the resource including serializations (e.g., application/cbor, application/json, application/xml).
- Table 5 and Table 6 provide an example description of an illustrative foobar Resource Type and its associated Properties.

**Table 5. Example foobar Resource Type**

Pre-defined URI	Resource Type Title	Resource Type ID ("rt" value)	interfaces	Description	Related Functional Interaction	M/CR/O
none	foobar	oic.r.foobar	"oic.if.a"	Example "foobar" resource	Actuation	O

**Table 6. Example foobar properties**

Property title	Property name	Value type	Value rule	Unit	Access mode	Mandatory	Description
Resource Type	rt	array			R	yes	Resource Type
Interface	if	array			R	yes	Interface
Foo value	value	string			R	yes	Foo value

An instance of the foobar Resource Type is as shown below

```
{
  "rt": ["oic.r.foobar"],
  "if": ["oic.if.a"],
  "value": "foo value"
}
```

An example schema for the foobar Resource Type is shown below

```
{
  "$schema": "http://json-schema.org/draft-04/schema",
  "type": "object",
  "properties": {
    "rt": {"type": "string"},
    "if": {"type": "string"},
    "value": {"type": "string"}
  },
  "required": ["rt", "if", "value"]
}
```

#### 7.4.4 Multi-value "rt" Resource

Multi-value "rt" Resource means a Resource with multiple Resource Types. Such a Resource is associated with multiple Resource Types and so has an "rt" Property Value of multiple Resource Type IDs (e.g. "rt": ["oic.r.switch.binary", "oic.r.light.brightness"]). The order of the Resource Type IDs in the "rt" Property Value is meaningless. For example, "rt": ["oic.r.switch.binary", "oic.r.light.brightness"] and "rt": ["oic.r.light.brightness", "oic.r.switch.binary"] have the same meaning.

Resource Types for multi-value "rt" Resources shall satisfy the following conditions.

- **Property Name** – Property Names for each Resource Type shall be unique (within the scope of the multi-value "rt" Resource) with the exception of Common Properties, otherwise there will be conflicting Property semantics. If two Resource Types have a Property with the same Property Name, a multi-value "rt" Resource shall not be composed of these Resource Types.

A multi-value "rt" Resource satisfies all the requirements for each Resource Type and conforms to the RAML/JSON definitions for each component Resource Type. Thus the mandatory Properties of a multi-value "rt" Resource shall be the union of all the mandatory Properties of each Resource Type. For example, mandatory Properties of a Resource with "rt": ["oic.r.switch.binary", "oic.r.light.brightness"] are "value" and "brightness", where the former is mandatory for "oic.r.switch.binary" and the latter for "oic.r.light.brightness".

The multi-value "rt" Resource Interface set shall be the union of the sets of interfaces from the component Resource Types. The Resource Representation in response to a CRUDN action on an Interface shall be the union of the schemas that are defined for that Interface. The Default Interface for a multi-value "rt" Resource shall be the baseline Interface ("oic.if.baseline") as that is the only guaranteed common Interface between the Resource Types.

For clarity if each Resource Type supports the same set of Interfaces, then the resultant multi-value "rt" Resource has that same set of Interfaces with a Default Interface of baseline ("oic.if.baseline").

An "rt" query for a multi-value "rt" Resource with the Default Interface of "oic.if.a", "oic.if.s", "oic.if.r", "oic.if.rw" or "oic.if.baseline" is an extension of a generic "rt" query. When a Server receives a RETRIEVE request for multi-value "rt" Resource with an "rt" query, (i.e. GET /ResExample?rt=oic.r.foo), the Server should respond only when the query value is an item of the "rt" Property Value of the target Resource and should send back only the Properties associated with the query value. For example, upon receiving GET /ResExample?rt=oic.r.switch.binary targeting a Resource with "rt": ["oic.r.switch.binary", "oic.r.light.brightness"], the Server responds with only the Properties of oic.r.switch.binary.

#### 7.5 Device Type

A Device Type is a class of Device. Each Device Type defined will include a list of minimum Resource Types that a device shall implement for that Device Type. A device may expose additional standard and vendor defined Resource Types beyond the minimum list. The Device Type is used in Resource discovery as specified in section 11.3.4.

Like a Resource Type, a Device Type can be used in the Resource Type Common Property or in a Link using the Resource Type Parameter.

A Device Type may either be pre-defined (in vertical domain specifications) or in custom definitions by manufacturers, end users, or developers of Devices (vendor-defined Device Types). Device Types and their definition details may be communicated out of band (like in documentation).

1312 Every Device Type shall be identified with a Resource Type ID using the same syntax constraints  
1313 as a Resource Type.

1314 **7.6 Interface**

1315 **7.6.1 Introduction**

1316 An Interface provides first a view into the Resource and then defines the requests and responses  
1317 permissible on that view of the Resource. So this view provided by an Interface defines the context  
1318 for requests and responses on a Resource. Therefore, the same request to a Resource when  
1319 targeted to different Interfaces may result in different responses.

1320 An Interface may be defined by either this specification (a Core Interface), the OCF vertical domain  
1321 specifications (a “vertical Interface) or manufacturers, end users or developers of Devices (a  
1322 “vendor-defined Interface”).

1323 The Interface Property lists all the Interfaces the Resource support. All resources shall have at  
1324 least one Interface. The Default Interface shall be defined by an OCF specification and inherited  
1325 from the Resource Type definition. The Default Interface associated with all Resource Types  
1326 defined in this specification shall be the supported Interface listed first within the applicable  
1327 enumeration in the definition of the Resource Type (see Annex D). All Default Interfaces specified  
1328 in an OCF specification shall be mandatory.

1329 In addition to any OCF specification defined interface, all Resources shall support the Baseline  
1330 Interface (“oic.if.baseline”) as defined in section 7.6.3.2.

1331 When an Interface is to be selected for a Request, it shall be specified as query parameter in the  
1332 URI of the Resource in the Request message. If no query parameter is specified, then the Default  
1333 Interface shall be used. If the selected Interface is not one of the permitted Interfaces on the  
1334 Resource then selecting that Interface is an error.

1335 An Interface may accept more than one media type. An Interface may respond with more than one  
1336 media type. The accepted media types may be different from the response media types. The media  
1337 types are specified with the appropriate header parameters in the transfer protocol. (NOTE: This  
1338 feature has to be used judiciously and is allowed to optimize representations on the wire) Each  
1339 Interface shall have at least one media type.

1340

1341 **7.6.2 Interface Property**

1342 **Table 7. Resource Interface Property definition**

Property title	Property name	Value type	Value rule	Unit	Access mode	Mandatory	Description
Interface	if	array	Array of strings, conveying interfaces		R	yes	Property to declare the Interfaces supported by a Resource.

1343 The Interfaces supported by a Resource shall be declared using the Interface Common Property  
1344 (Table 7) as "if=<array of Interfaces>". The Property Value of an Interface Property shall be a  
1345 lower case string with segments separated by a "." (dot). The string "oic", when used as the first  
1346 segment in the Interface Property Value, is reserved for OCF-defined Interfaces. The Interface  
1347 Property Value may also be a reference to an authority similar to IANA that may be used to find  
1348 the definition of an Interface. A Resource Type shall support one or more of the Interfaces defined  
1349 in section 7.6.3.

## 7.6.3 Interface methods

### 7.6.3.1 Overview

The OCF -defined Interfaces are listed in the table below:

**Table 8. OCF standard Interfaces**

Interface	Name	Applicable Methods	Description
baseline	"oic.if.baseline"	RETRIEVE, UPDATE	The baseline Interface defines a view into all Properties of a Resource including the Meta Properties. This Interface is used to operate on the full Representation of a Resource.
links list	"oic.if.ll"	RETRIEVE	The 'links list' Interface provides a view into Links in a Collection (Resource). Since Links represent relationships to other Resources, the links list interfaces may be used to discover Resources with respect to a context. The discovery is done by retrieving Links to these Resources. For example: the Core Resource "/oic/res" uses this Interface to allow discovery of Resource "hosted" on a Device.
batch	"oic.if.b"	RETRIEVE, UPDATE	The batch Interface is used to interact with a collection of Resources at the same time. This also removes the need for the Client to first discover the Resources it is manipulating – the Server forwards the requests and aggregates the responses
read-only	"oic.if.r"	RETRIEVE	The read-only Interface exposes the Properties of a Resource that may be 'read'. This Interface does not provide methods to update Properties or a Resource and so can only be used to 'read' Property Values.
read-write	"oic.if.rw"	RETRIEVE, UPDATE	The read-write Interface exposes only those Properties that may be both 'read' and "written" and provides methods to read and write the Properties of a Resource.
actuator	"oic.if.a"	CREATE, RETRIEVE, UPDATE	The actuator Interface is used to read or write the Properties of an actuator Resource.
sensor	"oic.if.s"	RETRIEVE	The sensor Interface is used to read the Properties of a sensor Resource.

### 7.6.3.2 Baseline Interface

#### 7.6.3.2.1 Overview

The Representation that is visible using the "baseline" Interface includes all the Properties of the Resource including the Common Properties. The "baseline" Interface shall be defined for all Resource Types. All Resources shall support the "baseline" Interface.

The "baseline" Interface is selected by adding "if=oic.if.baseline" to the list of query parameters in the URI of the target Resource. For example: "GET /oic/res?if=oic.if.baseline".

#### 7.6.3.2.2 Use of RETRIEVE

The "baseline" Interface is used when a Client wants to retrieve all Properties of a Resource. The Client includes the URI query parameter definition "?if=oic.if.baseline" in a RETRIEVE request. When this query parameter definition is included the Server shall respond with a Resource representation that includes all of the implemented Properties of the Resource. When the Server is unable to send back the whole Resource representation, it shall reply with an error message. The Server shall not return a partial Resource representation.



1370 An example response to a RETRIEVE request using the baseline Interface is shown below:

```
{
  "rt": ["oic.r.temperature"],
  "if": ["oic.if.a", "oic.if.baseline"],
  "temperature": 20,
  "units": "C",
  "range": [0,100]
}
```

1371

1372 **7.6.3.2.3 Use of UPDATE**

1373 Using the baseline Interface, all Properties of a Resource with the exception of Common Properties  
1374 may be modified using an UPDATE request with a list of Properties and their desired values if a  
1375 Resource Type has an associated schema for UPDATE using baseline.

1376 **7.6.3.3 Link List Interface**

1377 **7.6.3.3.1 Overview**

1378 The links list Interface provides a view into the list of Links in a Collection (Resource). The  
1379 Representation visible through this Interface has only the Links defined in the Property Value of  
1380 the “links” Property – so this Interface is used to manipulate or interact with the list of Links in a  
1381 Collection. The Links list may be RETRIEVED using this Interface.

1382 The Interface definition and semantics are given as follows:

- 1383 • The links list Interface name shall be “oic.if.ll”.
- 1384 • If specified in a request (usually in the request header), the serialization in the response shall  
1385 be in the format expected in the request.
- 1386 • In response to a RETRIEVE request on the “links list” Interface, the URIs of the referenced  
1387 Resources shall be returned as a URI reference.
- 1388 • If there are no links present in a Resource, then an empty list shall be returned.
- 1389 • The Representation determined by this Interface view only includes the Property Value of the  
1390 “links” Property. Hence Collection or /oic/res response with oic.if.ll is an array of OCF Links.

1391 **7.6.3.3.2 Example: “links list” Interface**

1392 **Example: Request to a Collection**

<b>Request to RETRIEVE the Links in room</b>  (the Links could be referencing lights, fans, electric sockets etc)	<pre>GET ocf://&lt;devID&gt;/a/room/1?if=oic.if.ll</pre> <p>The response would be the array of OCF Links</p> <pre>[   {     "href": "/the/light/1",     "rt": ["oic.r.switch.binary"],     "if": ["oic.if.a", "oic.if.baseline"],</pre>
---	---

	<pre> "eps": [   {     "ep": "coaps://[2001:db8:a::b1d4]:55555"   } ], {   "href": "/the/light/2",   "rt": ["oic.r.switch.binary"],   "if": ["oic.if.a", "oic.if.baseline"],   "eps": [     {       "ep": "coaps://[2001:db8:a::b1d4]:55555"     }   ],   {     "href": "/my/fan/1",     "rt": ["oic.r.switch.binary"],     "if": ["oic.if.a", "oic.if.baseline"],     "eps": [       {         "ep": "coaps://[2001:db8:a::b1d4]:55555"       }     ],     {       "href": "/his/fan/2",       "rt": ["oic.r.switch.binary"],       "if": ["oic.if.a", "oic.if.baseline"],       "eps": [         {           "ep": "coaps://[2001:db8:a::b1d4]:55555"         }       ]     }   ] </pre>
--	--

1393

#### 1394 7.6.3.4 Batch Interface

##### 1395 7.6.3.4.1 Overview

1396 The batch Interface is used to interact with a collection of Resources using a single/same Request.  
 1397 The batch Interface supports methods of Resources in the Links of the Collection, and can be used  
 1398 to RETRIEVE or UPDATE the Properties of the “linked” Resources with a single Resource  
 1399 representation.

1400 The batch Interface selects a view into the Links in a Collection – the Request is sent to all the  
 1401 Links in this view with potential modifications defined in the Parameters of the Link.

1402 The batch Interface is defined as follows:

- 1403 • The batch Interface name shall be "oic.if.b"
- 1404 • A Collection Resource with a batch Interface has Links that have Resource references that  
1405 may be URIs (fully qualified for remote Resources) or relative references (for local Resources).
- 1406 • The original request is modified to create new requests targeting each of the targets in the  
1407 Resource Links by substituting the URI in the original request with the URI of the target  
1408 Resource in the Link. The payload in the original request is replicated in the payload of the  
1409 new Requests.
- 1410 • The Requests shall be forwarded assuming use of the Default Interface of the referenced  
1411 Resources.
- 1412 • Requests shall only be forwarded to link targets that are identified as items in the collection by  
1413 relation types "item" or "hosts" ("hosts" is the default relation type). Requests shall not be  
1414 forwarded to targets of links that do not contain the "item" or "hosts" relation type values. The  
1415 default relation type "hosts" shall be allowed for relative and absolute links.
- 1416 • Resources of the collection itself may be included in the batch by using the link relation "self"  
1417 along with "item" and insuring that the default interface of the collection does not expose the  
1418 links property, i.e. not "oic.if.baseline" or "oic.if.ll".
- 1419 • All the Responses from the linked item Resources shall be aggregated into single Response  
1420 to the Client. The Server may timeout the Response to a time window (if a time window has  
1421 been negotiated with the Client then the Server shall not timeout within that window; in the  
1422 absence of negotiated window, the Server may choose any appropriate window based on  
1423 conditions). If the target Resources cannot process the new request, an empty response or  
1424 error response shall be returned. These empty/error Responses shall be included in  
1425 aggregated Response to the original Client Request.
- 1426 • The batch representation is an array of objects representing the responses from the linked  
1427 resources. Each object in the batch response shall include at least two items: (1) the URI of  
1428 the linked resource (fully qualified for remote resources, or a relative reference for local  
1429 resources) as "href": <URI> and (2) the individual response object as "rep" as the key i.e. "rep":  
1430 { < representation of individual response> }.
- 1431 • Resources referenced by links in the collection may be observed using the batch interface of  
1432 the collection. The observe mechanism shall work as defined in 11.4.2. Specifically, the  
1433 representations and status codes shall be the same as for RETRIEVE operations using the  
1434 Batch interface.
- 1435 • Properties of the collection resource itself may be observed by using the appropriate interface.  
1436 For example, a collection may be observed on its linked list or baseline interface to receive  
1437 notifications of changes to its links.
- 1438 • The Client may choose to restrict the list of Links to which the Request is forwarded by including  
1439 query parameters in the URI of the Collection to which this original 'batch' Interface Request  
1440 is made. The Server should process query parameters in a request that includes "oic.if.b", as  
1441 selectors for links in the Collection that are to be processed in the batch.
- 1442 • Batch UPDATE operations are performed by creating a payload according to the same schema  
1443 of the Batch RETRIEVE payload. A set of link-specific UPDATE requests is created according  
1444 to the "href" tags in the included items, and the payload contained in the value of the "rep"  
1445 property is applied to the corresponding "href" referenced item.
- 1446 • If requested property for UPDATE does not exist in linked resource, it shall silently ignore the  
1447 request.
- 1448 • If the "href" value is the empty URI, denoted by a zero length string or "" in JSON, the payload  
1449 in the value of the "rep" property is applied to all batch items in the Collection.

- Items with the empty "href" and link-specific "href" shall not be mixed in the same UPDATE payload.
- The Representation in the Link-specific Request may not match the Representation exposed by the Default Interface on the target Resource. In such cases, the 'subset' semantics apply where Properties in the Request which match Properties in the Resource view exposed shall be modified in the target Resource if the Property is writeable.
- The response to POST shall contain the updated values using the same payload schema as RETRIEVE operations, along with the appropriate status code. The response payload shall reflect the known state of the updated resource properties after the batch update was completed.

#### 7.6.3.4.2 Use of Query Parameters with Batch

Additional query parameters may be used with the batch interface in order to select particular items in the batch for retrieval or update. When additional parameters are included which are not interpreted in other ways, these parameters are used to select items in the batch by matching link attribute values.

A particular item in a batch is selected by additional query parameters in a request if, and only if, all of the link selection query parameters contained values which match corresponding values in the link attributes of that item.

When link selection query parameters are used with RETRIEVE operations, only the items with matching link attributes are returned.

When link selection query parameters are used with UPDATE operations, only the items having matching link attributes are updated.

See 7.6.3.4.3 for examples of RETRIEVE and UPDATE operations that use link selection query parameters.

#### 7.6.3.4.3 Examples: Batch Interface

Example 1

Resources	<pre> /a/room/1  {   "rt": ["oic.wk.col", "x.org.example.rt.room"],   "if": ["oic.if.baseline", "oic.if.b", "oic.if.ll", "oic.if.r"],   "x.org.example.color": "blue",   "x.org.example.dimension": "15bx15wx10h",   "links": [     {       "href": "/a/room/1", "rel": ["self", "item"], "rt": ["x.org.example.rt.room"], "if": ["oic.if.r", "oic.if.baseline", "oic.if.b", "oic.if.ll"] },     {       "href": "/the/light/1", "rel": ["item"], "rt": ["oic.r.switch.binary"], "if": ["oic.if.a", "oic.if.baseline"]}   ] }</pre>
-----------	---

```

        {"href": "/the/light/2", "rel": ["item"], "rt":
["oic.r.switch.binary"], if=["oic.if.a" ,"oic.if.baseline"]},

        {"href": "/my/fan/1", "rel": ["item"], "rt":
["oic.r.switch.binary"], if=["","oic.if.a", "oic.if.baseline"]},

        {"href": "/his/fan/2", "rel": ["item"], "rt":
["oic.r.switch.binary"], if=["oic.if.a", "oic.if.baseline"]}

    ]
}

/the/light/1
{
    "rt": ["oic.r.switch.binary"],
    "ins": "light-1",
    "if": ["oic.if.a", "oic.if.baseline"],
    "value": false
}

/the/light/2
{
    "rt": ["oic.r.switch.binary"],
    "ins": "light-2",
    "if": ["oic.if.a", "oic.if.baseline"],
    "value": true
}

/the/fan/1
{
    "rt": ["oic.r.switch.binary"],
    "ins": "fan-1",
    "if": ["oic.if.a", "oic.if.baseline"],
    "value": true
}

```

	<pre> }  /the/fan/2  {   "rt": ["oic.r.switch.binary"],   "ins": "fan-2",   "if": ["oic.if.a", "oic.if.baseline"],   "value": false } </pre>
Use of batch	<p>Request: GET /a/room/1?if=oic.if.b</p> <p>Becomes the following individual request messages issued by the Device in the Client role</p> <p>GET /a/room/1 (NOTE: Uses the default Interface as specified for this resource)</p> <p>GET /the/light/1 (NOTE: Uses the default Interface as specified for this resource)</p> <p>GET /the/light/2 (NOTE: Uses the default Interface as specified for this resource)</p> <p>GET /the/fan/1 (NOTE: Uses the default Interface as specified for this resource)</p> <p>GET /the/fan/2 (NOTE: Uses the default Interface as specified for this resource)</p> <p>Response:</p> <pre> [   {     "href": "/a/room/1",     "rep": {       "blue": "blue",       "x.org.example.dimension": "15bx15wx10h",       "x.org.example.color": "x.org.example.color"     }   },   {     "href": "/the/light/1",     "rep": {"value": false}   }, ] </pre>

	<pre> {   "href": "/the/light/2",   "rep": {"value": true} }, {   "href": "/my/fan/1",   "rep": {"value": true} }, {   "href": "/his/fan/2",   "rep": {"value": false} } ] </pre>
<p>Use of batch</p> <p>(UPDATE has POST semantics)</p>	<pre> UPDATE /a/room/1?if=oic.if.b [   {     "href": "",     "rep": {       "value": false     }   } ] </pre> <p>Since the "href" value in the batch update payload item is the empty URI, the request is forwarded to all items in the batch and becomes:</p> <pre> UPDATE /a/room/1 { "value": false } UPDATE /the/light/1 { "value": false } UPDATE /the/light/2 { "value": false } UPDATE /my/fan/1 { "value": false } UPDATE /his/fan/2 { "value": false } </pre> <p>The response will be same as response for GET /a/room/1?if=oic.if.b.</p> <p>Since /a/room/1 does not have a "value" property exposed by its default interface, the update request will be silently ignored.</p>
Use of batch	<pre> UPDATE /a/room/1?if=oic.if.b [   {     "href": "/the/light/1",     "rep": {       "value": false     }   } ] </pre>

<p>(UPDATE has POST semantics)</p>	<pre>     },     {       "href": "/the/light/2",       "rep": {         "value": true       }     },     {       "href": "/a/room/1",       "rep": {         "x.org.example.color": "red"       }     }   ] </pre> <p>This turns /the/light/1 off, turns /the/light/2 on, and sets the color of the room to "red".</p> <p>The response will be same as response for GET /a/room/1?if=oic.if.b.</p> <p>Example use of additional query parameters to select items by matching link attributes.</p> <p>Turn on light 1 based on the "ins" link attribute value of "light-1"</p> <pre> UPDATE /a/room/1?if=oic.if.b&amp;ins=light-1 [   {     "href": "",     "rep": {       "value": false     }   } ] </pre> <p>Similar to the earlier example, "href": "" applies the payload to all selected links. Since the additional query parameter ins=light-1 selects only links that have a matching "ins" value, only one link is selected. The payload is applied to the target resource of that link, /the/light/1.</p> <p>Retrieving the item using the same query parameter:</p> <pre> RETRIEVE /a/room/1?if=oic.if.b&amp;ins=light-1 </pre> <p>Response payload:</p> <pre> [   {     "href": "",     "rep": {       "value": false     }   } ] </pre>
------------------------------------	--



1477 Example that further shows the “links list” and “batch” interface

Example	<pre>/myexample {   "rt": ["oic.r.foo"],   "if": [ "oic.if.baseline", "oic.if.ll" ],   "links": [     { "href": "/acme/switch", "di": "&lt;deviceID1&gt;", "rt": ["oic.r.switch.binary"], "if": ["oic.if.a"] },     { "href": "ocf://&lt;deviceID1&gt;/acme/fan", "rt": ["oic.r.fan"], "if": ["oic.if.a"] }   ] }</pre>
Use of Baseline	<pre>GET /myexample?if=oic.if.baseline will return {   "rt": ["oic.r.foo"],   "if": [ "oic.if.baseline", "oic.if.ll" ],   "links": [     { "href": "/acme/switch", "di": "&lt;deviceID1&gt;", "rt": ["oic.r.switch.binary"], "if": ["oic.if.a"] },     { "href": "ocf://&lt;deviceID1&gt;/acme/fan", "rt": "oic.r.fan", "if": "oic.if.a" }   ] }</pre>
Use of Links List	<pre>GET /myexample?if=oic.if.ll. will return  [   { "href": "/acme/switch", "di": "&lt;deviceID1&gt;", "rt": ["oic.r.switch.binary"], "if": ["oic.if.a"] },   { "href": "ocf://&lt;deviceID1&gt;/acme/fan", "rt": ["oic.r.fan"], "if": ["oic.if.a"] } ]</pre>

- 1478
- 1479 **7.6.3.5 Actuator Interface**
- 1480 The actuator Interface is the Interface for viewing Resources that may be actuated i.e. changes
- 1481 some value within or the state of the entity abstracted by the Resource:
- 1482 
  - The actuator Interface name shall be “oic.if.a”
  - The actuator Interface shall expose in the Resource Representation all mandatory Properties as defined by the applicable JSON; the actuator interface may also expose in the Resource Representation optional Properties as defined by the applicable JSON schema that are implemented by the target Device.
- 1483
- 1484
- 1485
- 1486

For the following Resource

**NOTE: “prm” is the Property name for ‘parameters’ Property**

```
/a/act/heater
{
  "rt": ["acme.gas"],
  "if": ["oic.if.baseline", "oic.if.r", "oic.if.a", "oic.if.s"],
  "prm": { "sensitivity": 5, "units": "C", "range": "0 .. 10" },
  "settemp": 10,
```

```
    "currenttemp" : 7
}
```

**Figure 8: Example - "Heater" Resource (for illustration only)**

**NOTE: The example here is with respect to Figure 8**

**1. Retrieving values of an actuator**

Request: GET /a/act/heater?if="oic.if.a"

Response:

```
{
  "prm": { "sensitivity": 5, "units": "C", "range": "0 .. 10" },
  "settemp": 10,
  "currenttemp" : 7
}
```

**2. Correct use of actuator:**

Request: POST /a/act/heater?if="oic.if.a"

```
{
  "settemp": 20
}
```

Response:

```
{
  Ok
}
```

**3. Incorrect use of actuator**

Request: POST /a/act/heater?if="oic.if.a"

```
{
  "if": [ "oic.if.s" ]   ← this is visible through baseline
}
```

Interface

Response:

```
{
  Error
}
```

**Figure 9: Example - Actuator Interface**

- A RETRIEVE request using this Interface shall return the Representation for this Resource subject to any query and filter parameters that may also exist
- An UPDATE request using this Interface shall provide a payload or body that contains the Properties that will be updated on the target Resource.

### 7.6.3.6 Sensor Interface

The sensor Interface is the Interface for retrieving measured, sensed or capability specific information from a Resource that senses:

- The sensor Interface name shall be "oic.if.s"
- The sensor Interface shall expose in the Resource Representation all mandatory Properties as defined by the applicable JSON; the sensor interface may also expose in the Resource

Representation optional Properties as defined by the applicable JSON schema that are implemented by the target Device.

- A RETRIEVE request using this Interface shall return this Representation for the Resource subject to any query and filter parameters that may also exist

**NOTE: The example here is with respect to Figure 8**

**1. Retrieving values of sensor**

Request: GET /a/act/heater?if="oic.if.s"

Response:  
{  
 "currenttemp": 7  
}

**2. Incorrect use of sensor**

Request: PUT /a/act/heater?if="oic.if.s" ← PUT is not allowed  
{  
 "settemp": 20 ← this is possible through actuator Interface  
}

Response:  
{  
 Error  
}

**3. Incorrect use of sensor**

Request: POST /a/act/heater?if="oic.if.s" ← POST is not allowed  
{  
 "currenttemp": 15 ← this is possible through actuator Interface  
}

Response:  
{  
 Error  
}

### 7.6.3.7 Read-only Interface

The read-only Interface exposes only the Properties that may be “read”. This includes Properties that may be “read-only”, “read-write” but not Properties that are “write-only” or “set-only”. The applicable methods that can be applied to a Resource is RETRIEVE only. An attempt by a Client to apply a method other than RETRIEVE to a Resource shall be rejected with an error response code.

### 7.6.3.8 Read-write Interface

The read-write Interface exposes only the Properties that may be “read” and “written”. The “read-only” Properties shall not be included in Representation for the “read-write” Interface. This is a generic Interface to support “reading” and “setting” Properties in a Resource. The applicable methods that can be applied to a Resource are RETRIEVE and UPDATE only. An attempt by a

1517 Client to apply a method other than RETRIEVE or UPDATE to a Resource shall be rejected with  
1518 an error response code.

## 1519 **7.7 Resource representation**

1520 Resource representation captures the state of a Resource at a particular time. The resource  
1521 representation is exchanged in the request and response interactions with a Resource. A Resource  
1522 representation may be used to retrieve or update the state of a resource.

1523 The resource representation shall not be manipulated by the data connectivity protocols and  
1524 technologies (e.g., CoAP, UDP/IP or BLE).

## 1525 **7.8 Structure**

### 1526 **7.8.1 Introduction**

1527 In many scenarios and contexts, the Resources may have either an implicit or explicit structure  
1528 between them. A structure can, for example, be a tree, a mesh, a fan-out or a fan-in. The  
1529 Framework provides the means to model and map these structures and the relationships among  
1530 Resources. The primary building block for resource structures in Framework is the collection. A  
1531 collection represents a container, which is extensible to model complex structures.

### 1532 **7.8.2 Resource Relationships**

1533 Resource relationships are expressed as Links. A Link embraces and extends typed web links  
1534 concept as a means of expressing relationships between Resources. A Link consists of a set of  
1535 Parameters that define:

- 1536 • a context URI,
- 1537 • a target URI,
- 1538 • a relation from the context URI to the target URI
- 1539 • elements that provide metadata about the target URI, the relationship or the context of the Link.

1540 The target URI is mandatory and the other items in a Link are optional. Additional items in the Link  
1541 may be made mandatory based on the use of the links in different contexts (e.g. in collections, in  
1542 discovery, in bridging etc.). Schema for the Link payload is provided in Annex D.

1543 An example of a Link is shown in

```
{ "href": "/switch", "rt": [ "oic.r.switch.binary" ], "if": [ "oic.if.a" , /room2"oic.if.baseline"], "p":  
{"bm": 3}, "rel": "item" }
```

1544 **Figure 10: Example of a Link**

1545 Two Links are distinct from each other when at least one parameter is different. For example the  
1546 two Links shown in Figure 11 are distinct and can appear in the same list of Links.

```
{ "href": "/switch", "rt": [ "oic.r.switch.binary" ], "if": [ "oic.if.a" , " oic.if.baseline"], "p": {"bm":  
2}, "rel": "item" }  
  
{ "href": "/switch", "rt": [ "oic.r.switch.binary" ], "if": [ "oic.if.a" , " oic.if.baseline"], "p": {"bm":  
2}}
```

1547 **Figure 11: Example of distinct Links**

1548 The specification may mandate Parameters and Parameter values as required for certain  
1549 capabilities. For all Links returned in a response to a RETRIEVE on "/oic/res", if a Link does not

1550 explicitly include the “rel” Parameter, a value of “rel”=“hosts” shall be assumed . The relation value  
1551 of “hosts” is defined by IETF RFC 6690, the value of “item” by IETF RFC 6573, and the value of  
1552 “self” by IETF RFC 4287 and all are registered in the IANA Registry for Link Relations at  
1553 [<http://www.iana.org/assignments/link-relations/link-relations.xhtml>]

1554 As shown in D.2.8 the relation between the context URI and target URI in a Link is specified using  
1555 the “rel” JSON element and the value of this element specifies the particular relation.

1556 The context URI of the Link shall implicitly be the URI of the Resource (or specifically a Collection)  
1557 that contains the Link unless the Link specifies the anchor parameter. The anchor parameter is  
1558 used to change the context URI of a Link – the relationship with the target URI is based off the  
1559 anchor URI when the anchor is specified. Anchor parameter uses transfer protocol URI for OIC 1.1  
1560 Link (e.g. “anchor”: “coaps://[fe80::b1d6]:44444”) and OCF URI defined in Sec 6 for OCF 1.0 Links  
1561 (e.g. “anchor”: “ocf://dc70373c-1e8d-4fb3-962e-017eaa863989”).

1562 An example of using anchors in the context of Collections – a floor has rooms and rooms have  
1563 lights – the lights may be defined in floor as Links but the Links will have the anchor set to the URI  
1564 of the rooms that contain the lights (the relation is contains). This allows all lights in a floor to be  
1565 turned on or off together while still having the lights defined with respect to the rooms that contain  
1566 them (lights may also be turned on by using the room URI too).

```
/a/floor {
  "links": [
    {
      "href": "/x/light1",
      "anchor": "/a/room1",    ** Note: /a/room1 has the “item” relationship with /x/light1;
not /a/floor **
      "rel": "item"
    }
  ]
}

/a/room1 {
  "links": [
    {
      ** Note: /a/room1 “contains” the /x/light since /a/room1 is the implicit context URI **
      "href": "/x/light1",
      "rel": "item"
    }
  ]
}
```

1567 **Figure 12: Example of use of anchor in Link**

## 1568 **7.8.2.1 Parameters**

### 1569 **7.8.2.1.1 “ins” or Link Instance Parameter**

1570 The “ins” parameter identifies a particular Link instance in a list of Links. The “ins” parameter may  
1571 be used to modify or delete a specific Link in a list of Links. The value of the “ins” parameter is set  
1572 at instantiation of the Link by the OCF Device (Server) that is hosting the list of Links – once it has  
1573 been set, the “ins” parameter shall not be modified for as long as the Link is a member of that list.

### 1574 **7.8.2.1.2 “p” or Policy Parameter**

1575 The Policy Parameter defines various rules for correctly accessing a Resource referenced by a  
1576 target URI. The Policy rules are configured by a set of key-value pairs as defined below.

1577 The policy Parameter “p” is defined by:

- 1578 • “bm” key: The “bm” key corresponds to an integer value that is interpreted as an 8-bit bitmask.  
 1579 Each bit in the bitmask corresponds to a specific Policy rule. The following rules are specified  
 1580 for “bm”:  
 1581

Bit Position	Policy rule	Comment
Bit 0 (the LSB)	discoverable	<p>The discoverable rule defines whether the Link is to be included in the Resource discovery message via “/oic/res”.</p> <ul style="list-style-type: none"> <li>• If the Link is to be included in the Resource discovery message, then “p” shall include the “bm” key and set the discoverable bit to value 1.</li> <li>• If the Link is NOT to be included in the Resource discovery message, then “p” shall either include the “bm” key and set the discoverable bit to value 0 or omit the “bm” key entirely.</li> </ul>
Bit 1 (2 <sup>nd</sup> LSB)	observable	<p>The observable rule defines whether the Resource referenced by the target URI supports the NOTIFY operation. With the self-link, i.e. the Link with “rel” value of “self”, “/oic/res” can have a Link with the target URI of “/oic/res” and indicate itself observable. The “self” is defined by IETF RFC 4287 and registered in the IANA Registry for “rel” value at [http://www.iana.org/assignments/link-relations/link-relations.xhtml].</p> <ul style="list-style-type: none"> <li>• If the Resource supports the NOTIFY operation, then “p” shall include the “bm” key and set the observable bit to value 1.</li> <li>• If the Resource does NOT support the NOTIFY operation, then “p” shall either include the “bm” key and set the observable bit to value 0 or omit the “bm” key entirely.</li> </ul>
Bits 2-7	--	Reserved for future use. All reserved bits in “bm” shall be set to value 0.

- 1582
- 1583 Note that if all the bits in “bm” are defined to value 0, then the “bm” key may be omitted entirely  
 1584 from “p” as an efficiency measure. However, if any bit is set to value 1, then “bm” shall be  
 1585 included in “p” and all the bits shall be defined appropriately.
- 1586 • “sec” and “port” in the remaining bullets shall be used only in an OIC 1.1 payload. In OCF 1.0  
 1587 payload, “sec” and “port” shall not be used and instead the “eps” Parameter shall provide the  
 1588 information for an encrypted connection.
- 1589 • “sec” key: The “sec” key corresponds to a Boolean value that indicates whether the Resource  
 1590 referenced by the target URI is accessed via an encrypted connection. If “sec” is true, the  
 1591 resource is accessed via an encrypted connection, using the “port” specified (see below). If  
 1592 “sec” is false, the resource is accessed via an unencrypted connection, or via an encrypted

connection (if such a connection is made using the “port” settings for another Resource, for which “sec” is true).

- “port” key: The “port” key corresponds to an integer value that is used to indicate the port number where the Resource referenced by the target URI may be accessed via an encrypted connection.
- If the Resource is only available via an encrypted connection (i.e. DTLS over IP), then
  - “p” shall include the “sec” key and its value shall be true.
  - “p” shall include the “port” key and its value shall be the port number where the encrypted connection may be established.
- If the Resource is not available via an encrypted connection, then
  - “p” shall include the “sec” key and its value shall be false or “p” shall omit the “sec” key; the default value of “sec” is false.
  - “p” shall omit the “port” key.
  - A Resource that is available via either an encrypted or unencrypted connection follows the population scheme defined in this clause.
- Access to the Resource on the port specified by the “port” key shall be made by an encrypted connection (e.g. coaps://). (Note that unencrypted connection to the Resource may be possible on a separate port discovered thru multicast discovery).
- Note that access to the Resource is controlled by the ACL for the Resource. A successful encrypted connection does not ensure that the requested action will succeed. See OCF Security – Access Control section for more information.

Example 1: below shows the Policy Parameter for a Resource that is discoverable but not observable, and for which authenticated accesses shall be done via CoAPS port 33275:

```
"p": { "bm": 1 }
```

Example 2: below shows a self-link, i.e. the “/oic/res” Link in itself that is discoverable and observable.

```
{
  "href": "/oic/res",
  "rel": "self",
  "rt": ["oic.wk.res"],
  "if": ["oic.if.ll", "oic.if.baseline"],
  "p": { "bm": 3 }
}
```

### 7.8.2.1.3 “type” or Media Type Parameter

The “type” Parameter may be used to specify the various media types that are supported by a specific target Resource. The default type of “application/cbor” shall be used when the “type” element is omitted. Once a Client discovers this information for each Resource, it may use one of the available representations in the appropriate header field of the Request or Response.

### 7.8.2.1.4 “bp” or the Batch Interface Parameter

The “batch” Parameter “bp” is used to specify the modifications to the target URI as the “batch” Request is forwarded through this Link. The “q” element in the value defines the query string that

1635 shall be appended to the "href" to make the target URI. The "q" query string may contain Property  
1636 strings that are valid in that context. For example: Given a Collection as follows

```
/room2
{
  "if": [ "oic.if.b" ],
  "colour": "blue",
  "links": [
    { "href": "/switch", "rt": [ "oic.r.switch.binary" ], "if": [ "oic.if.a", "oic.if.baseline" ], "p":
    { "bm": 2 }, "rel": "contains", "bp": { "q": "if=oic.if.baseline" } }
  ]
}
```

1637 The following is the sequence for batch request to /room2

1. GET /room2?if=oic.if.b
2. This request is transformed to: GET /switch?if=oic.if.baseline when the batch request is propagated through the Link to the target /switch

1638 See the Interfaces section 7.5 for more details on the "batch" Interface.

#### 1639 7.8.2.1.5 “di” or Device ID parameter

1640 The “di” Parameter specifies the device ID of the Device that hosts the target Resource defined in  
1641 the in the “href” Parameter.

1642 The device ID may be used to qualify a relative reference used in the “href” or to lookup endpoint  
1643 information for the relative reference.

#### 1644 7.8.2.1.6 “eps” Parmeter

1645 The "eps" Parameter indicates the Endpoint information of the target Resource.

1646 "eps" shall have as its value an array of items and each item represents Endpoint information with  
1647 "ep" and "pri" as specified in 10.2. "ep" is mandatory but "pri" is optional.

1648 Figure 13 is illustrated for "eps" with multiple Endpoints.

```
"eps": [
  { "ep": "coap://[fe80::b1d6]:1111", "pri": 2 },
  { "ep": "coaps://[fe80::b1d6]:1122" },
  { "ep": "coap+tcp://[2001:db8:a::123]:2222", "pri": 3 }
]
```

1649 **Figure 13: Example of “eps” Parameter**

1650 When "eps" is present in a link, the Endpoint information in "eps" can be used to access the target  
1651 Resource referred by the "href" Parameter.

1652 When present, max-age information (e.g. Max-Age option for CoAP defined in IETF RFC 7252)  
1653 determines the maximum time "eps" values may be cached before they are considered stale.



### 7.8.2.2 Formatting

When formatting in JSON, the list of Links shall be an array.

### 7.8.2.3 List of Links in a Collection

A list of Links in a Resource shall be included in that Resource as the value of the “links” Property of that Resource. A Resource that contains Links is a Collection.

A Resource with a list of Links

```
/Room1
{
  "rt": [ "my.room" ],
  "if": [ "oic.if.ll", "oic.if.baseline" ],
  "color": "blue",
  "links":
  [
    {
      "href": "/oic/d",
      "rt": [ "oic.d.light", "oic.wk.d" ],
      "if": [ "oic.if.r", "oic.if.baseline" ],
      "p": { "bm": 1 }
    },
    {
      "href": "/oic/p",
      "rt": [ "oic.wk.p" ],
      "if": [ "oic.if.r", "oic.if.baseline" ],
      "p": { "bm": 1 }
    },
    {
      "href": "/switch",
      "rt": [ "oic.r.switch.binary" ],
      "if": [ "oic.if.a", "oic.if.baseline" ],
      "p": { "bm": 3 },
      "mt": [ "application/cbor", "application/exi+xml" ]
    },
    {
      "href": "/brightness",
      "rt": [ "oic.r.light.brightness" ],
      "if": [ "oic.if.a", "oic.if.baseline" ],
      "p": { "bm": 3 }
    }
  ]
}
```

Figure 14: List of Links in a Resource

## 7.8.3 Collections

### 7.8.3.1 Overview

A Resource that contains one or more references (specified as Links) to other resources is an Collection. These reference may be related to each other or just be a list; the Collection provides a means to refer to this set of references with a single handle (i.e. the URI). A simple resource is kept distinct from a collection. Any Resource may be turned into an Collection by binding resource

1667 references as Links. Collections may be used for creating, defining or specifying hierarchies,  
1668 indexes, groups, and so on.

1669 A Collection shall have at least one Resource Type and at least one Interface bound at all times  
1670 during its lifetime. During creation time of a collection the Resource Type and interfaces are  
1671 specified. The initial defined Resource Types and interfaces may be updated during its life time.  
1672 These initial values may be overridden using mechanism used for overriding in the case of a  
1673 Resource. Additional Resource Types and Interfaces may be bound to the Collection at creation  
1674 or later during the lifecycle of the Collection.

1675 A Collection shall define the “links” Property. The value of the “links” Property is an array with zero  
1676 or more Links. The target URIs in the Links may reference another Collection or another Resource.  
1677 The referenced Collection or Resource may reside on the same Device as the Collection that  
1678 includes that Link (called a local reference) or may reside on another Device (called a remote  
1679 reference). The context URI of the Links in the “links” array shall (implicitly) be the Collection that  
1680 contains that “links” property. The (implicit) context URI may be overridden with explicit  
1681 specification of the “anchor” parameter in the Link where the value of “anchor” is the new base of  
1682 the Link.

1683 A Resource may be referenced in more than one Collection, therefore, a unique parent-child  
1684 relationship is not guaranteed. There is no pre-defined relationship between a Collection and the  
1685 Resource referenced in the Collection, i.e., the application may use Collections to represent a  
1686 relationship but none is automatically implied or defined. The lifecycles of the Collection and the  
1687 referenced Resource are also independent of one another.

1688 If the “drel” property is defined for the Collection then all Links that don’t explicitly specify a  
1689 relationship shall inherit this default relationship in the context of that Collection. The default  
1690 relationship defines the implicit relationship between the Collection and the target URI in the Link.

1691 A Property "links" represents the list of Links in a Collection. "links" Property has, as its value, an  
1692 array of items and each item is an OCF Link as shown in Figure 15.



**Figure 15: Example showing Collection and Links**

A Collection may be:

- A pre-defined Collection where the Collection has been defined a priori and the Collection is static over its lifetime. Such Collections may be used to model, for example, an appliance that is composed of other devices or fixed set of resource representing fixed functions.
- A Device local Collection where the Collection is used only on the Device that hosts the Collection. Such collections may be used as a short-hand on a client for referring to many Servers as one.
- A centralized Collection where the Collection is hosted on an Device but other Devices may access or update the Collection
- A hosted Collection where the collection is centralized but is managed by an authorized agent or party.

### 7.8.3.2 Collection Properties

An Collection shall define the “links” Property. In addition, other Properties may be defined for the Collection by the Resource Type. The mandatory and recommended Common Properties for Collection are shown in Table 9. This list of Common Properties are in addition to those defined for Resources in section 7.3.2. When a property is repeated in Table 9 , the conditions in this definition shall override those in the general list for Resources.

1713 **Table 9. Common Properties for Collections (in addition to Common Properties defined in**  
1714 **section 7.3.2)**

Property	Description	Property name	Value Type	Mandatory
<b>Links</b>	The set of links in the collection	"links"	json Array of Links	Yes
<b>Name</b>	Human friendly name for the collection	"n"	string	No
<b>Identity</b>	The id of the collection	"id"	UUID	No
<b>Resource Types</b>	The list of allowed Resource Types for links in the collection. Requests for addition of links using link list or link batch interfaces will be validated against this list.  If this property is not defined or is null string then any Resource Type is permitted	"rts"	json Array of Resource Type names	No
<b>Default relationship</b>	Specifies the default relationship to use for Links in the collection where the "rel" parameter has not been explicitly defined.  It is permissible to have no "drel" property defined for the collection and the Links to also not have "rel" defined either. In such case, the use of the collection is, for example, as a random bag of links	"drel"	string	No

1715

1716 The Properties of a Collection may not be modified.

1717 **7.8.3.3 Default Resource Type**

1718 A default Resource Type, "oic.wk.col", shall be available for Collections. This Resource Type shall  
1719 be used only when another type has not been defined on the Collection or when no Resource Type  
1720 has been specified at the creation of the Collection.

1721 The default Resource Type provides support for the Common Properties including the "links"  
1722 Property. For the default Resource Type, the value of "links" shall be a simple array of Links.

1723 The default Resource Type shall support the 'baseline' and 'links list' Interfaces. The default  
1724 Interface shall be the 'links list' Interface.

1725

1726 **7.9 Third (3<sup>rd</sup>) party specified extensions**

1727 This section describes how a 3<sup>rd</sup> party may add Device Types, Resource Types, 3<sup>rd</sup> party defined  
1728 Properties to an existing or 3<sup>rd</sup> party defined Resource Type, 3<sup>rd</sup> party defined enumeration values  
1729 to an existing enumeration and 3<sup>rd</sup> party defined parameters to an existing defined Property.

1730 A 3<sup>rd</sup> party may specify additional (non-OCF) Resources within an OCF Device. A 3<sup>rd</sup> party may  
1731 also specify additional Properties within an existing OCF defined Resource Type. Further a 3<sup>rd</sup>  
1732 party may extend an OCF defined enumeration with 3<sup>rd</sup> party defined values.

1733 A 3<sup>rd</sup> party defined Device Type may expose both 3<sup>rd</sup> party and OCF defined Resource Types. A  
1734 3<sup>rd</sup> party defined Device Type must expose the mandatory Resources for all OCF Devices defined  
1735 within this specification.

1736 A 3<sup>rd</sup> party defined Resource Type shall include any mandatory Properties defined in this  
1737 specification and also any vertical specified mandatory Properties. All Properties defined within a  
1738 3<sup>rd</sup> party defined Resource Type that are part of the OCF namespace that are not Common  
1739 Properties as defined in this specification shall follow the 3<sup>rd</sup> party defined Property rules in Table  
1740 10.

1741 The following table defines the syntax rules for 3<sup>rd</sup> party defined Resource Type elements. Within  
1742 the table the term “Domain\_Name” refers to a domain name that is owned by the 3<sup>rd</sup> party that is  
1743 defining the new element.

1744 **Table 10. 3rd party defined Resource elements**

	Resource Element	Vendor Definition Rules
New 3 <sup>rd</sup> party defined Device Type	“rt” Property Value of “/oic/d”	x.<Domain_Name>.<resource identification>
New 3 <sup>rd</sup> party defined Resource Type	“rt” Property Value	x.<Domain_Name>.<resource identification>
New 3 <sup>rd</sup> party defined Property within the OCF namespace	Resource Property Name	x.<Domain_Name>.<property>
Additional 3 <sup>rd</sup> party defined values in an OCF specified enumeration	Enumeration Property Value	x.<Domain_Name>.<enum value>
Additional 3 <sup>rd</sup> party defined parameter in an OCF specified Property	Parameter key word	x.<Domain_Name>.<parameter keyword>

1745  
1746 With respect to the use of the Domain\_Name in this scheme the labels are reversed from how they  
1747 appear in DNS or other resolution mechanisms. The 3<sup>rd</sup> party defined Device Type and Resource  
1748 Type otherwise follow the rules defined in Section 7.4.2 Resource Type Property. 3<sup>rd</sup> party defined  
1749 Resource Types should be registered in the IANA Constrained RESTful Environments (CoRE)  
1750 Parameters registry.

1751 For example:

1752 x.com.samsung.galaxyphone.accelerator

1753 x.com.cisco.ciscorouterport

1754 x.com.hp.printerhead

1755 x.org.allseen.newinterface.newproperty

1756

## 1757 **8 CRUDN**

### 1758 **8.1 Overview**

1759 CREATE, RETRIEVE, UPDATE, DELETE, and NOTIFY (CRUDN) are operations defined for  
1760 manipulating Resources. These operations are performed by a Client on the resources contained  
1761 in n Server.

1762 On reception of a valid CRUDN operation n Server hosting the Resource that is the target of the  
 1763 request shall generate a response depending on the Interface included in the request; or based  
 1764 on the Default Interface for the Resource Type if no Interface is included.

1765 CRUDN operations utilize a set of parameters that are carried in the messages and are defined in  
 1766 Table 11. A Device shall use CBOR as the default payload (content) encoding scheme for resource  
 1767 representations included in CRUDN operations and operation responses; a Device may negotiate  
 1768 a different payload encoding scheme (e.g, see in section 12.2.4 for CoAP messaging). The  
 1769 following subsections specify the CRUDN operations and use of the parameters. The type  
 1770 definitions for these terms will be mapped in the messaging section for each protocol.

1771

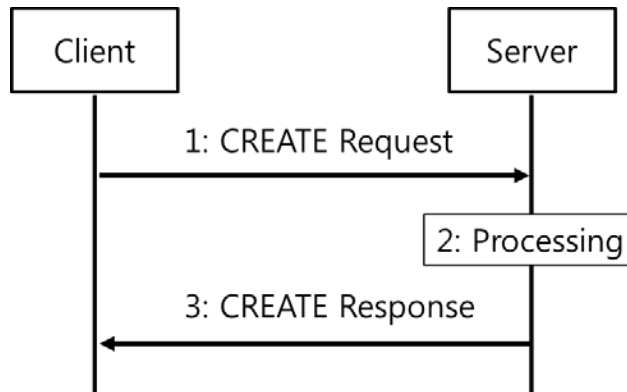
**Table 11. Parameters of CRUDN messages**

Applicability	Name	Denotation	Definition
All messages	<i>fr</i>	From	The URI of the message originator.
	<i>to</i>	To	The URI of the recipient of the message.
	<i>ri</i>	Request Identifier	The identifier that uniquely identifies the message in the originator and the recipient.
	<i>cn</i>	Content	Information specific to the operation.
Requests	<i>op</i>	Operation	Specific operation requested to be performed by the Server.
	<i>obs</i>	Observe	Indicator for an observe request.
Responses	<i>rs</i>	Response Code	Indicator of the result of the request; whether it was accepted and what the conclusion of the operation was. The values of the response code for CRUDN operations shall conform to those as defined in section 5.9 and 12.1.2 in IETF RFC 7252.
	<i>obs</i>	Observe	Indicator for an observe response.

1772

## 8.2 CREATE

1773 The CREATE operation is used to request the creation of new Resources on the Server. The  
 1774 CREATE operation is initiated by the Client and consists of three steps, as depicted in Figure 16  
 1775 and described below.



**Figure 16. CREATE operation**

### 8.2.1 CREATE request

The CREATE request message is transmitted by the Client to the Server to create a new Resource by the Server. The CREATE request message will carry the following parameters:

- *fr*: Unique identifier of the Client
- *to*: URI of the target resource responsible for creation of the new resource.
- *ri*: Identifier of the CREATE request
- *cn*: Information of the resource to be created by the Server
  - i) *cn* will include the URI and Resource Type property of the resource to be created.
  - ii) *cn* may include additional properties of the resource to be created.
- *op*: CREATE

### 8.2.2 Processing by the Server

Following the receipt of a CREATE request, the Server may validate if the Client has the appropriate rights for creating the requested resource. If the validation is successful, the Server creates the requested resource. The Server caches the value of *ri* parameter in the CREATE request for inclusion in the CREATE response message.

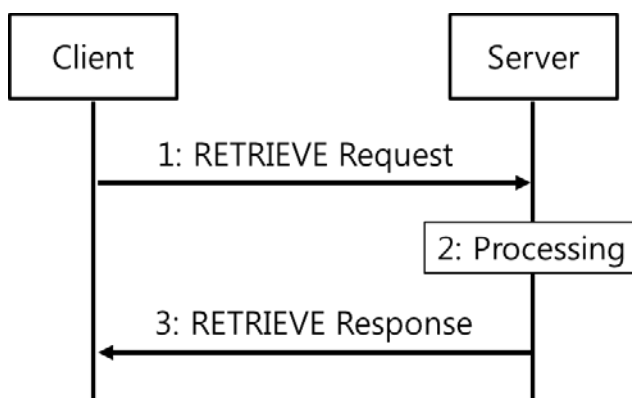
### 8.2.3 CREATE response

The Server shall transmit a CREATE response message in response to a CREATE request message from a Client. The CREATE response message will include the following parameters.

- *fr*: Unique identifier of the Server
- *to*: Unique identifier of the Client
- *ri*: Identifier included in the CREATE request
- *cn*: Information of the resource as created by the Server.
  - i) *cn* will include the URI of the created resource.
  - ii) *cn* will include the resource representation of the created resource.
- *rs*: The result of the CREATE operation

## 8.3 RETRIEVE

The RETRIEVE operation is used to request the current state or representation of a Resource. The RETRIEVE operation is initiated by the Client and consists of three steps, as depicted in Figure 17 and described below.



**Figure 17. RETRIEVE operation**

### 8.3.1 RETRIEVE request

RETRIEVE request message is transmitted by the Client to the Server to request the representation of a Resource from a Server. The RETRIEVE request message will carry the following parameters.

- *fr*: Unique identifier of the Client
- *to*: URI of the resource the Client is targeting
- *ri*: Identifier of the RETRIEVE request
- *op*: RETRIEVE

### 8.3.2 Processing by the Server

Following the receipt of a RETRIEVE request, the Server may validate if the Client has the appropriate rights for retrieving the requested data and the properties are readable. The Server caches the value of *ri* parameter in the RETRIEVE request for use in the response.

### 8.3.3 RETRIEVE response

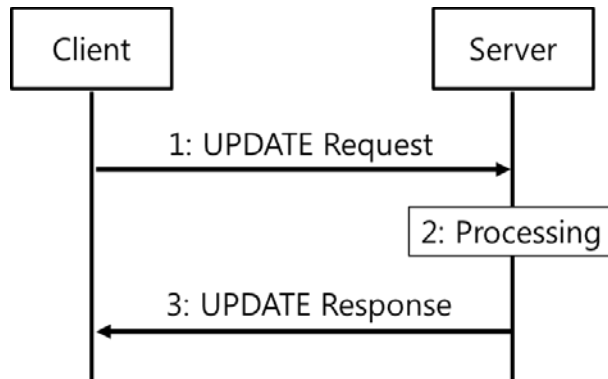
The Server shall transmit a RETRIEVE response message in response to a RETRIEVE request message from a Client. The RETRIEVE response message will include the following parameters.

- *fr*: Unique identifier of the Server
- *to*: Unique identifier of the Client
- *ri*: Identifier included in the RETRIEVE request
- *cn*: Information of the resource as requested by the Client
  - i) *cn* should include the URI of the resource targeted in the RETRIEVE request
- *rs*: The result of the RETRIEVE operation

## 8.4 UPDATE

The UPDATE operation is either a Partial UPDATE or a complete replacement of the information in a Resource in conjunction with the interface that is also applied to the operation. The UPDATE operation is initiated by the Client and consists of three steps, as depicted in Figure 18 and described below.





**Figure 18. UPDATE operation**

#### 8.4.1 UPDATE request

The UPDATE request message is transmitted by the Client to the Server to request the update of information of a Resource on the Server. The UPDATE request message will carry the following parameters.

- *fr*: Unique identifier of the Client
- *to*: URI of the resource targeted for the information update
- *ri*: Identifier of the UPDATE request
- *op*: UPDATE
- *cn*: Information, including properties, of the resource to be updated at the target resource

#### 8.4.2 Processing by the Server

Following the receipt of an UPDATE request, the Server may validate if the Client has the appropriate rights for updating the requested data. If the validation is successful the Server updates the target Resource information according to the information carried in *cn* parameter of the UPDATE request message. The Server caches the value of *ri* parameter in the UPDATE request for use in the response.

An UPDATE request that includes Properties that are read-only shall be rejected by the Server with an *rs* indicating a bad request.

An UPDATE request shall be applied only to the Properties in the target resource visible via the applied interface that support the operation. An UPDATE of non-existent Properties is ignored.

#### 8.4.3 UPDATE response

The UPDATE response message will include the following parameters:

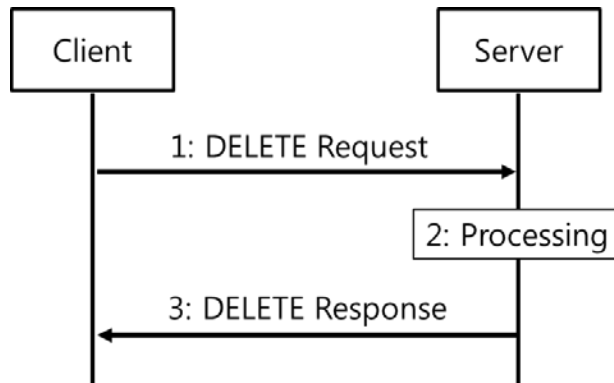
- *fr*: Unique identifier of the Server
- *to*: Unique identifier of the Client
- *ri*: Identifier included in the UPDATE request
- *rs*: The result of the UPDATE request

The UPDATE response message may also include the following parameters:

- *cn*: The Resource representation following processing of the UPDATE request

### 8.5 DELETE

The DELETE operation is used to request the removal of a Resource. The DELETE operation is initiated by the Client and consists of three steps, as depicted in Figure 19 and described below.



**Figure 19. DELETE operation**

### 8.5.1 DELETE request

DELETE request message is transmitted by the Client to the Server to delete a Resource on the Server. The DELETE request message will carry the following parameters:

- *fr*: Unique identifier of the Client
- *to*: URI of the target resource which is the target of deletion
- *ri*: Identifier of the DELETE request
- *op*: DELETE

### 8.5.2 Processing by the Server

Following the receipt of a DELETE request, the Server may validate if the Client has the appropriate rights for deleting the identified resource, and whether the identified resource exists. If the validation is successful, the Server removes the requested resource and deletes all the associated information. The Server caches the value of *ri* parameter in the DELETE request for use in the response.

### 8.5.3 DELETE response

The Server shall transmit a DELETE response message in response to a DELETE request message from a Client. The DELETE response message will include the following parameters.

- *fr*: Unique identifier of the Server
- *to*: Unique identifier of the Client
- *ri*: Identifier included in the DELETE request
- *rs*: The result of the DELETE operation

## 8.6 NOTIFY

The NOTIFY operation is used to request asynchronous notification of state changes. Complete description of the NOTIFY operation is provided in section 11.4. The NOTIFY operation uses the NOTIFICATION response message which is defined here.

### 8.6.1.1 NOTIFICATION response

The NOTIFICATION response message is sent by a Server to notify the URLs identified by the Client of a state change. The NOTIFICATION response message carries the following parameters.

- *fr*: Unique identifier of the Server
- *to*: URI of the Resource target of the NOTIFICATION message

- 1899       • *ri*: Identifier included in the CREATE request
- 1900       • *op*: NOTIFY
- 1901       • *cn*: The updated state of the resource

## 1902   **9   Network and connectivity**

### 1903   **9.1   Introduction**

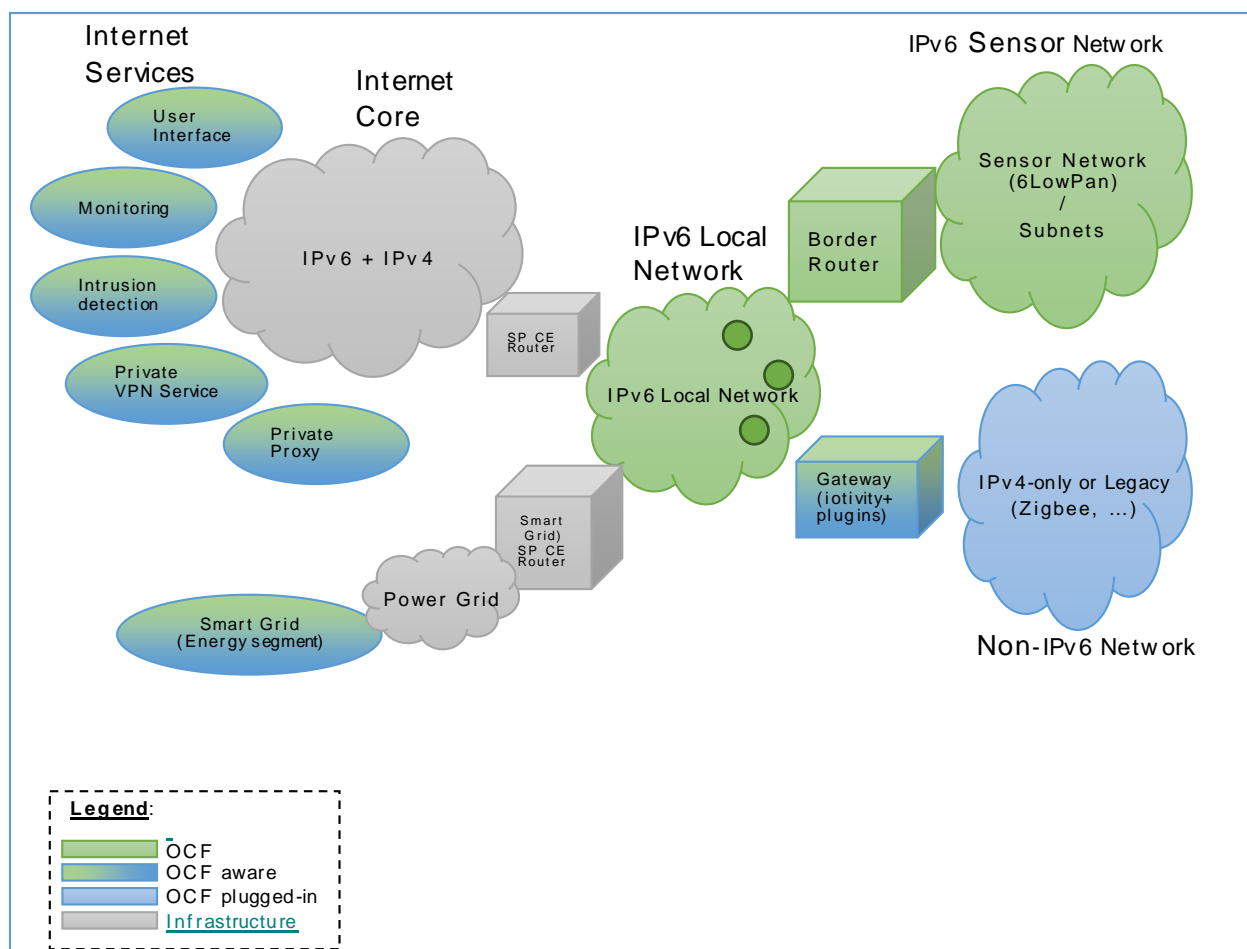
1904   The Internet of Things is comprised of a wide range of applications which sense and actuate the  
1905   physical world with a broad spectrum of device and network capabilities: from battery powered  
1906   nodes transmitting 100 bytes per day and able to last 10 years on a coin cell battery, to mains  
1907   powered nodes able to maintain MBit video streams. It is estimated that many 10s of billions of  
1908   IoT devices will be deployed over the coming years.

1909   It is desirable that the connectivity options be adapted to the IP layer. To that end, IETF has  
1910   completed considerable work to adapt Bluetooth®, Wi-Fi, 802.15.4, LPWAN, etc. to IPv6. These  
1911   adaptations, plus the larger address space and improved address management capabilities, make  
1912   IPv6 the clear choice for the OCF network layer technology.

### 1913   **9.2   Architecture**

1914   While the aging IPv4 centric network has evolved to support complex topologies, its deployment  
1915   was primarily provisioned by a single Internet Service Provider (ISP) as a single network. More  
1916   complex network topologies, often seen in residential home, are mostly introduced through the  
1917   acquisition of additional home network devices, which rely on technologies like private Network  
1918   Address Translation (NAT). These technologies require expert assistance to set up correctly and  
1919   should be avoided in a home network as they most often result in breakage of constructs like  
1920   routing, naming and discovery services.

1921   The multi-segment ecosystem OCF addresses will not only cause a proliferation of new devices  
1922   and associated routers, but also new services introducing additional edge routers. All these new  
1923   requirements require advance architectural constructs to address complex network topologies like  
1924   the one shown in Figure 20.



**Figure 20. High Level Network & Connectivity Architecture**

In terms of IETF RFC 6434, IPv6 nodes assume either a router or host role. Nodes may further implement various specializations of those roles:

- A Router may implement Customer Edge Router capabilities as defined in IETF RFC 7084.
- Nodes limited in processing power, memory, non-volatile storage or transmission capacity requires special IP adaptation layers (6LoWPAN) and/or dedicated routing protocols (RPL). Examples include devices transmitting over low power physical layer like IEEE 802.14.5, ITU G9959, Bluetooth Low Energy, DECT Ultra Low Energy, Near Field Communication (NFC).
- A node may translate and route messaging between IPv6 and non-IPv6 networks.

### 9.3 IPv6 network layer requirements

#### 9.3.1 Introduction

Projections indicate that many 10s of billions of new IoT endpoints and related services will be brought online in the next few years. These endpoint's capabilities will span from battery powered nodes with limited compute, storage, and bandwidth to more richly resourced devices operating over Ethernet and WiFi links.

Internet Protocol version 4 (IPv4), deployed some 30 years ago, has matured to support a wide variety of applications such as Web browsing, email, voice, video, and critical system monitoring and control. However, the capabilities of IPv4 are at the point of exhaustion, not the least of which is that available address space has been consumed.

1945 The IETF long ago saw the need for a successor to IPv4, thus the development of IPv6. OCF  
1946 recommends IPv6 at the network layer. Amongst the reasons for IPv6 recommendations are:

- 1947 • Larger address space. Side-effect: greatly reduce the need for NATs.
- 1948 • More flexible addressing architecture. Multiple addresses and types per interface: Link-local,  
1949 ULA, GUA, variously scoped Multicast addresses, etc. Better ability to support multi-homed  
1950 networks, better re-numbering capability, etc.
- 1951 • More capable auto configuration capabilities: DHCPv6, SLAAC, Router Discovery, etc.
- 1952 • Technologies enabling IP connectivity on constrained nodes are based upon IPv6.
- 1953 • All major consumer operating systems (iOS, Android, Windows, Linux) are already IPv6 enabled.
- 1954 • Major Service Providers around the globe are deploying IPv6.

### 1955 **9.3.2 IPv6 node requirements**

#### 1956 **9.3.2.1 Introduction**

1957 In order to ensure network layer services interoperability from node to node, mandating a common  
1958 network layer across all nodes is vital. The protocol should enable the network to be: secure,  
1959 manageable, scalable and to include constrained and self-organizing meshed nodes. OCF  
1960 mandates IPv6 as the common network layer protocol to ensure interoperability across all Devices.  
1961 More capable devices may also include additional protocols creating multiple-stack devices. The  
1962 remainder of this section will focus on interoperability requirements for IPv6 hosts, IPv6  
1963 constrained hosts and IPv6 routers. The various protocol translation permutations included in  
1964 multi-stack gateway devices may be addresses in subsequent addendums of this specification.

#### 1965 **9.3.2.2 IP Layer**

1966 An IPv6 node shall support IPv6 and it shall conform to the requirements as specified in  
1967 IETF RFC 6434:

1968

## 1969 **10 Endpoint**

### 1970 **10.1 Endpoint definition**

1971 The specific definition of an Endpoint depends on the Transport Protocol Suite being used. For the  
1972 example of CoAP over UDP over IPv6, the endpoint is identified by an IPv6 address and UDP port  
1973 number.

1974 Each OCF Device shall associate with at least one Endpoint with which it can exchange request  
1975 and response messages. When a message is sent to an Endpoint, it shall be delivered to the OCF  
1976 Device which is associated with the Endpoint. When a request message is delivered to an Endpoint,  
1977 path component is enough to locate the target Resource.

1978 OCF Device can be associated with multiple Endpoints. For example, an OCF Device can have  
1979 several IP addresses or port numbers or support both CoAP and HTTP transfer protocol.

1980 On the other hand, an Endpoint can be shared among multiple OCF Devices, only when there is a  
1981 way to clearly designate the target Resource with request URI. For example, when multiple CoAP  
1982 servers use uniquely different URI paths for all their hosted Resources, and the CoAP  
1983 implementation demuxes by path, they can share the same CoAP Endpoint. However, this is not  
1984 possible for OIC 1.1 and OCF 1.0 because pre-determined URI (e.g. "/oic/d") is mandatory for  
1985 some mandatory Resources (e.g. "oic.wk.d").

## 10.2 Endpoint information

### 10.2.1 Introduction

Endpoint is represented by Endpoint information which consists of two items of key-value pair, "ep" and "pri".

### 10.2.2 “ep”

"ep" represents Transport Protocol Suite and Endpoint Locator specified as follows:

- **Transport Protocol Suite** - a combination of protocols (e.g. CoAP + UDP + IPv6) with which request and response messages can be exchanged for RESTful transaction (i.e. CRUDN). Transport Protocol Suites shall be indicated by IANA registered schemes (e.g. "coap" or "coaps" in Table 12). Vendor or OCF defined schemes are also allowed (e.g. "org.ocf.foo" or "com.samsung.bar").
- **Endpoint Locator** – an address (e.g. IPv6 address + Port number) through which a message can be sent to the Endpoint and in turn associated OCF Device. The Endpoint Locator for "coap", "coaps", "coap+tcp", "coaps+tcp", "http", and "https" shall be specified as "IP address: port number". Temporary addresses should not be used because Endpoint Locators are for the purpose of accepting incoming sessions, whereas temporary addresses are for initiating outgoing sessions (IETF RFC 4941). Moreover its inclusion in "/oic/res" can cause a privacy concern (IETF RFC 7721).

"ep" shall have as its value a URI (as specified in IETF RFC 3986) with the scheme component indicating Transport Protocol Suite and the authority component indicating the Endpoint Locator. Figure 21 illustrate an example.

```
"ep": "coap://[fe80::b1d6]:1111"
```

**Figure 21: Example of "ep"**

The current list of "ep" with corresponding Transport Protocol Suite is shown in Table 12:

**Table 12. “ep” value for Transport Protocol Suite**

Transport Protocol Suite	scheme	Endpoint Locator	"ep" Value example
<b>coap + udp + ip</b>	coap	IP address + port number	coap://[fe80::b1d6]:1111
<b>coaps + udp + ip</b>	coaps	IP address + port number	coaps://[fe80::b1d6]:1122
<b>coap + tcp + ip</b>	coap+tcp	IP address + port number	coap+tcp://[2001:db8:a::123]:2222
<b>coaps + tcp + ip</b>	coaps+tcp	IP address + port number	coaps+tcp://[2001:db8:a::123]:2233
<b>http + tcp + ip</b>	http	IP address + port number	http://[2001:db8:a::123]:1111
<b>https + tcp + ip</b>	https	IP address + port number	https://[2001:db8:a::123]:1122

### 10.2.3 “pri”

When there are multiple Endpoints, "pri" indicates the priority among them.

"pri" shall be represented as a positive integer (e.g. "pri": 1) and the lower the value, the higher the priority.

The default "pri" value is 1, i.e. when "pri" is not present, it shall be equivalent to "pri": 1.

## 10.2.4 Endpoint information in "eps" Parameter

To carry Endpoint information, a new Link Parameter "eps" is defined in 7.8.2.1.6. "eps" has an array of items as its value and each item represents Endpoint information with two key-value pairs, "ep" and "pri", of which "ep" is mandatory and "pri" is optional. Figure 22 illustrates a link with "eps".

```
{
  "anchor": "ocf://light_device_id",
  "href": "/myLightSwitch",
  "rt": ["oic.r.switch.binary"],
  "if": ["oic.if.a", "oic.if.baseline"],
  "p": {"bm": 3},
  "eps": [{"ep": "coap://[fe80::b1d6]:1111", "pri": 2}, {"ep":
"coaps://[fe80::b1d6]:1122"}]
}
```

**Figure 22: Example of Link with "eps" Parameter**

In Figure 22, "anchor" represents the hosting OCF Device, "href", target Resource and "eps" the two Endpoints for the target Resource.

If the target Resource of a Link requires a secure connection (e.g. CoAPS), "eps" Parameter shall be used to indicate the necessary information (e.g. port number) in OCF 1.0 payload, because "sec" and "port" shall be used only in OIC 1.1 payload.

## 10.3 Endpoint discovery

### 10.3.1 Introduction

"Endpoint discovery" is defined as the process for a Client to acquire the Endpoint information for OCF Device or Resource.

### 10.3.2 Implicit discovery

If a Device is the source of a CoAP message (e.g. "/oic/res" response), the source IP address and port number can be combined to form the Endpoint Locator for the Device. Along with a "coap" scheme and default "pri" value, Endpoint information for the Device can be constructed.

In other words, an "/oic/res" response message with CoAP can implicitly carry the Endpoint information of the responding Device and in turn all the hosted Resources, which can be accessed with the same transfer protocol of CoAP.

### 10.3.3 Explicit discovery with "/oic/res" response

Endpoint information can be explicitly indicated with the "eps" Parameter of the Links in "/oic/res".

As in 10.3.2, an "/oic/res" response can implicitly indicate the Endpoint information for the target Resources hosted by the responding Device. However "/oic/res" may expose a target Resource which belongs to another Device. When the Endpoint for a target Resource of a Link cannot be implicitly inferred, the "eps" Parameter shall be included to provide explicit Endpoint information with which a Client can access the target Resource.

This applies to the case of "/oic/res" for a Resource Directory or Bridge Device which usually carries the Links for Resources which another Device hosts.

Figure 23 is a "/oic/res" response with the "eps" Parameter in Links.

```
[
{
  "anchor": "ocf://e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",
  "href": "/oic/res",
  "rel": "self",
  "rt": ["oic.wk.res"],
  "if": ["oic.if.ll", "oic.if.baseline"],
  "p": {"bm": 3},
  "eps": [{"ep": "coap://[2001:db8:a::b1d4]:55555"},
           {"ep": "coaps://[2001:db8:a::b1d4]:11111"}]
},
{
  "anchor": "ocf://e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",
  "href": "/oic/d",
  "rt": ["oic.wk.d", "oic.d.bridge"],
  "if": ["oic.if.r", "oic.if.baseline"],
  "p": {"bm": 3},
  "eps": [{"ep": "coap://[2001:db8:a::b1d4]:55555"},
           {"ep": "coaps://[2001:db8:a::b1d4]:11111"}]
},
{
  "anchor": "ocf://e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",
  "href": "/oic/p",
  "rt": ["oic.wk.p"],
  "if": ["oic.if.r", "oic.if.baseline"],
  "p": {"bm": 3},
  "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:11111"}]
},
{
  "anchor": "ocf://e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",
  "href": "/mySecureMode",
  "rt": ["oic.r.securemode"],
  "if": ["oic.if.rw", "oic.if.baseline"],
  "p": {"bm": 3},
  "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:11111"}]
},
{
  "anchor": "ocf://e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",
  "href": "/oic/sec/doxm",
  "rt": ["oic.r.doxm"],
  "if": ["oic.if.baseline"],
  "p": {"bm": 1},
  "eps": [{"ep": "coap://[2001:db8:a::b1d4]:55555"},
           {"ep": "coaps://[2001:db8:a::b1d4]:11111"}]
},
{
  "anchor": "ocf://e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",
  "href": "/oic/sec/pstat",
  "rt": ["oic.r.pstat"],
  "if": ["oic.if.baseline"],
  "p": {"bm": 1},
  "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:11111"}]
},
{
  "anchor": "ocf://e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",
  "href": "/oic/sec/cred",
  "rt": ["oic.r.cred"],
  "if": ["oic.if.baseline"],
  "p": {"bm": 1},
  "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:11111"}]
}
]
```



```

},
{
  "anchor": "ocf://e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",
  "href": "/oic/sec/acl2",
  "rt": ["oic.r.acl2"],
  "if": ["oic.if.baseline"],
  "p": {"bm": 1},
  "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:11111"}]
},
{
  "anchor": "ocf://e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",
  "href": "/myIntrospection",
  "rt": ["oic.wk.introspection"],
  "if": ["oic.if.r", "oic.if.baseline"],
  "p": {"bm": 3},
  "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:11111"}]
},
{
  "anchor": "ocf://dc70373c-1e8d-4fb3-962e-017eaa863989",
  "href": "/oic/res",
  "rt": ["oic.wk.res"],
  "if": ["oic.if.ll", "oic.if.baseline"],
  "p": {"bm": 3},
  "eps": [{"ep": "coap://[2001:db8:a::b1d4]:66666"},
          {"ep": "coaps://[2001:db8:a::b1d4]:22222"}]
},
{
  "anchor": "ocf://dc70373c-1e8d-4fb3-962e-017eaa863989",
  "href": "/oic/d",
  "rt": ["oic.wk.d", "oic.d.light", "oic.d.virtual"],
  "if": ["oic.if.r", "oic.if.baseline"],
  "p": {"bm": 3},
  "eps": [{"ep": "coap://[2001:db8:a::b1d4]:66666"},
          {"ep": "coaps://[2001:db8:a::b1d4]:22222"}]
},
{
  "anchor": "ocf://dc70373c-1e8d-4fb3-962e-017eaa863989",
  "href": "/oic/p",
  "rt": ["oic.wk.p"],
  "if": ["oic.if.r", "oic.if.baseline"],
  "p": {"bm": 3},
  "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:22222"}]
},
{
  "anchor": "ocf://dc70373c-1e8d-4fb3-962e-017eaa863989",
  "href": "/myLight",
  "rt": ["oic.r.switch.binary"],
  "if": ["oic.if.a", "oic.if.baseline"],
  "p": {"bm": 3},
  "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:22222"}]
},
{
  "anchor": "ocf://dc70373c-1e8d-4fb3-962e-017eaa863989",
  "href": "/oic/sec/doxm",
  "rt": ["oic.r.doxm"],
  "if": ["oic.if.baseline"],
  "p": {"bm": 1},
  "eps": [{"ep": "coap://[2001:db8:a::b1d4]:66666"},
          {"ep": "coaps://[2001:db8:a::b1d4]:22222"}]
},
{
  "anchor": "ocf://dc70373c-1e8d-4fb3-962e-017eaa863989",

```

```

    "href": "/oic/sec/pstat",
    "rt": ["oic.r.pstat"],
    "if": ["oic.if.baseline"],
    "p": {"bm": 1},
    "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:22222"}]
  }, {
    "anchor": "ocf://dc70373c-1e8d-4fb3-962e-017eaa863989",
    "href": "/oic/sec/cred",
    "rt": ["oic.r.cred"],
    "if": ["oic.if.baseline"],
    "p": {"bm": 1},
    "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:22222"}]
  },
  {
    "anchor": "ocf://dc70373c-1e8d-4fb3-962e-017eaa863989",
    "href": "/oic/sec/acl2",
    "rt": ["oic.r.acl2"],
    "if": ["oic.if.baseline"],
    "p": {"bm": 1},
    "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:22222"}]
  },
  {
    "anchor": "ocf://dc70373c-1e8d-4fb3-962e-017eaa863989",
    "href": "/myLightIntrospection",
    "rt": ["oic.wk.introspection"],
    "if": ["oic.if.r", "oic.if.baseline"],
    "p": {"bm": 3},
    "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:22222"}]
  },
  {
    "anchor": "ocf://88b7c7f0-4b51-4e0a-9faa-cfb439fd7f49",
    "href": "/oic/res",
    "rt": ["oic.wk.res"],
    "if": ["oic.if.ll", "oic.if.baseline"],
    "p": {"bm": 3},
    "eps": [{"ep": "coap://[2001:db8:a::b1d4]:77777"},
            {"ep": "coaps://[2001:db8:a::b1d4]:33333"}]
  },
  {
    "anchor": "ocf://88b7c7f0-4b51-4e0a-9faa-cfb439fd7f49",
    "href": "/oic/d",
    "rt": ["oic.wk.d", "oic.d.fan", "oic.d.virtual"],
    "if": ["oic.if.r", "oic.if.baseline"],
    "p": {"bm": 3},
    "eps": [{"ep": "coap://[2001:db8:a::b1d4]:77777"},
            {"ep": "coaps://[2001:db8:a::b1d4]:33333"}]
  },
  {
    "anchor": "ocf://88b7c7f0-4b51-4e0a-9faa-cfb439fd7f49",
    "href": "/oic/p",
    "rt": ["oic.wk.p"],
    "if": ["oic.if.r", "oic.if.baseline"],
    "p": {"bm": 3},
    "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:33333"}]
  },
  {
    "anchor": "ocf://88b7c7f0-4b51-4e0a-9faa-cfb439fd7f49",
    "href": "/myFan",
    "rt": ["oic.r.switch.binary"],
    "if": ["oic.if.a", "oic.if.baseline"],
    "p": {"bm": 3},
    "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:33333"}]
  }

```

```

    },
    {
      "anchor": "ocf://88b7c7f0-4b51-4e0a-9faa-cfb439fd7f49",
      "href": "/oic/sec/doxm",
      "rt": ["oic.r.doxm"],
      "if": ["oic.if.baseline"],
      "p": {"bm": 1},
      "eps": [{"ep": "coap://[2001:db8:a::b1d4]:7777"},
        {"ep": "coaps://[2001:db8:a::b1d4]:3333"}]
    },
    {
      "anchor": "ocf://88b7c7f0-4b51-4e0a-9faa-cfb439fd7f49",
      "href": "/oic/sec/pstat",
      "rt": ["oic.r.pstat"],
      "if": ["oic.if.baseline"],
      "p": {"bm": 1},
      "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:3333"}]
    },
    {
      "anchor": "ocf://88b7c7f0-4b51-4e0a-9faa-cfb439fd7f49",
      "href": "/oic/sec/cred",
      "rt": ["oic.r.cred"],
      "if": ["oic.if.baseline"],
      "p": {"bm": 1},
      "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:3333"}]
    },
    {
      "anchor": "ocf://88b7c7f0-4b51-4e0a-9faa-cfb439fd7f49",
      "href": "/oic/sec/acl2",
      "rt": ["oic.r.acl2"],
      "if": ["oic.if.baseline"],
      "p": {"bm": 1},
      "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:3333"}]
    },
    {
      "anchor": "ocf://88b7c7f0-4b51-4e0a-9faa-cfb439fd7f49",
      "href": "/myFanIntrospection",
      "rt": ["oic.wk.introspection"],
      "if": ["oic.if.r", "oic.if.baseline"],
      "p": {"bm": 3},
      "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:3333"}]
    }
  ]

```

**Figure 23: Example of “/oic/res” with Endpoint information**

The exact format of the “/oic/res” response and a way for a Client to acquire a “/oic/res” response message is specified in D.10 and 11.3.5 respectively.

#### 10.4 CoAP based Endpoint discovery

The following describes CoAP based Endpoint discovery:

- a) Advertising or publishing Devices shall join the ‘All OCF Nodes’ multicast groups (as defined in [IANA IPv6 Multicast Address Space Registry]) with scopes 2, 3, and 5 (i.e., ff02::158, ff03::158 and ff05::158) and shall listen on the port 5683. For compliance to IETF RFC 7252 a Device may additionally join the ‘All CoAP Nodes’ multicast groups.
- b) Clients intending to discover resources shall join the multicast groups as defined in a).
- c) Clients shall send discovery requests (GET request) to the ‘All OCF Nodes’ multicast group address with scope 2 (ff02::158) at port 5683. The requested URI shall be “/oic/res”. For

2062 compliance to IETF RFC 7252 a Client may additionally send to the 'All CoAP Nodes' multicast  
 2063 groups.

2064 d) If the discovery request is intended for a specific Resource Type, the Query parameter "rt" shall  
 2065 be included in the request (section 6.2.1) with its value set to the desired Resource Type. Only  
 2066 Devices hosting the Resource Type shall respond to the discovery request.

2067 e) When the "rt" Query parameter is omitted, all Devices shall respond to the discovery request.

2068 f) Handling of multicast requests shall be as described in section 8 of IETF RFC 7252 and section  
 2069 4.1 in IETF RFC 6690.

2070 g) Devices which receive the request shall respond using CBOR payload encoding. A Device shall  
 2071 indicate support for CBOR payload encoding for multicast discovery as described in section  
 2072 12.4. Later versions of the specification may support alternate payload encodings (JSON,  
 2073 XML/EXI, etc.).

## 2074 11 Functional interactions

### 2075 11.1 Introduction

2076 The functional interactions between a Client and n Server are described in section 11.2 through  
 2077 section 11.6 respectively. The functional interactions use CRUDN messages (section 8) and  
 2078 include Discovery, Notification, and Device management. These functions require support of core  
 2079 defined resources as defined in Table 13. More details about these resources are provided later  
 2080 in this section.

2081 **Table 13. List of Core Resources**

Pre-defined URI	Resource Name	Resource Type	Related Functional Interaction	Mandatory
<b>"/oic/res"</b>	Default	"oic.wk.res"	Discovery	Yes
<b>"/oic/p"</b>	Platform	oic.wk.p	Discovery	Yes
<b>/oic/d</b>	Device	"oic.wk.d"	Discovery	Yes
<b>(none)</b>	Configuration	oic.wk.con	Device Management	No
<b>"/oic/mnt"</b>	Maintenance	"oic.wk.mnt"	Device Management	No

2082

### 2083 11.2 Onboarding, Provisioning and Configuration

2084 Onboarding and Provisioning are fully defined by the OCF Security Specification.

2085

2086 Should a Device support Client update of configurable information it shall do so via exposing the  
 2087 Core Resource **"/example/oic/con"** (Table 14) in **"/oic/res"**;

2088

2089 **Table 14. Configuration Resource**

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

				The <b>resource properties</b> exposed in "oic.wk.con" are listed in Table 15.	
<b>"/example/oic/con"</b>	Platform Configuration	"oic.wk.con.p"	"oic.if.rw"	The optional Resource Type through which configurable information specific to the Platform is exposed. The resource properties exposed in "oic.wk.con.p" are listed in Table 16.	Configuration

2090

2091 Table 15 defines the "oic.wk.con" resource type.

2092

2093

**Table 15. "oic.wk.con" Resource Type definition**

Property title	Property name	Value type	Value rule	Unit	Access mode	Mandatory	Description
<b>(Device) Name</b>	n (Common Property of "/example/oic/con")	string			R, W	yes	Human friendly name configurable by the end user (e.g. Bob's thermostat). "n" Common Property of "/example/oic/con" and "n" Common Property "/oic/d" shall have the same Value. When the "n" Common Property Value of "/example/oic/con" is modified, it shall be reflected to the "n" Common Property of "/oic/d".
<b>Location</b>	loc	array of float (has two elements, the first is latitude, the second is longitude)		Degrees	R, W	no	Provides location information where available.
<b>Location Name</b>	locn	string			R, W	no	Human friendly name for location For example, "Living Room".
<b>Currency</b>	c	string			R, W	no	Indicates the currency that is used for any monetary transactions
<b>Region</b>	r	string			R, W	no	Free form text Indicating the current region in which the device is located geographically. The free form text shall not start with a quote (").
<b>Localized Names</b>	ln	array			R, W	no	Human-friendly name of the Device, in one or more languages. This property is an array of objects where each object has a 'language' field (containing an IETF RFC 5646 language tag) and a 'value' field

							containing the device name in the indicated language. If this property and the Device Name (n) property are both supported, the Device Name (n) value shall be included in this array.
<b>Default Language</b>	dl	language-tag			R,W	no	The default language supported by the Device, specified as an IETF RFC 5646 language tag. By default, clients can treat any string property as being in this language unless the property specifies otherwise.

2094

2095 Table 16 defines the “oic.wk.con.p” resource type.

2096 **Table 16. “oic.wk.con.p” Resource Type definition**

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

2097

2098

## 2099 11.3 Resource discovery

### 2100 11.3.1 Introduction

2101 Discovery is a function which enables endpoint discovery as well as resource based discovery.  
2102 Endpoint discovery is described in detail in section 10. This section mainly describes the resource  
2103 based discovery.

### 2104 11.3.2 Resource based discovery: mechanisms

#### 2105 11.3.2.1 Overview

2106 As part of discovery, a Client may find appropriate information about other OCF peers. This  
2107 information could be instances of Resources, Resource Types or any other information  
2108 represented in the resource model that an OCF peer would want another OCF peer to discover.

2109 At the minimum, Resource based discovery uses the following:

- 2110 1) A resource to enable discovery shall be defined. The representation of that resource shall  
2111 contain the information that can be discovered.
- 2112 2) The resource to enable discovery shall be specified and commonly known a-priori. A Device for  
2113 hosting the resource to enable discovery shall be identified.
- 2114 3) A mechanism and process to publish the information that needs to be discovered with the  
2115 resource to enable discovery.
- 2116 4) A mechanism and process to access and obtain the information from the resource to enable  
2117 discovery. A query may be used in the request to limit the returned information.
- 2118 5) A scope for the publication
- 2119 6) A scope for the access.
- 2120 7) A policy for visibility of the information.
- 2121

2122 Depending on the choice of the base aspects defined above, the Framework defines three resource  
2123 based discovery mechanisms:

- 2124 • Direct discovery, where the Resources are published locally at the Device hosting the  
2125 resources and are discovered through peer inquiry.
- 2126 • Indirect discovery, where Resources are published at a third party assisting with the  
2127 discovery and peers publish and perform discovery against the resource to enable  
2128 discovery on the assisting 3<sup>rd</sup> party.
- 2129 • Advertisement discovery, where the resource to enable discovery is hosted local to the  
2130 initiator of the discovery inquiry but remote to the Devices that are publishing discovery  
2131 information.

2132 A Device shall support direct discovery.

### 2133 **11.3.2.2 Direct discovery**

2134 In direct discovery,

- 2135 1) The Device that is providing the information shall host the resource to enable discovery.
  - 2136 2) The Device publishes the information available for discovery with the local resource to  
2137 enable discovery (i.e. local scope).
  - 2138 3) Clients interested in discovering information about this Device shall issue RETRIEVE  
2139 requests directly to the resource. The request may be made as a unicast or multicast.  
2140 The request may be generic or may be qualified or limited by using appropriate queries in  
2141 the request.
  - 2142 4) The “server” Device that receives the request shall send a response with the discovered  
2143 information directly back to the requesting “client” Device.
  - 2144 5) The information that is included in the request is determined by the policies set for the  
2145 resource to be discovered locally on the responding Device.
- 2146

### 2147 **11.3.2.3 Indirect discovery of Resources (resource directory based discovery)**

2148 In indirect discovery the information about the resource to be discovered is hosted on a Server  
2149 that is not hosting the resource. See section 11.3.6 for details on resource directory based  
2150 discovery.

2151 In indirect discovery:

- 2152 a) The resource to be discovered is hosted on a Device that is neither the client initiating  
2153 the discovery nor the Device that is providing or publishing the information to be

- 2154 discovered. This Device may use the same resource to provide discovery for multiple  
2155 agents looking to discover and for multiple agents with information to be discovered.  
2156 b) The Device to be discovered or with information to discover, publishes that information  
2157 with resource to be discovered on a different Device. The policies on the information  
2158 shared including the lifetime/validity are specified by the publishing Device. The  
2159 publishing Device may modify these policies as required.  
2160 c) The client doing the discovery may send a unicast discovery request to the Device  
2161 hosting the discovery information or send a multicast request that shall be monitored and  
2162 responded to by the Device. In both cases, the Device hosting the discovery information  
2163 is acting on behalf of the publishing Device.  
2164 d) The discovery policies may be set by the Device hosting the discovery information or by  
2165 the party that is publishing the information to be discovered. The discovery information  
2166 that is returned in the discovery response shall adhere to the policies that are in effect at  
2167 the time of the request.  
2168

#### 2169 **11.3.2.4 Advertisement Discovery**

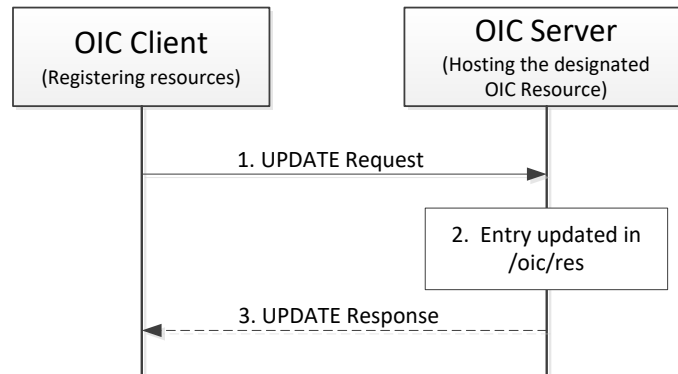
2170 In advertisement discovery:

- 2171 a) The resource to enable discovery is hosted local to the Device that is initiating the discovery  
2172 request (client). The resource to enable discovery may be an Core Resource or discovered  
2173 as part of a bootstrap.  
2174 b) The request could be an implementation dependent lookup or be a local RETRIEVE request  
2175 against the resource that enables discovery.  
2176 c) The Device with information to be discovered shall publish the appropriate information to  
2177 the resource that enables discovery.  
2178 d) The publishing Device is responsible for the published information. The publishing Device  
2179 may UPDATE the information at the resource to enable discovery based on its needs by  
2180 sending additional publication requests. The policies on the information that is discovered  
2181 including lifetime is determined by the publishing Device.  
2182

#### 2183 **11.3.3 Resource based discovery: Information publication process**

2184 The mechanism to publish information with the resource to enable discovery can be done either  
2185 locally or remotely. The publication process is depicted in Figure 24. The Device which has  
2186 discovery information to publish shall a) either update the resource that enables discovery if  
2187 hosted locally or b) issue an UPDATE request with the information to the Device which hosts the  
2188 resource that enables discovery. The Device hosting the resource to enable discovery  
2189 adds/updates the resource to enable discovery with the provided information and then responds  
2190 to the Device which has requested the publication of the resource with an UPDATE response.  
2191

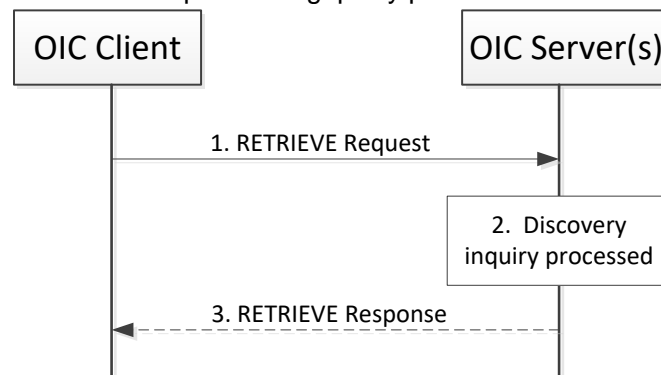




**Figure 24. Resource based discovery: Information publication process**

### 11.3.4 Resource based discovery: Finding information

The discovery process (Figure 25) is initiated as a RETRIEVE request to the resource to enable discovery. The request may be sent to a single Device (as in a Unicast) or to multiple Devices (as in Multicast). The specific mechanisms used to do Unicast or Multicast are determined by the support in the data connectivity layer. The response to the request has the information to be discovered based on the policies for that information. The policies can determine which information is shared, when and to which requesting agent. The information that can be discovered can be resources, types, configuration and many other standards or custom aspects depending on the request to appropriate resource and the form of request. Optionally the requester may narrow the information to be returned in the request using query parameters in the URI query.



**Figure 25. Resource based discovery: Finding information**

### Discovery Resources

Some of the Core Resources shall be implemented on all Devices to support discovery. The Core Resources that shall be implemented to support discovery are:

- “/oic/res” for discovery of resources
- “/oic/p” for discovery of platform
- “/oic/d” for discovery of device information

Details for these mandatory Core Resources are described in Table 17

Platform resource –

The OCF recognizes that more than one instance of Device may be hosted on a single platform. Clients need a way to discover and access the information on the platform. The core resource, “/oic/p” exposes platform specific properties. All instances of Device on the same Platform shall have the same values of any properties exposed (i.e. a Device may choose to expose optional properties within “/oic/p” but when exposed the value of that property should be the same as the value of that property on all other Devices on that Platform)

#### Device resource

The device resource shall have the pre-defined URI “/oic/d”. The resource “/oic/d” exposes the properties pertaining to a Device as defined in Table 17. The properties exposed are determined by the specific instance of Device and defined by the Resource Type(s) of “/oic/d” on that Device. Since all the Resource Types of “/oic/d” are not known a priori, the Resource Type(s) of “/oic/d” shall be determined by discovery through the core resource “/oic/res”. The device resource “/oic/d” shall have a default Resource Type that helps in bootstrapping the interactions with this device (the default type is described in Table 17.)

#### Protocol indication

A Device may need to support different messaging protocols depending on requirements for different application profiles. For example, the Smart Home profile may use CoAP and the Industrial profile may use DDS. To enable interoperability, a Device uses the protocol indication to indicate the transport protocols they support and can communicate over.

**Table 17. Mandatory discovery Core Resources**

Pre-define d URI	Resource Type Title	Resource Type ID ("rt" value)	Interfaces	Description	Related Functional Interaction
“/oic/res”	Default	“oic.wk.res”	“oic.if.ll”	The resource through which the corresponding Server is discovered and introspected for available resources.  “/oic/res” shall expose the resources that are discoverable on a Device. When a Server receives a RETRIEVE request targeting “/oic/res” (e.g., “GET /oic/res”), it shall respond with the link list of all the discoverable resources of itself. The “/oic/d” and “/oic/p” are discoverable resources, hence their links are included in “/oic/res” response. The resource properties exposed by “/oic/res” are listed in Table 18.	Discovery
“/oic/p”	Platform	“oic.wk.p”	“oic.if.r”	The discoverable resource through which platform specific information is discovered.  The <b>resource properties</b> exposed by “/oic/p” are listed in Table 21	Discovery
“/oic/d”	Device	“oic.wk.d” and/or one Device Specific Resource Type ID	“oic.if.r”	The discoverable via “/oic/res” resource which exposes properties specific to the Device instance. The <b>resource properties</b> exposed by “/oic/d” are listed in Table 20  “/oic/d” may have one Resource Type that is specific to the Device in addition to the default Resource Type or if present overriding the default Resource Type.  The base type “oic.wk.d” defines the properties that shall be exposed by all Devices.  The device specific Resource Type exposed is dependent on the class of device (e.g. air conditioner, smoke alarm); applicable values are defined by the vertical specifications.	Discovery

2238

2239 Table 18 defines “oic.wk.res” Resource Type.

2240 **Table 18. “oic.wk.res” Resource Type definition**

Property title	Property name	Value type	Value rule	Unit	Access mode	Mandatory	Description
<b>Name</b>	n	string			R	no	Human-friendly name defined by the vendor
<b>Links</b>	links	array	See 7.8.2		R	yes	The array of Links describes the URI, supported Resource Types and interfaces, and access policy.
<b>Messaging Protocol</b>	mpro	SSV			R	No	String with Space Separated Values (SSV) of messaging protocols supported as a SI Number from Table 19 For example, “1 and 3” indicates that the Device supports coap and http as messaging protocols.

2241 A Device which wants to indicate its messaging protocol capabilities may add the property ‘mpro’  
2242 in response to a request on “/oic/res”. A Device shall support CoAP based discovery as the  
2243 baseline discovery mechanism (see section 10.4). A Client which sees this property in a discovery  
2244 response can choose any of the supported messaging protocols for communicating with the Server  
2245 for further messages. For example, if a Device supporting multiple protocols indicates it supports  
2246 a value of “1 3” for the ‘mpro’ property in the discovery response, then it cannot be assumed that  
2247 there is an implied ordering or priority. But a vertical service specification may choose to specify  
2248 an implied ordering or priority. If the ‘mpro’ property is not present in the response, A Client shall  
2249 use the default messaging protocol as specified in the vertical specification for further  
2250 communication.

2251 The “/oic/res” shall list all Resources that are indicated as discoverable (see section 11.3). Also  
2252 the following architecture Resource Types shall be listed:

- 2253 • Introspection resource indicated with an “rt” value of “oic.wk.introspection”  
2254 • “/oic/p” indicated with an “rt” value of “oic.wk.p”  
2255 • “/oic/d” indicated with an “rt” value of “oic.wk.d”  
2256 • “/oic/sec/doxm” indicated with an “rt” value of “oic.r.doxm”  
2257 • “/oic/sec/pstat” indicated with an “rt” value of “oic.r.pstat”

2258 Conditionally required:

- 2259 • “/oic/res” with an “rt” value of “oic.wk.res” as self-reference, on the condition that “oic/res” has  
2260 to signal that it is observable by a Client.

2261 The Introspection Resource is only applicable for Devices that host Vertical Resource Types (e.g.  
2262 “oic.r.switch.binary”) or vendor-defined Resource Types. Devices that only host Resources  
2263 required to onboard the Device as a Client do not have to implement the Introspection Resource.

2264 Table 19 provides an OCF registry for protocol schemes.

2265 **Table 19. Protocol scheme registry**

SI Number	Protocol
1	coap

2	coaps
3	http
4	https
5	coap+tcp
6	coaps+tcp

Note: The discovery of an endpoint used by a specific protocol is out of scope. The mechanism used by a Client to form requests in a different messaging protocol other than discovery is out of scope.

The following applies to the use of “/oic/d” as defined above:

- A vertical may choose to expose its Device Type (e.g., refrigerator or A/C) by adding the Device Type to the list of Resource Types associated with “/oic/d”.
  - For example; “rt” of “/oic/d” becomes ["oic.wk.d", "oic.d.<thing>"]; where “oic.d.<thing>” is defined in another spec such as the Smart Home vertical.
  - This implies that the properties exposed by “/oic/d” are by default the mandatory properties in Table 20.
- A vertical may choose to extend the list of properties defined by the Resource Type 'oic.wk.d'. In that case, the vertical shall assign a new Device Type specific Resource Type ID. The mandatory properties defined in Table 20 shall always be present.

Table 20 “oic.wk.d” Resource Type definition defines the base Resource Type for the “/oic/d” resource.

**Table 20. “oic.wk.d” Resource Type definition**

Property title	Property name	Value type	Value rule	Unit	Access mode	Mandatory	Description
<b>(Device) Name</b>	n	string			R	no	Human friendly name defined by the vendor.” In the presence of "n" Property of “/oic/con”, both have the same Property Value. When "n" Property Value of “/oic/con” is modified, it shall be reflected to "n" Property Value of “/oic/d”.
<b>Spec Version</b>	icv	string			R	yes	Spec version of the core specification this device is implemented to, The syntax is "ocf.<major>.<minor>.<sub-version>" where <major>, <minor>, and <sub-version> are the major, minor and sub-version numbers of the specification respectively. This version of the specification the string value shall be “ocf.1.0.0”.
<b>Device ID</b>	di	uuid			R	yes	Unique identifier for Device. This value shall be the same value (i.e. mirror) as the doxm.deviceuuid Property as defined in OCF Security. Handling privacy-sensitivity for the “di” Property, refer to section 13.8 in OCF Security.
<b>Data Model Version</b>	dmv	csv			R	yes	Spec version of the Resource Specification to which this device data model is implemented; if

							implemented against a Vertical specific device specification(s), then the Spec version of the vertical specification this device model is implemented to. The syntax is a comma separated list of " <res>.<major>.<minor>.<sub-version> or <vertical>.<major>.<minor>.<sub-version>. <res> is the string "ocf.res" and <vertical> is the name of the vertical defined in the Vertical specific resource specification. The <major>, <minor>, and <sub-version> are the major, minor and sub-version numbers of the specification respectively. This version of the specification one entry in the csv string shall be"ocf.res.1.0.0". Another entry in the csv shall be the vertical(s) being (e.g. "ocf.sh.1.0.0").. This value may be extended by the vendor. The syntax for extending this value, as a comma separated entry, by the vendor shall be by adding x.<Domain_Name>.<vendor_string>. For example "ocf.res.1.0.0, ocf.sh.1.0.0, x.com.example.string", The order of the values in the comma separated string can be in any order (i.e. no prescribed order). This property shall not exceed 256 octets.
Protocol Independent ID	piid	UUID			R	yes	A unique and immutable Device identifier. A Client can detect that a single Device supports multiple communication protocols if it discovers that the Device uses a single Protocol Independent ID value for all the protocols it supports. Handling privacy-sensitivity for the "piid" Property, refer to section 13.8 in OCF Security.
Localized Descriptions	ld	array			R	no	Detailed description of the Device, in one or more languages. This property is an array of objects where each object has a 'language' field (containing an IETF RFC 5646 language tag) and a 'value' field containing the device description in the indicated language.
Software Version	sv	string			R	no	Version of the device software.
Manufacturer Name	dmn	array			R	no	Name of manufacturer of the Device, in one or more languages. This property is an array of objects where each object has a 'language' field (containing an IETF RFC 5646 language tag) and a 'value' field containing the manufacturer name in the indicated language.
Model Number	dmno	string			R	no	Model number as designated by manufacturer.

2283

2284 The additional Resource Type(s) of the “/oic/d” resource are defined by the vertical specification.

2285

2286 Table 21 defines “oic.wk.p” Resource Type.

2287

2288

**Table 21. “oic.wk.p” Resource Type definition**

Property title	Property name	Value type	Value rule	Unit	Access mode	Mandatory	Description
<b>Platform ID</b>	pi	string			R	yes	Unique identifier for the physical platform (UUID); this shall be a UUID in accordance with IETF RFC 4122. It is recommended that the UUID be created using the random generation scheme (version 4 UUID) specific in the RFC. Handling privacy-sensitivity for the “pi” Property, refer to section 13.8 in OCF Security.
<b>Manufacturer Name</b>	mnmn	string			R	yes	Name of manufacturer
<b>Manufacturer Details Link</b>	mnml	uri			R	no	Reference to manufacturer, represented as a URI
<b>Model Number</b>	mnmo	string			R	no	Model number as designated by manufacturer
<b>Date of Manufacture</b>	mndt	date		Time ( <i>show RFC</i> )	R	no	Manufacturing date of Platform.
<b>Platform Version</b>	mnpv	string			R	no	Version of platform – string (defined by manufacturer)
<b>OS Version</b>	mnos	string			R	no	Version of platform resident OS – string (defined by manufacturer)
<b>Hardware Version</b>	mnhw	string			R	no	Version of platform hardware
<b>Firmware version</b>	mnfv	string			R	no	Version of Platform firmware
<b>Support link</b>	mnsi	uri			R	no	URI that points to support information from manufacturer
<b>SystemTime</b>	st	date-time			R	no	Reference time for the Platform.

Vendor ID	vid	string			R	no	Vendor defined string for the platform. The string is freeform and up to the vendor on what text to populate it.
-----------	-----	--------	--	--	---	----	--

2289

## 2290 Composite Device

2291 A physical device may be modelled as a single device or as a composition of other devices. For  
 2292 example a refrigerator may be modelled as a composition, as such part of its definition of may  
 2293 include a sub-tending thermostat device which itself may be composed of a sub-tending  
 2294 thermometer device.

2295 There may be more than one way to model a server as a composition. One example method would  
 2296 be to have Platform which represents the composite device to have more than one instance of a  
 2297 Device on the Platform. Each Device instance represents one of the distinct devices in the  
 2298 composition. Each instance of Device may itself have or host multiple instances of other resources.

2299 An implementation irrespective of how it is composed shall only expose a single instance of “/oic/d”  
 2300 with an ‘rt’ of choice for each logical Server.

2301 Thus, for the above refrigerator example if modeled as a single Server; “/oic/res” would expose  
 2302 “/oic/d” with a Resource Type name appropriate to a refrigerator. The sub-tending thermostat and  
 2303 thermometer devices would be exposed simply as instances of a resource with a device  
 2304 appropriate Resource Type with an associated URI assigned by the implementation; e.g.,  
 2305 /MyHost/MyRefrigerator/Thermostat and /MyHost/MyRefrigerator/Thermostat/Thermometer.

2306

### 2307 11.3.5 Resource discovery using “/oic/res”

2308 Discovery using “/oic/res” is the default discovery mechanism that shall be supported by all Devices  
 2309 as follows:

- 2310 a) Every Device updates its local “/oic/res” with the resources that are discoverable (see section  
 2311 7.3.2.2). Every time a new resource is instantiated on the Device and if that resource is  
 2312 discoverable by a remote Device then that resource is published with the “/oic/res” resource  
 2313 that is local to the Device (as the instantiated resource).
- 2314 b) A Device wanting to discover resources or Resource Types on one or more remote Devices  
 2315 makes a RETRIEVE request to the “/oic/res” on the remote Devices. This request may be sent  
 2316 multicast (default) or unicast if only a specific host is to be probed. The RETRIEVE request  
 2317 may optionally be restricted using appropriate clauses in the query portion of the request.  
 2318 Queries may select based on Resource Types, interfaces, or properties.
- 2319 c) The query applies to the representation of the resources. “/oic/res” is the only resource whose  
 2320 representation has “rt”. So “/oic/res” is the only resource that can be used for Multicast  
 2321 discovery at the transport protocol layer.
- 2322 d) The Device receiving the RETRIEVE request responds with a list of resources, the Resource  
 2323 Type of each of the resources and the interfaces that each resource supports. Additionally,  
 2324 information on the policies active on the resource can also be sent. The policy supported  
 2325 includes observability and discoverability. (More details below)
- 2326 e) The receiving Device may do a deeper discovery based on the resources returned in the  
 2327 request to “/oic/res”.

2328

2329 The information that is returned on discovery against “/oic/res” is at the minimum:

- 2330 • The URI (relative or fully qualified URL) of the resource
- 2331 • The Resource Type(s) of each resource. More than one Resource Type may be returned if the  
2332 resource enables more than one type. To access resources of multiple types, the specific  
2333 Resource Type that is targeted shall be specified in the request.
- 2334 • The Interfaces supported by that Resource. Multiple interfaces may be returned. To access a  
2335 specific interface that interface shall be specified in the request. If the interface is not specified,  
2336 then the Default Interface is assumed.

2337 Different “/oic/res” responses are returned according to requesting Clients, which indicate their  
2338 preference with Content Format in Accept Option. OCF 1.0 Clients request with the Content Format  
2339 of “application/vnd.ocf+cbor”, whereas the absence of that Content Format  
2340 (i.e. “application/vnd.ocf+cbor”) indicates OIC 1.1 Clients.

2341 For OIC 1.1 Clients, “/oic/res” response shall use “sec” and “port” to provide the information for an  
2342 encrypted connection.

2343 For OCF 1.0 Clients, “/oic/res” response only includes the “array of Links to conform to  
2344 IETF RFC 6690. Each Link shall use “eps” Parameter to provide the information for an encrypted  
2345 connection and carry “anchor” of the value OCF URI where the authority component of <deviceId>  
2346 indicates the Device hosting the target Resource.

2347 The JSON schemas for discovery using “/oic/res” are described in D.10. Also refer to Section 10  
2348 (Endpoint Discovery) for details of Multicast discovery using “/oic/res” on a CoAP transport.

2349 For example, a Light device might return the following to OIC 1.1 clients:

```
2350 [
2351   {
2352     "di": "e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",
2353     "links": [
2354       {
2355         "href": "coaps://[fe80::b1d6]:44444/oic/res",
2356         "rel": "self",
2357         "rt": ["oic.wk.res"],
2358         "if": ["oic.if.ll", "oic.if.baseline"],
2359         "p": {"bm": 3}
2360       },
2361       {
2362         "href": "/oic/p",
2363         "rt": ["oic.wk.p"],
2364         "if": ["oic.if.r", "oic.if.baseline"],
2365         "p": {"bm": 3, "sec": true, "port": 11111}
2366       },
2367       {
2368         "href": "/oic/d",
2369         "rt": ["oic.wk.d", "oic.d.light"],
2370         "if": ["oic.if.r", "oic.if.baseline"],
2371         "p": {"bm": 3, "sec": true, "port": 11111}
2372       },
2373       {
2374         "href": "/myLight",
2375         "rt": ["oic.r.switch.binary"],
2376         "if": ["oic.if.a", "oic.if.baseline"],
2377         "p": {"bm": 3, "sec": true, "port": 11111}
2378       }
2379     ]
2380   }
2381 ]
```



2382 The light device might return the following to clients that request with the Content Format of  
2383 "application/vnd.ocf+cbor" in Accept Option:

```
2384 [
2385   {
2386     "href": "/oic/res",
2387     "anchor": "ocf://dc70373c-1e8d-4fb3-962e-017eaa863989/oic/res",
2388     "rel": "self",
2389     "rt": ["oic.wk.res"],
2390     "if": ["oic.if.ll", "oic.if.baseline"],
2391     "p": {"bm": 3},
2392     "eps": [{"ep": "coap://[fe80::b1d6]:44444"}]
2393   },
2394   {
2395     "href": "/oic/p",
2396     "anchor": "ocf://dc70373c-1e8d-4fb3-962e-017eaa863989",
2397     "rt": ["oic.wk.p"],
2398     "if": ["oic.if.r", "oic.if.baseline"],
2399     "p": {"bm": 3},
2400     "eps": [{"ep": "coap://[fe80::b1d6]:44444"},
2401             {"ep": "coaps://[fe80::b1d6]:11111"}]
2402   },
2403   {
2404     "href": "/oic/d",
2405     "anchor": "ocf://dc70373c-1e8d-4fb3-962e-017eaa863989",
2406     "rt": ["oic.wk.d", "oic.d.light"],
2407     "if": ["oic.if.r", "oic.if.baseline"],
2408     "p": {"bm": 3},
2409     "eps": [{"ep": "coap://[fe80::b1d6]:44444"},
2410             {"ep": "coaps://[fe80::b1d6]:11111"}]
2411   },
2412   {
2413     "href": "/myLight",
2414     "anchor": "ocf://dc70373c-1e8d-4fb3-962e-017eaa863989",
2415     "rt": ["oic.r.switch.binary"],
2416     "if": ["oic.if.a", "oic.if.baseline"],
2417     "p": {"bm": 3},
2418     "eps": [{"ep": "coap://[fe80::b1d6]:44444"},
2419             {"ep": "coaps://[fe80::b1d6]:11111"}]
2420   }
2421 ]
```

2425 After performing discovery using "/oic/res", Clients may discover additional details about Server  
2426 by performing discovery using "/oic/p", /oic/rts etc. If a Client already knows about Server it may  
2427 discover using other resources without going through the discovery of "/oic/res".

## 2428 11.3.6 Resource directory (RD) based discovery

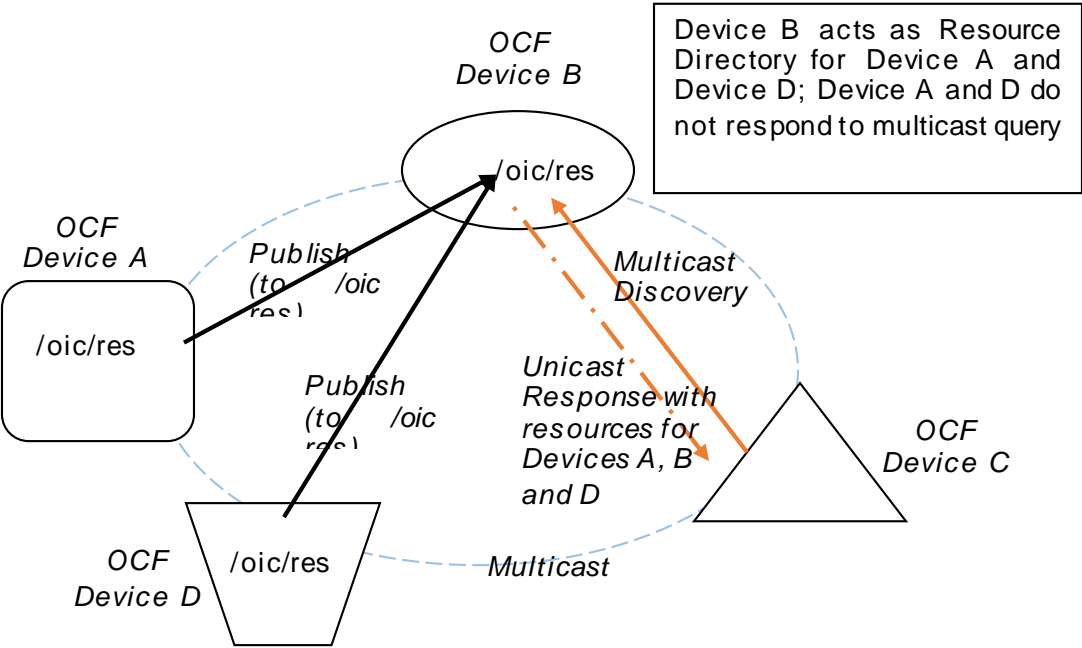
### 2429 11.3.6.1 Introduction

#### 2430 11.3.6.1.1 Indirect discovery for lookup of the resources

2431 Direct discovery is the mechanism used currently to find resources in the network. When needed,  
2432 resources are queried at a particular node directly or a multicast packet is sent to all nodes. Each  
2433 queried node responds directly with its discoverable resources to the discovering device.  
2434 Resources available locally are registered on the same device.

2435 In some situations, one of the other mechanisms described in section 11.3.2.3, called indirect  
2436 discovery, may be required. Indirect discovery is when a 3rd party device, other than the

discovering device and the discovered device, assists with the discovery process. The 3rd party only provides information on resources on behalf of another device but does not host resources on part of that device.



**Figure 26. Indirect discovery of resource by resource directory**

Indirect discovery is useful for a resource constrained device that needs to sleep to manage power and cannot process every discovery request, or when devices may not be on the same network and requires optimization for discovery. Once resources are discovered using indirect discovery then the access to the resource is done by a request directly to the Device that hosts that resource.

### 11.3.6.1.2 Resource directory

A resource directory (RD) is a Device that assists with indirect discovery. A Device which acts as an RD will be involved in the following operations.

- **RD discovery** – the procedure with which OCF Devices discover an RD and acquire the criteria to select one among multiple RDs.
- **Resource publish** – the procedures with which OCF Devices publish their Resource information, i.e. Links, subsequently update the published Links or deletes the ones.
- **Resource exposure** – the feature with which RDs expose the Links hosted by the 3<sup>rd</sup> party Devices via their “/oic/res”.

For the above, RDs make use of a core Resource Type “/oic/rd” i.e., “oic.wk.rd” defined in Table 22 and Table 23. A Device exposes “oic.wk.rd” in its “/oic/res” to announce that it serves as an RD along with selection criteria. A publishing Device can send POST request to “/oic/rd” with its Links in the payload to publish or update the Links in “/oic/res” of the RD. Also the publishing Device can send DELETE request to “/oic/rd” to delete the existing Links from “/oic/res” of the RD.

**Table 22. “oic.wk.rd” Resource Type definition**

Pre-defined URI	Resource Type Title	Resource Type ID (“rt” value)	Interfaces	Description	Related Functional Interaction
-----------------	---------------------	-------------------------------	------------	-------------	--------------------------------

“/oic/rd”	Resource directory	“oic.wk.rd”	“oic.if.baseline”	<p>The discoverable Resource Type through which an RD 1) facilitates its discovery and provides the criteria to select an RD and 2) allows OCF Devices to publish, update and delete their Links in “/oic/res” of the RD.</p> <p>A Device can find the presence of “oic.wk.rd” to discover an RD, then sends GET request to “/oic/rd” to acquire the selection criteria. An OCF Device can send POST request with Links in its payload to expose those Links in “/oic/res” of the RD. Also OCF Device can send DELETE request with suitable query (e.g. “di” or “ins”) to remove its Links from “/oic/res” of the RD.</p>	Discovery
-----------	--------------------	-------------	-------------------	---	-----------

2461

2462

**Table 23. “oic.wk.rd” Properties**

Property title	Property name	Value type	Value rule	Unit	Access mode	Mandatory	Description
<b>Selector</b>	sel	Integer or JSON Object			R	yes	Provides the criteria for RD selection. Either JSON Object describing the selection criteria (e.g. Power) specified in 11.3.6.2.2.1 or an integer representing a bias factor calculated by RD. The value is in the range of 0 to 100 - 0 implies that RD is not to be selected. Client chooses RD with highest bias factor or randomly between RDs that have same bias factor.

2463

2464 A RD can be queried at its “/oic/res” resource to find resources hosted on other Devices. These  
2465 Devices can be sleepy nodes or any other device that cannot or may not respond to discovery  
2466 requests. Device can publish all or partial list of resources they host to a RD. The RD then responds  
2467 to queries for Resource discovery on behalf of the publishing Device (for example: when a Device  
2468 may go to sleep). For general Resource discovery, the RD behaves like any other Server in  
2469 responding to requests to “/oic/res”.

2470 Any Device that serves or acts as a RD shall expose a well-known resource “/oic/rd”. The Devices  
2471 that want to discover RDs shall use this resource and one of the Resource discovery mechanisms  
2472 to discover the RD and get the parameters of the RD. The information discovered through this  
2473 resource shall be used to select the appropriate RD to use for resource publication. The bias  
2474 information includes the following criteria: power source (AC, battery powered or safe/reliable),  
2475 connectivity (wireless, wired), CPU, memory, load statistics (processing publishing to 100.  
2476 Optionally, the RD may also return a context - the value which shall be a string and semantics of  
2477 the context are not discussed in this document but it is expected that the context will be used to  
2478 establish a domain, region or some such scope that is meaningful to the application, deployment  
2479 or usage.

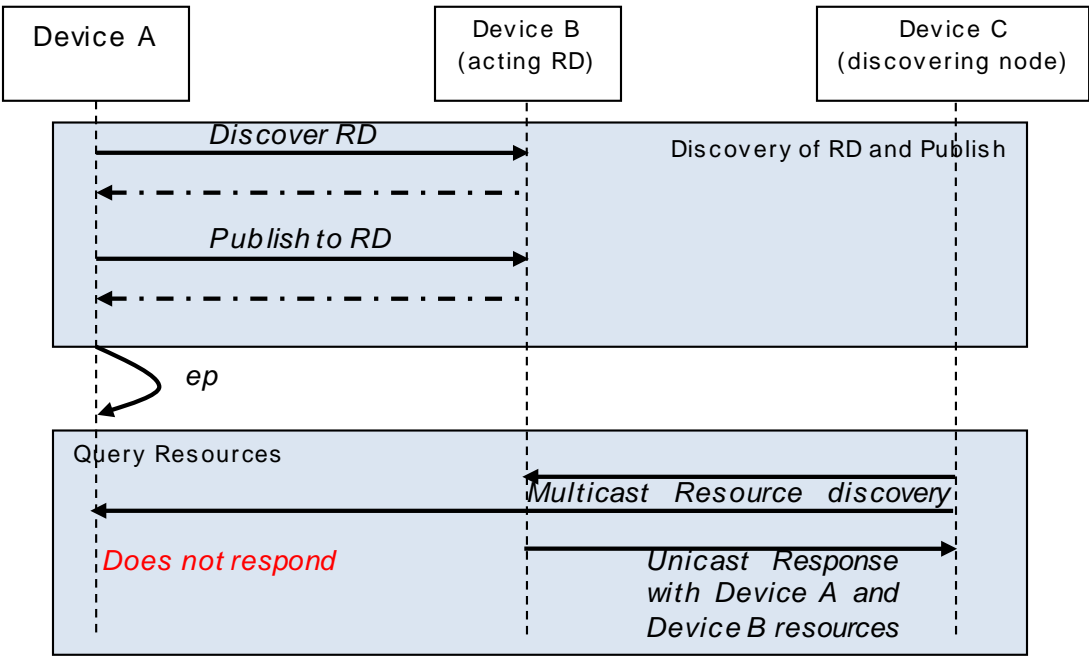
2480 Using these criteria or the bias factor, the Device should select one RD (per context) to publish its  
2481 resources. A context describes the state of an OCF Device with respect to Resource discovery. A  
2482 context is usually determined at deployment and from application requirements. An example of a  
2483 context could be a multicast group- a Device that is a member of more than one multicast group  
2484 may have to find and select a RD in each of the multicast groups (i.e. per context) to publish its  
2485 information.

This remainder of this section is divided into three parts. The first part covers “RD Discovery” (section 11.3.6.2.2), i.e., discovering and selecting of the RD. The second part “Resource publish” (section ), i.e., publishing, updating and deleting of resources for the constrained/sleepy device. The third part “Resource exposure” (section) where RD replies to queries from devices looking to discover resources.

**11.3.6.2 RD discovery**

**11.3.6.2.1 Discovering a resource directory**

An RD in the OCF network shall support RD discovery, shall provide the facility to allow devices to publish their resource information to a RD, to update resource information in an RD and to delete resource information from an RD.



**Figure 27. RD discovery and RD supported query of resources support**

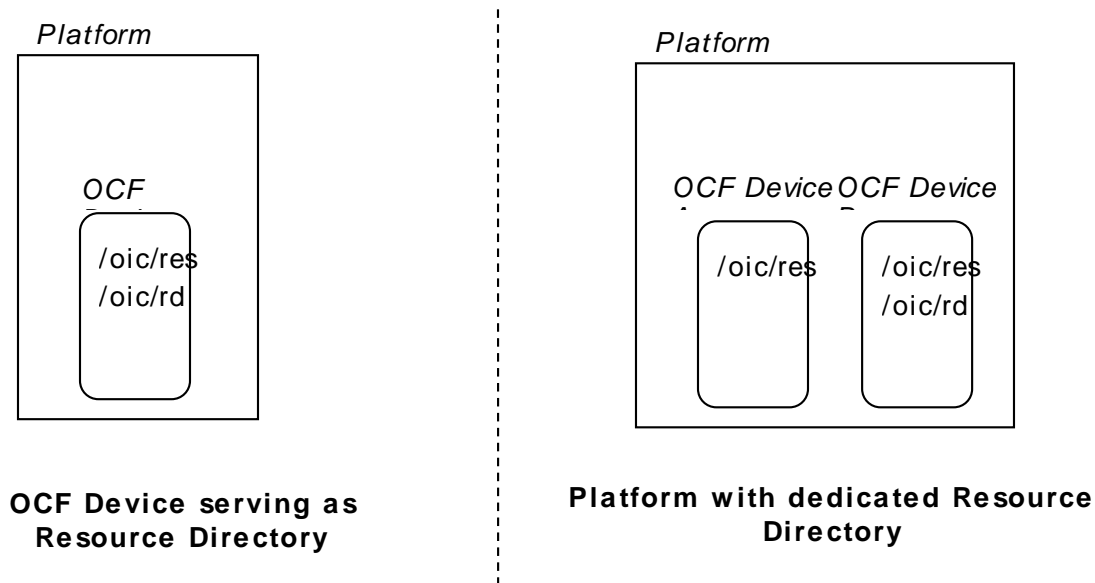
As shown in Figure 27, the Device that wishes to advertise its resources: first discovers a resource directory and then publishes the desired resource information. Once a set of resources have been published to an RD then the publishing device should not respond to multicast Resource discovery queries for those published resources when the RD is on the same multicast domain. In that case, only the RD should respond to multicast Resource discovery requests on the resource published to it.

An OCF network allows for more than one device acting as an RD. The reason to have multiple RD support is to make network scalable, handle network failures and centralized device failure bottleneck. This does not preclude a scenario where a use case or deployment environment may require single device in the environment to be deployed as the only resource directory (e.g. gateway model). There may be more than one Device acting as RD on a Platform.

Discovering of an RD may result in responses from more than one RD. The discovering device shall select an RD. The selection may be based on the weightage parameter(s) provided in the response from the RD.

An RD will be application agnostic i.e., application should not be aware whether resource directory was queried to get the resource information. All the handling of the retrieval is kept opaque to the

application. A Client that performs Resource discovery uses an RD just like it may use any other Server for discovery. It may send a unicast request to the RD when it needs only the resource advertised on the RD or do a multicast query when it does not require or have explicit knowledge of an RD.



**Figure 28. Resource Direction Deployment Scenarios**

RD can also be discovered in the following manners:

- **Pre-configuration:** Devices wishing to publish resource information may be configured a priori with the information (e.g. IP address, port, transport etc.) of a specific resource directory. This pre-configuration may be done at onboarding or may be updated on the device using an out-of-band method. This pre-configuration may be done by the manufacturer or by the user/device manager.
- **Query-oriented:** A Client wanting to discover resource directories using query-oriented discovery (i.e. pull) can issue a multicast Resource discovery request for “/oic/res?rt=oic.wk.rd”. Only and all Devices that can be an RD will respond to this query. The “/oic/rd” response shall include information about the RD i.e., the presence of “oic.wk.rd” Link (as defined by the Resource Type) and subsequent query to “/oic/rd” would produce weightage parameters to allow the discovering device to select between RDs (see details in RD selection section). The “oic.wk.rd” resource shall be instantiated on the OCF Devices acting as a resource directory. The “oic.wk.rd” schema is as defined in D.14.
- **Advertisement:** An RD may advertise about itself to devices. It is an advertisement packet. The devices that are already publishing to a RD may use this as a heartbeat message of the RD. If the RD advertisement does not arrive at a stipulated interval, publishing device starts searching for other RDs in the network, as this is a signal that RD is not online. Other usage of this message is it serves as an advertisement for a device seeking a RD to publish their resources. The details from the advertisement can then be used to query directly to a RD to get weightage details instead of sending a multicast packet in a network. As it is intended this is sent at a regular interval and does not include weightage information to keep packet sizes small. Further details may be presented in the later version of this specification.
- One of the important benefits of an RD is to make services discoverable in networks that don't support site wide multicast but do support site wide routing. An example of such a network is Homenet..To enable an RD function across such a network a site discovery mechanism is needed to discover the RD service (IP address & port number). In order to make itself

discoverable beyond the link local scope, an RD with a routable ip address should implement the mDNS responder requirements defined in IETF RFC 6762. Further details for such an operation may be specified in a later version of this specification, when the needs arise.

### 11.3.6.2.2 RD selection process

#### 11.3.6.2.2.1 Selection criteria

When a device discovers more than one RD then it should decide to use one of these RDs based on the selection criteria described here. A device should use or publish information to only one RD within a multicast domain at a given time. This is to minimize the burden of processing duplicate information in the Resource discovery phase.

There two ways to select an RD. One is based on a bias factor (RD generated) and the other is based on clients determination based on granular parameters provided by the server (client/device generated). Devices may use one or both methods to select an RD.

**Bias factor:** The bias factor is a server generated positive number in the range of 0 to 100, where 0 is the lowest to 100 being the highest. If two RDs have the same bias factor then the selecting device may choose either based auxiliary criteria or at random. Either way only one RD should be selected and used at a time. No specific method is defined in this specification to determine the bias factor for an RD. The number may be a pre-configured value at the time of onboarding or subsequent configuration of the RD or may be based on a formula determined by the implementation of the RD. (OCF may provide a standard formula for this calculation in a future version or release of this specification, if the needs arise).

The bias factor can be calculated by the RD by adding the contribution values determined for each of the parameters in Table 24 and divided by the number of parameters. An RD may advertise a bias factor larger than the calculated value when there is reason to believe that the RD is highly capable for example an installed service provider gateway.

**Parameters:** Optionally, parameters defined in Table 24 (like direct power supply, network connectivity, load conditions, CPU power, memory, etc.) may be returned in the "/oic/rd" discovery response. Discovering device may use the details to make granular selection decisions based on client defined policies and criteria that use the RD parameters. For example, a device in an industrial deployment may not weight power connectivity high but another in home environments may give more weightage for power.

**Table 24: Selection parameters**

Parameter	Values (Contribution)	Description
Power	Safe (100) AC (70) Batt (40)	<ul style="list-style-type: none"> <li>Safe implies that the power supply is reliable and is backed up with battery for power outages etc.</li> <li>Implementation may lower the number for Batt based on the type of battery the RD device runs on. If battery conservation is important then this number should be lowered.</li> </ul>
Mobility	Fixed (100) Mobile (50)	<ul style="list-style-type: none"> <li>Implementation may further grade the mobility number based on how mobile the RD device is; lower number for highly mobile and larger numbers for limited mobility</li> <li>The mobility number shall not be larger than 80</li> </ul>
Network Product	Type: <ul style="list-style-type: none"> <li>Wired (10)</li> <li>Wireless (4)</li> </ul> Bandwidth: <ul style="list-style-type: none"> <li>High (10)</li> <li>Low (5)</li> </ul>	<ul style="list-style-type: none"> <li>Network product = [sum of (type * bandwidth per network interface)]/[number of interfaces]</li> <li>Normalized to 100</li> </ul>

	• Lossy (3) Interfaces	
Memory Factor	Available Total	<ul style="list-style-type: none"> <li>Memory is the volatile or non-volatile storage used to store the resource information</li> <li>Memory Factor = [Available]/[Total]</li> <li>Normalized to 100 (i.e. expressed as percentage)</li> </ul>
Request Load Factor	1-minute 5-minute 15-minutes	<ul style="list-style-type: none"> <li>Current request loading of the RD</li> <li>Similar to UNIX load factor (using observable, pending and processing requests instead of runnable processes)</li> <li>Expressed as a load factor 3-tuple (up to two decimal points each). Factor is based on request processed in a 1-minute (L1), 5-minute (L5) and 15-minute (L15) windows</li> <li>See <a href="http://www.teamquest.com/import/pdfs/whitepaper/ldavq1.pdf">http://www.teamquest.com/import/pdfs/whitepaper/ldavq1.pdf</a></li> <li>Factor = <math>100 - ([L1*3 + L5*7 + L15*10]/3)</math></li> </ul>

2578

### 2579 11.3.6.2.2 Selection scenarios

2580 The device that wants to use an RD will find zero or more RDs on the network. After discovering  
2581 the RDs, the device needs to select an RD of all found RDs on the network. The selection based  
2582 on the bias factor will ensure that a Device can judge if the found RD is suitable for its needs.

2583 The following situation can occur during the selection of an RD:

2584 1) A single or multiple RDs are present in the network

2585 2) No RD is present in the network

2586 3) an additional RD arrives on the network

2587 In the first scenario the RDs are already present. If a single RD is detected then that RD can be  
2588 used . When multiple RDs are detected the Device uses the bias information to select the RD.

2589 In the second scenario, device will listen to the advertisement of the devices that hosts the RDs.  
2590 Once an RD advertisement packet is received it judges if the bias criteria are met and starts using  
2591 the RDs.

2592 In the third scenario the Device has already published its resources to an existing RD. In this  
2593 scenario it discovers a new RD on the network.

2594 After judging the bias factor the Device may choose to move to the new RD. If the decision is made  
2595 to select the new RD, the then Device should delete its resource information from the current used  
2596 RD and then after removal publish the information to the new RD. During the transition period the  
2597 Device itself should respond to Resource discovery requests.

### 2598 11.3.6.3 Resource publish

#### 2599 11.3.6.3.1 Publish resources

##### 2600 11.3.6.3.1.1 Overview

2601 After the selection process of an RD, a device may choose one of the following mechanisms:

- 2602 • Push its resources information to the selected RD or
- 2603 • Request the RD to pull the resource information by doing a unicast discovery request against
- 2604 its "/oic/res"

2605 The publishing device may decide to publish all resources or few resources on the resource  
2606 directory. The publishing device shall only publish resources that are otherwise published to its  
2607 own “/oic/res”. A publishing device may respond to discovery requests (on its “/oic/res” resource)  
2608 for the resources it does not publish to a RD. Nonetheless, it is highly recommended that when an  
2609 RD is used, all discoverable resources on the publisher be published to the RD.

#### 2610 **11.3.6.3.1.2 Publish: Push resource information**

2611 Resource information is published using an UPDATE operation to “/oic/rd” with “rt” query of  
2612 “?rt=oic.wk.rdpub” and the “oic.if.baseline” interface.

2613 A Device, which hosts a Resource, can publish the Resource information, i.e. the Link targeting  
2614 the Resource, to an RD by sending a POST request with the Link in the payload. The published  
2615 Link will be exposed through the “/oic/res” of the RD.

2616 When a Device first publishes a Link or Links, it sends a POST request to “/oic/rd” Resource  
2617 including the following key-value pairs in the payload

- 2618 • **di** – as its value, a unique identifier for the publishing Device, i.e. its device ID.
- 2619 • **links** – as its value, the array of Links to be published. Links may not include “ins” Parameter.
- 2620 • **ttl** – as its value, the time to indicate the RD how long to keep this published item. After this  
2621 time (in seconds) elapses, the RD invalidates the links. To keep link alive the publishing device  
2622 updates the ttl using the update schema.

2623 Take notice that the payload shall carries the appropriate Content-Format of  
2624 “application/vnd.ocf+cbor”.

```
{
  "di": "e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",
  "links": [
    {
      "anchor": "ocf://e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",
      "href": "/myLightSwitch",
      "rt": ["oic.r.switch.binary"],
      "if": ["oic.if.a", "oic.if.baseline"],
      "p": {"bm": 3},
      "eps": [
        {"ep": "coaps://[fe80::b1d6]:1111", "pri": 2},
        {"ep": "coaps://[fe80::b1d6]:1122"},
        {"ep": "coaps+tcp://[2001:db8:a::123]:2222", "pri": 3}
      ]
    },
    {
      "anchor": "ocf://e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",
      "href": "/myLightBrightness",
      "rt": ["oic.r.brightness"],
      "if": ["oic.if.a", "oic.if.baseline"],
      "p": {"bm": 3},
```



```

        "eps": [
            {"ep": "coaps://[2001:db8:a::123]:2222"}
        ]
    },
    "ttl": 600
}

```

**Figure 29. Example of POST request payload**

When an RD receives the POST request, it determines whether to grant the request or not. Upon granting the request, for each Link to be published, the RD assigns a unique instance value identifying the Link among all the Links it advertises and includes the identifying value in “ins” Parameter of the Link. The RD may use the “ins” value which the publishing Device includes in the POST request payload as long as the “ins” value doesn't match with any existing “ins” value in the published Link. The RD adds the new Links to its “/oic/res” and exposes them to a valid discovery query, i.e. GET request.

The RD sends back the POST response to the Publishing Device with the same payload as in the matching POST request with possible differences of 1) “ins” inclusion in each Link and 2) different “ttl” value. Take notice that each published Link in RD response payload shall carry “ins” Parameter to provide the publishing Device of the identifier with which it can further UPDATE or DELETE the Link.

```

{
    "di": "e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",
    "links": [
        {
            "anchor": "ocf://e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",
            "href": "/myLightSwitch",
            "rt": ["oic.r.switch.binary"],
            "if": ["oic.if.a", "oic.if.baseline"],
            "p": {"bm": 3},
            "eps": [
                {"ep": "coaps://[fe80::b1d6]:1111", "pri": 2},
                {"ep": "coaps://[fe80::b1d6]:1122"},
                {"ep": "coaps+tcp://[2001:db8:a::123]:2222", "pri": 3}
            ],
            "ins": "11235"
        },
        {
            "anchor": "ocf://e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",
            "href": "/myLightBrightness",
            "rt": ["oic.r.brightness"],

```

```

        "if":      ["oic.if.a", "oic.if.baseline"],
        "p":      {"bm": 3},
        "eps": [
            {"ep": "coaps://[[2001:db8:a::123]:2222"]}
        ],
        "ins":     "112358"
    }
],
"ttl": 600
}

```

**Figure 30. Example of POST response payload**

Once a publishing device has published resources to an RD, it may not respond to the multicast discovery queries for the same resources against its own "/oic/res", especially when on the same multicast domain as the RD. After publishing resources, primarily it is an RD responsibility to reply to the queries for the published resources.

If the publishing device is in sleep mode and an RD has replied on behalf of the publishing device, then a discovering device will try to access resource on the provided URI.

There is another possibility that the resource directory and the publishing device both respond to the multicast query from the discovering device. This will create a duplication of the information but is an alternate that may be used for non-robust network. It is not a recommended option but for industrial scenarios, this is one of the possibilities. Either way, discovering clients shall always be prepared to process duplicate information in responses to multicast discovery request. The "/oic/rd" schema is as defined in D.14 to specify publishing to the "/oic/rd" Resource.

#### 11.3.6.3.2 Update resource information

An RD will hold the published Link till the time specified in the ttl field. A publishing Device can send update if it seeks the RD to keep holding the Link or modify the published Link (e.g. changing Endpoint information). UPDATE can be used for updating about all resources that are published on an RD or per resource published.

UPDATES in CoAP are done using the same POST request to "oic/rd". POST request message will be of the same payload format but the each Link to be modified shall include the "ins" Parameter which the RD previously provided in POST response message.

Upon granting the request, the RD reflects the change to the Link in its "/oic/res" and sends back the POST response of the same format as the initial publishing.

#### 11.3.6.3.3 Delete resource information

A resource information hold at the resource directory can be removed anytime by the publishing device. It can be either for the whole device information or for a particular resource. This request should be only allowed when device meets a certain requirement, as it can create potential security issue.

A publishing Device can delete published Link or Links from an RD by sending a DELETE request with the query "di" or "ins" indicating the Links to be deleted.

- **di** – This is used to determine which set of links to delete. (Need authentication to ensure that there is no spoofing). It's the form of di=value, where value is a device ID indicating the Device

2672 to operate on. When present, the entire set of links corresponding to the device ID is deleted,  
 2673 i.e. the Links published by the publishing Device with the same device ID are deleted.

- 2674 • **ins** – Instance of the Link to delete. Value of parameter is a string indicating the instance to be  
 2675 deleted. When present, the Link with the same instance value is deleted.

```
DELETE /oic/rd?di=0685B960-736F-46F7-BEC0-9E6CBD671ADC1

DELETE /oic/rd?ins=20
```

**Figure 31. Example of DELETE request with "di" or "ins" query**

2677 When a publishing Device wants to remove published Link or Links from an RD, it sends DELETE  
 2678 request with "di" or "ins" indicating the Links to be removed. Upon granting the request, the RD  
 2679 removes the identified Links and sends back the DELETE response.

2680 Selective deletion of information for individual resources is not possible the case where the RD  
 2681 pull the resource information due to the absence of :ins" value. The publishing device can request  
 2682 a delete but only for all the resource information that the RD has pulled from that device. In this  
 2683 case, the DELETE request has the device ID "di" in the query.

#### 2684 **11.3.6.3.4 Transfer resource information from one RD to another**

2685 When a publishing device identifies an RD that is better suited, it may decide to publish to that RD.  
 2686 Since the device should publish to only one RD at a time, the client should ensure that previously  
 2687 published information is deleted from the currently used RD before publishing to the newly selected  
 2688 RD. The deletion of the resource may be done either by allowing the TTL to expire or explicitly  
 2689 deleting the resource information.

2690 RDs shall not transfer Resource information between themselves. It is the Client's responsibility  
 2691 to choose the RD and to manage the published Resources.

#### 2692 **11.3.6.4 Resource exposure**

##### 2693 **11.3.6.4.1 "/oic/res" and retrieving of the resources**

2694 The "/oic/res" based discovery process remains the same as that in the absence of an RD.  
 2695 Resources may be discovered by retrieving the "/oic/res" resource by sending a multicast or  
 2696 unicast request. In the case of a multicast discovery request, an RD will respond for the device  
 2697 that hosts the resources. Clients shall be prepared to process duplicate resource information from  
 2698 more than one RD responding with the same information or from an RD and the hosting device  
 2699 (publishing the resource information) both responding to the request. Interaction with resources  
 2700 discovered using the RD is done using the same mechanism and methods as with resources  
 2701 discovered by retrieving the "/oic/res" resource of the device hosting the resources (e.g., connect  
 2702 to the resource and perform CRUDN operations on the resource).

2703 Resource Directory provides different "/oic/res" response according to requesting Clients, which  
 2704 indicate their preference with content format. OCF 1.0 Clients request with the "Content Format of  
 2705 "application/vnd.ocf+cbor", whereas the absence of the Content-Format indicates OIC 1.1 Clients.

2706 For OIC 1.1 Clients, "/oic/res" response includes to OIC 1.1 Link and anchor parameter has transfer  
 2707 protocol URI (e.g. coap URI), if present. The Resources hosted by the same Device are grouped  
 2708 together within a single JSON Object with "di" indicating the hosting Device. The Resources  
 2709 belonging to the responding RD may omit "anchor" parameter. However, the Resources of other  
 2710 Devices shall include "anchor" parameter when "rel" value is "hosts" and its "href" value should be  
 2711 (fully qualified) transfer protocol URI with IP address and port number as its authority component  
 2712 (e.g., coaps://[2001:db8:b::c2e5]:2222/myLightSwitch) .

2713 For example, a Resource Directory might return the following to OIC 1.1 clients:

```

2714 [
2715 {
2716   "di": "88b7c7f0-4b51-4e0a-9faa-cfb439fd7f49",
2717   "links": [
2718     {
2719       "href": "/oic/res",
2720       "rel": "self",
2721       "rt": ["oic.wk.res"],
2722       "if": ["oic.if.ll", "oic.if.baseline"],
2723       "p": {"bm": 3, "sec": false, "port": 33333}
2724     },
2725     {
2726       "href": "/oic/d",
2727       "rt": ["oic.wk.d", "oic.d.fan"],
2728       "if": ["oic.if.r", "oic.if.baseline"],
2729       "p": {"bm": 3, "sec": false, "port": 33333}
2730     },
2731     {
2732       "href": "/oic/p",
2733       "rt": ["oic.wk.p"],
2734       "if": ["oic.if.r", "oic.if.baseline"],
2735       "p": {"bm": 3, "sec": true, "port": 33333}
2736     },
2737     {
2738       "href": "/myFanIntrospection",
2739       "rt": ["oic.wk.introspection"],
2740       "if": ["oic.if.r", "oic.if.baseline"],
2741       "p": {"bm": 3, "sec": true, "port": 33333}
2742     },
2743     {
2744       "href": "/oic/rd",
2745       "rt": ["oic.wk.rd"],
2746       "if": ["oic.if.baseline"],
2747       "p": {"bm": 3, "sec": true, "port": 33333}
2748     },
2749     {
2750       "href": "/myFanSwitch",
2751       "rt": ["oic.r.switch.binary"],
2752       "if": ["oic.if.a", "oic.if.baseline"],
2753       "p": {"bm": 3, "sec": true, "port": 33333}
2754     },
2755     {
2756       "href": "/oic/sec/doxm",
2757       "rt": ["oic.r.doxm"],
2758       "if": ["oic.if.baseline"],
2759       "p": {"bm": 1, "sec": false, "port": 33333}
2760     },
2761     {
2762       "href": "/oic/sec/pstat",
2763       "rt": ["oic.r.pstat"],
2764       "if": ["oic.if.baseline"],
2765       "p": {"bm": 1, "sec": true, "port": 33333}
2766     },
2767     {
2768       "href": "/oic/sec/cred",
2769       "rt": ["oic.r.cred"],
2770       "if": ["oic.if.baseline"],
2771       "p": {"bm": 1, "sec": true, "port": 33333}
2772     },
2773     {
2774       "href": "/oic/sec/acl2",
2775       "rt": ["oic.r.acl2"],
2776       "if": ["oic.if.baseline"],

```

```

2777     "p": {"bm": 1, "sec": true, "port": 33333}
2778   }
2779 ]
2780 },
2781 {
2782   "di": "dc70373c-1e8d-4fb3-962e-017eaa863989",
2783   "links": [
2784     {
2785       "anchor": "coap://[2001:db8:b::c2e5]:66666",
2786       "href": "coap://[2001:db8:b::c2e5]:66666/oic/d",
2787       "rt": ["oic.wk.d", "oic.d.light", "oic.d.virtual"],
2788       "if": ["oic.if.r", "oic.if.baseline"],
2789       "p": {"bm": 3, "sec": false, "port": 22222}
2790     },
2791     {
2792       "anchor": "coaps://[2001:db8:b::c2e5]:22222",
2793       "href": "coaps://[2001:db8:b::c2e5]:22222/oic/p",
2794       "rt": ["oic.wk.p"],
2795       "if": ["oic.if.r", "oic.if.baseline"],
2796       "p": {"bm": 3, "sec": true, "port": 22222}
2797     },
2798     {
2799       "anchor": "coaps://[2001:db8:b::c2e5]:22222",
2800       "href": "coaps://[2001:db8:b::c2e5]:22222/myLightSwitch",
2801       "rt": ["oic.r.switch.binary"],
2802       "if": ["oic.if.a", "oic.if.baseline"],
2803       "p": {"bm": 3, "sec": true, "port": 22222}
2804     },
2805     {
2806       "anchor": "coaps://[2001:db8:b::c2e5]:22222",
2807       "href": "coaps://[2001:db8:b::c2e5]:22222/myLightBrightness",
2808       "rt": ["oic.r.brightness"],
2809       "if": ["oic.if.a", "oic.if.baseline"],
2810       "p": {"bm": 3, "sec": true, "port": 22222}
2811     }
2812   ]
2813 }
2814 ]
2815

```

2816 For OCF 1.0 Clients, "/oic/res" response includes adds to the OCF 1.0 Link and the anchor  
 2817 parameter has OCF URI. "/oic/res" response has the single array of OCF 1.0 Links to conform to  
 2818 IETF RFC 6690. Each Link shall carry "anchor" of the value OCF URI where the authority  
 2819 component of <deviceId> indicates the Device hosting the target Resource.

2820 The Resource Directory might return the following to clients that request with the Content Format  
 2821 of "application/vnd.ocf+cbor":

```

2822 [
2823   {
2824     "anchor": "ocf://88b7c7f0-4b51-4e0a-9faa-cfb439fd7f49",
2825     "href": "/oic/res",
2826     "rel": "self",
2827     "rt": ["oic.wk.res"],
2828     "if": ["oic.if.ll", "oic.if.baseline"],
2829     "p": {"bm": 3},
2830     "eps": [{"ep": "coap://[2001:db8:a::b1d4]:77777"},
2831             {"ep": "coaps://[2001:db8:a::b1d4]:33333"}]
2832   },
2833   {
2834     "anchor": "ocf://88b7c7f0-4b51-4e0a-9faa-cfb439fd7f49",
2835     "href": "/oic/d",

```

```

2836     "rt": ["oic.wk.d", "oic.d.fan"],
2837     "if": ["oic.if.r", "oic.if.baseline"],
2838     "p": {"bm": 3},
2839     "eps": [{"ep": "coap://[2001:db8:a::b1d4]:7777"},
2840             {"ep": "coaps://[2001:db8:a::b1d4]:33333"}]
2841 },
2842 {
2843     "anchor": "ocf://88b7c7f0-4b51-4e0a-9faa-cfb439fd7f49",
2844     "href": "/oic/p",
2845     "rt": ["oic.wk.p"],
2846     "if": ["oic.if.r", "oic.if.baseline"],
2847     "p": {"bm": 3},
2848     "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:33333"}]
2849 },
2850 {
2851     "anchor": "ocf://88b7c7f0-4b51-4e0a-9faa-cfb439fd7f49",
2852     "href": "/myFanIntrospection",
2853     "rt": ["oic.wk.introspection"],
2854     "if": ["oic.if.r", "oic.if.baseline"],
2855     "p": {"bm": 3},
2856     "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:33333"}]
2857 },
2858 {
2859     "anchor": "ocf://88b7c7f0-4b51-4e0a-9faa-cfb439fd7f49",
2860     "href": "/oic/rd",
2861     "rt": ["oic.wk.rd"],
2862     "if": ["oic.if.baseline"],
2863     "p": {"bm": 3},
2864     "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:33333"}]
2865 },
2866 {
2867     "anchor": "ocf://88b7c7f0-4b51-4e0a-9faa-cfb439fd7f49",
2868     "href": "/myFanSwitch",
2869     "rt": ["oic.r.switch.binary"],
2870     "if": ["oic.if.a", "oic.if.baseline"],
2871     "p": {"bm": 3},
2872     "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:33333"}]
2873 },
2874 {
2875     "anchor": "ocf://88b7c7f0-4b51-4e0a-9faa-cfb439fd7f49",
2876     "href": "/oic/sec/doxm",
2877     "rt": ["oic.r.doxm"],
2878     "if": ["oic.if.baseline"],
2879     "p": {"bm": 1},
2880     "eps": [{"ep": "coap://[2001:db8:a::b1d4]:7777"},
2881             {"ep": "coaps://[2001:db8:a::b1d4]:33333"}]
2882 },
2883 {
2884     "anchor": "ocf://88b7c7f0-4b51-4e0a-9faa-cfb439fd7f49",
2885     "href": "/oic/sec/pstat",
2886     "rt": ["oic.r.pstat"],
2887     "if": ["oic.if.baseline"],
2888     "p": {"bm": 1},
2889     "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:33333"}]
2890 },
2891 {
2892     "anchor": "ocf://88b7c7f0-4b51-4e0a-9faa-cfb439fd7f49",
2893     "href": "/oic/sec/cred",
2894     "rt": ["oic.r.cred"],
2895     "if": ["oic.if.baseline"],
2896     "p": {"bm": 1},
2897     "eps": [{"ep": "coaps://[2001:db8:a::b1d4]:33333"}]
2898 },

```

```

2899 {
2900   "anchor": "ocf://88b7c7f0-4b51-4e0a-9faa-cfb439fd7f49",
2901   "href": "/oic/sec/acl2",
2902   "rt": ["oic.r.acl2"],
2903   "if": ["oic.if.baseline"],
2904   "p": {"bm": 1},
2905   "eps": [{"ep": "coaps://[2001:db8:a:b1d4]:3333"}]
2906 },
2907
2908 {
2909   "anchor": "ocf://dc70373c-1e8d-4fb3-962e-017eaa863989",
2910   "href": "/oic/d",
2911   "rt": ["oic.wk.d", "oic.d.light"],
2912   "if": ["oic.if.r", "oic.if.baseline"],
2913   "p": {"bm": 3},
2914   "eps": [{"ep": "coap://[2001:db8:b:c2e5]:6666"},
2915           {"ep": "coaps://[2001:db8:b:c2e5]:2222"}]
2916 },
2917 {
2918   "anchor": "ocf://dc70373c-1e8d-4fb3-962e-017eaa863989",
2919   "href": "/oic/p",
2920   "rt": ["oic.wk.p"],
2921   "if": ["oic.if.r", "oic.if.baseline"],
2922   "p": {"bm": 3},
2923   "eps": [{"ep": "coaps://[2001:db8:b:c2e5]:2222"}]
2924 },
2925 {
2926   "anchor": "ocf://dc70373c-1e8d-4fb3-962e-017eaa863989",
2927   "href": "/myLightSwitch",
2928   "rt": ["oic.r.switch.binary"],
2929   "if": ["oic.if.a", "oic.if.baseline"],
2930   "p": {"bm": 3},
2931   "eps": [{"ep": "coaps://[2001:db8:b:c2e5]:2222"}]
2932 },
2933 {
2934   "anchor": "ocf://dc70373c-1e8d-4fb3-962e-017eaa863989",
2935   "href": "/myLightBrightness",
2936   "rt": ["oic.r.brightness"],
2937   "if": ["oic.if.a", "oic.if.baseline"],
2938   "p": {"bm": 3},
2939   "eps": [{"ep": "coaps://[2001:db8:b:c2e5]:2222"}]
2940 }
2941 ]

```

2942

## 2943 11.4 Notification

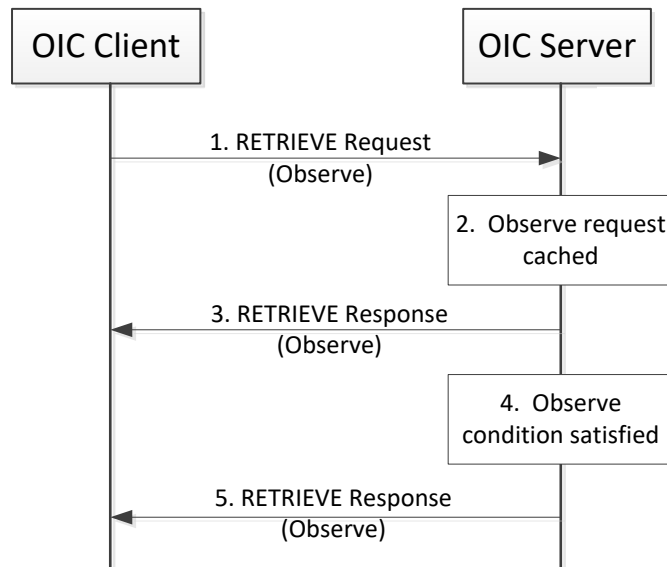
### 2944 11.4.1 Overview

2945 A Server shall support NOTIFY operation to enable a Client to request and be notified of desired  
 2946 states of one or more Resources in an asynchronous manner. Section 11.4.2 specifies the observe  
 2947 mechanism in which updates are delivered to the requester.

### 2948 11.4.2 Observe

2949 In observe mechanism the Client utilizes the RETRIEVE operation to require the Server for updates  
 2950 in case of Resource state changes. The Observe mechanism consists of five steps which are  
 2951 depicted in Figure 32 and described below.

2952 Note: the observe mechanism can only be used for a resource with a property of observable  
 2953 (section 7.3.2.2).



**Figure 32. Observe Mechanism**

#### 11.4.2.1 RETRIEVE request with observe indication

The Client transmits a RETRIEVE request message to the Server to request updates for the Resource on the Server if there is a state change. The RETRIEVE request message carries the following parameters:

- *fr*: Unique identifier of the Client
- *to*: Resource that the Client is requesting to observe
- *ri*: Identifier of the RETRIEVE request
- *op*: RETRIEVE
- *obs*: Indication for observe request

#### 11.4.2.2 Processing by the Server

Following the receipt of the RETRIEVE request, the Server may validate if the Client has the appropriate rights for the requested operation and the properties are readable and observable. If the validation is successful, the Server caches the information related to the observe request. The Server caches the value of the *ri* parameter from the RETRIEVE request for use in the initial response and future responses in case of a change of state.

#### 11.4.2.3 RETRIEVE response with observe indication

The Server shall transmit a RETRIEVE response message in response to a RETRIEVE request message from a Client. The RETRIEVE response message shall include the following parameters. If validation succeeded, the response includes an observe indication. If not, the observe indication is omitted from the response which signals to the requesting client that registration for notification was not allowed.

The RETRIEVE response message shall include the following parameters:

- *fr*: Unique identifier of the Server
- *to*: Unique identifier of the Client
- *ri*: Identifier included in the RETRIEVE request
- *cn*: Information resource representation as requested by the Client



- *rs*: The result of the RETRIEVE operation
- *obs*: Indication that the response is made to an observe request

#### 11.4.2.4 Resource monitoring by the Server

The Server shall monitor the state the Resource identified in the observe request from the Client. Anytime there is a change in the state of the observed resource, the Server sends another RETRIEVE response with the observe indication. The mechanism does not allow the client to specify any bounds or limits which trigger a notification, the decision is left entirely to the server.

#### 11.4.2.5 Additional RETRIEVE responses with observe indication

The Server shall transmit updated RETRIEVE response messages following observed changes in the state of the Resources indicated by the Client. The RETRIEVE response message shall include the parameters listed in section 11.4.2.3.

#### 11.4.2.6 Cancelling Observe

The Client can explicitly cancel observe by sending a RETRIEVE request without the observe indication field to the same resource on Server which it was observing. For certain protocol mappings, the client may also be able to cancel an observe by ceasing to respond to the RETRIEVE responses.

### 11.5 Device management

#### 11.5.1 Overview

The Device Management includes the following functions:

- Diagnostics and maintenance

The device management functionalities specified in this version of specification are intended to address the basic device management features. Addition of new device management features in the future versions of the specification is expected.

#### 11.5.2 Diagnostics and maintenance

The Diagnostics and Maintenance function is intended for use by administrators to resolve issues encountered with the Devices while operating in the field. If diagnostics and maintenance is supported by a Device, the Core Resource “/oic/mnt” shall be supported as described in Table 25.

**Table 25. Optional diagnostics and maintenance device management Core Resources**

Pre-defined URI	Resource Type Title	Resource Type ID (“rt” value)	Interfaces	Description	Related Functional Interaction
“/oic/mnt”	Maintenance	“oic.wk.mnt”	“oic.if.rw”	The resource through which the device is maintained and can be used for diagnostic purposes. The <b>resource properties</b> exposed by “/oic/mnt” are listed in Table 26.	Device Management

Table 26 defines the “oic.wk.mnt” Resource Type. At least one of the Factory\_Reset, and Reboot properties shall be implemented.

**Table 26. “oic.wk.mnt” Resource Type definition**

Property title	Property name	Value type	Value rule	Unit	Access mode	Mandatory	Description
<b>Factory_Reset</b>	fr	boolean			R, W	no	When writing to this Property: 0 – No action (Default*) 1 – Start Factory Reset After factory reset, this value shall be changed back to the default value (i.e., 0). After factory reset all configuration and state data will be lost. When reading this Property, a value of “1” indicates a pending factory reset, otherwise the value shall be “0” after the factory reset.
<b>Reboot</b>	rb	boolean			R, W	no	When writing to this Property: 0 – No action (Default) 1 – Start Reboot After Reboot, this value shall be changed back to the default value (i.e., 0)

3014

3015 Note: \* - Default indicates the value of this property as soon as the device is rebooted or factory reset

3016

3017 **11.6 Scenes**3018 **11.6.1 Introduction**

3019 Scenes are a mechanism for automating certain operations.

3020 A scene is a static entity that stores a set of defined resource property values for a collection of  
 3021 resources. Scenes provide a mechanism to store a setting over multiple Resources that may be  
 3022 hosted by multiple separate Servers. Scenes, once set up, can be used by multiple Clients to recall  
 3023 a setup.

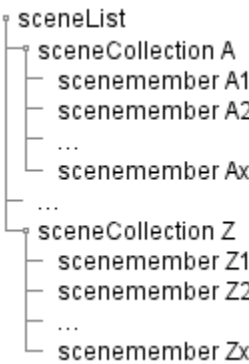
3024 Scenes can be grouped and reused, a group of scenes is also a scene.

3025 In short, scenes are bundled user settings.

3026 **11.6.2 Scenes**3027 **11.6.2.1 Introduction**

3028 Scenes are described by means of resources. The scene resources are hosted by a Server and  
 3029 the top level resource is listed in “/oic/res”. This means that a Client can determine if the scene  
 3030 functionality is hosted on a Server via a RETRIEVE on “/oic/res” or via Resource discovery. The  
 3031 setup of scenes is driven by Client interactions. This includes creating new scenes, and mappings  
 3032 of Server resource properties that are part of a scene.

3033 The scene functionality is created by multiple resources and has the structure depicted in Figure  
3034 33. The sceneList and sceneCollection resources are overloaded collection resources. The  
3035 sceneCollection contains a list of scenes. This list contains zero or more scenes. The  
3036 sceneMember resource contains the mapping between a scene and what needs to happen  
3037 according to that scene on an indicated resource.



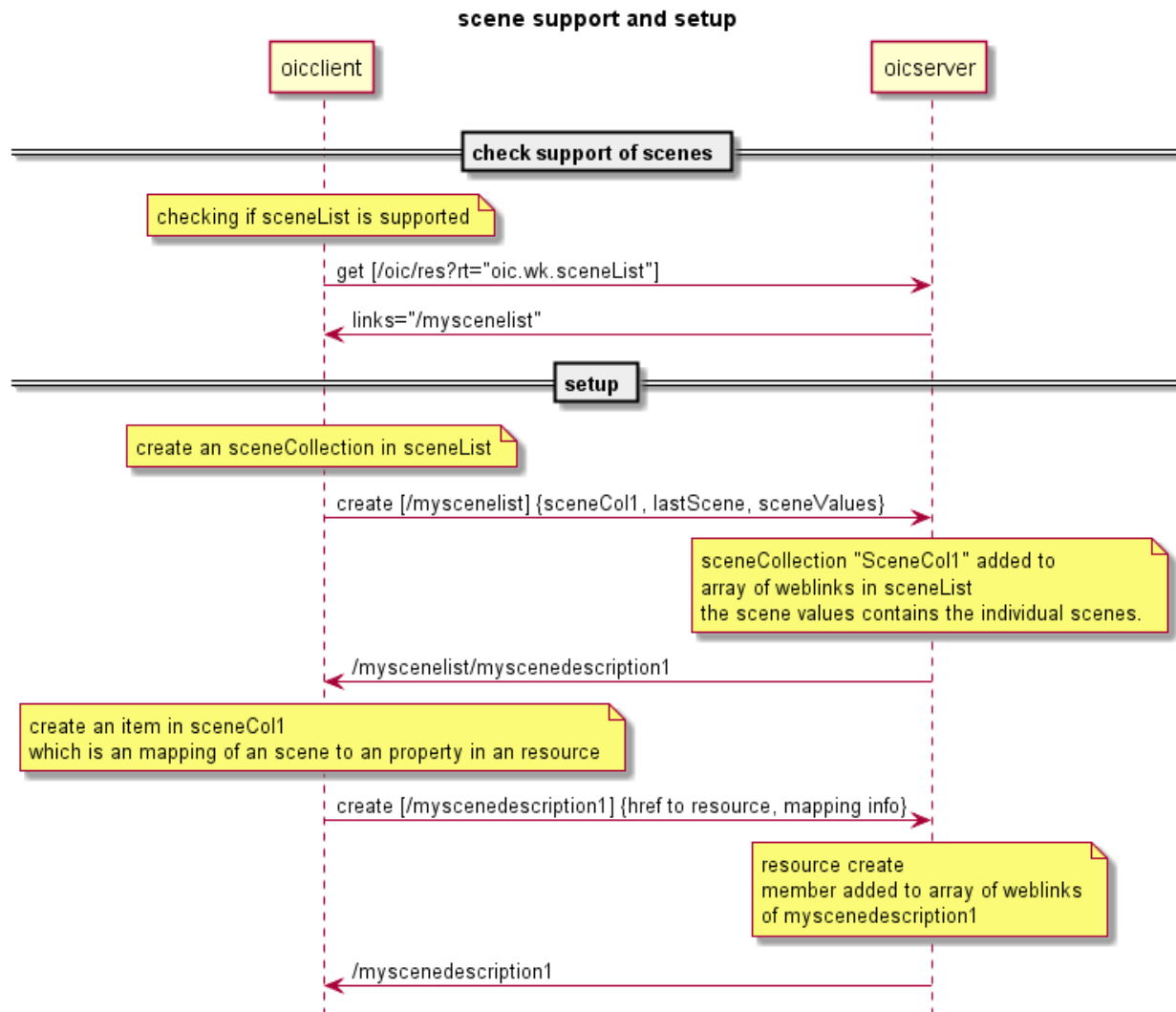
3038

3039

**Figure 33 Generic scene resource structure**

3040 **11.6.2.2 Scene creation**

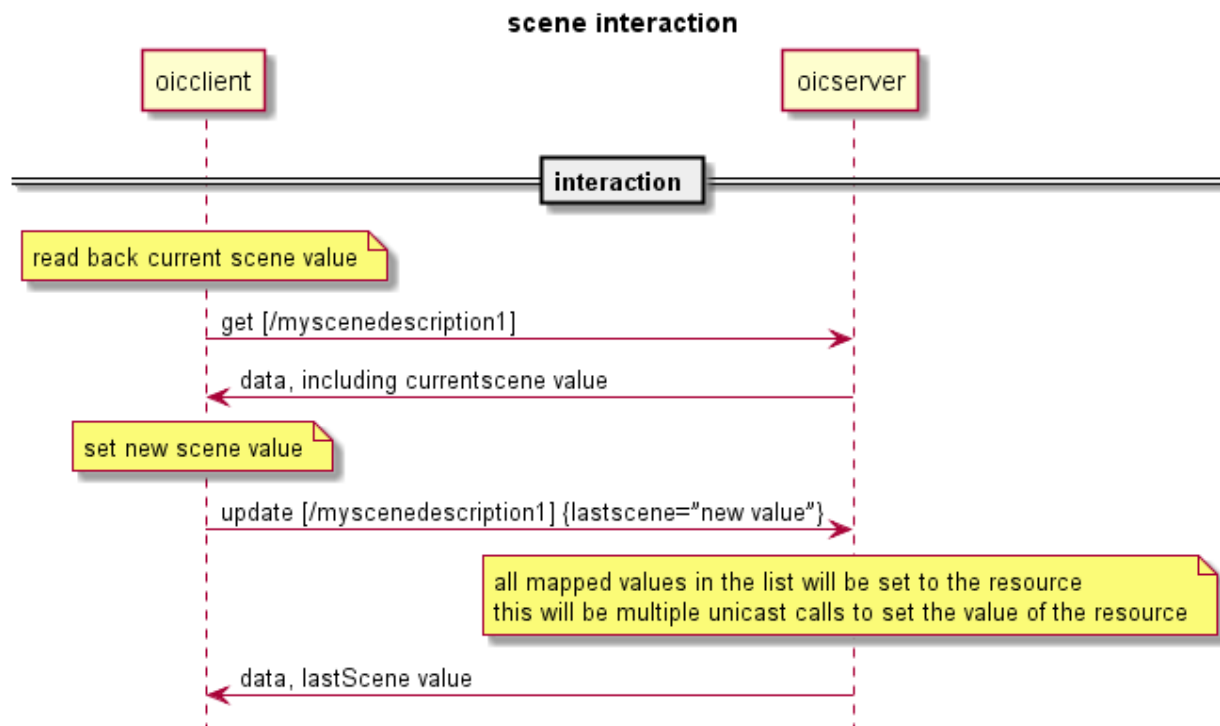
3041 A Client desiring to interact with scenes needs to first determine if the server supports the scene  
3042 feature; the sceneMembers of a scene do not have to be co-located on the server supporting the  
3043 scene feature. This can be done by checking if “/oic/res” contains the rt of the sceneList resource.  
3044 This is depicted in first steps of Figure 34. The sceneCollection is created by the Server using  
3045 some out of bound mechanism, Client creation of scenes is not supported at this time. This will  
3046 entail defining the scene with an applicable list of scene values and the mappings for each  
3047 Resource being part of the scene. The mapping for each resource being part of the sceneCollection  
3048 is described by a resource called sceneMember. The sceneMember resource contains the link to  
3049 a resource and the mapping between the scene listed in the sceneValues property and the actual  
3050 resource property value of the Resource indicated by the link.



**Figure 34 Interactions to check Scene support and setup of specific scenes**

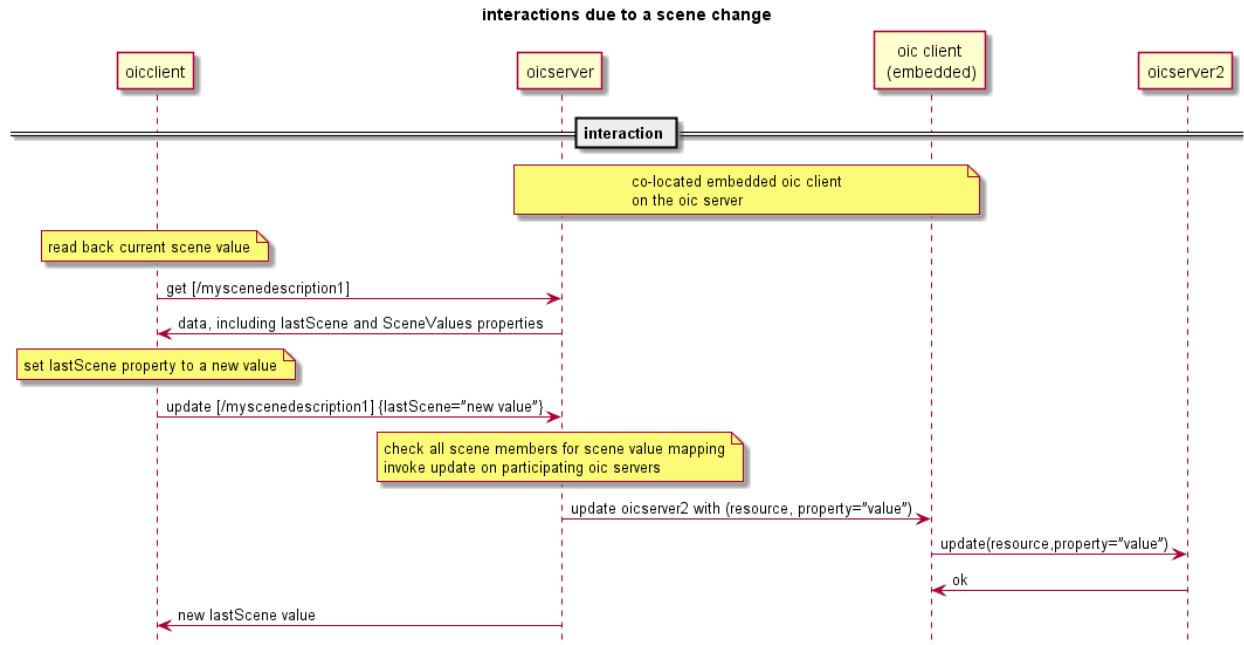
### 11.6.2.3 Interacting with Scenes

All capable Clients can interact with scenes. The allowed scene values and the last applied scene value can be retrieved from the server hosting the scene. The scene value shall be changed by issuing an UPDATE operation with a payload that sets the lastScene property to one of the listed allowed scene values. These steps are depicted in Figure 35. Note that the lastScene value does not imply that the current state of all resources that are part of the scene will be at the mapped value. This is due to that the setting the scene values are not modelled as actual states of the system. This means that another Client can change just one resource being part of the scene without having feedback that the state of the scene is changed.



**Figure 35 Client interactions on a specific scene**

As described previously, a scene can reference one or more resources that are present on one or more Servers. The scene members are re-evaluated each time a scene change takes place. This evaluation is triggered by a Client that is either embedded as part of the Server hosting the scene, or separate to the server having knowledge of the scene via a RETRIEVE operation, observing the referenced resources using the mechanism described in section 11.4.2. During the evaluation the mappings for the new scene value will be applied to the Server. This behaviour is depicted in Figure 36.



**Figure 36 Interaction overview due to a Scene change**

#### 11.6.2.4 Summary of Resource Types defined for Scene functionality

Table 27 summarizes the list of Resource Types that are part of Scenes.

**Table 27 list of Resource Types for Scenes**

Friendly Name (informative)	Resource Type (rt)	Short Description	Section
<b>sceneList</b>	oic.wk.scenelist	Top Level collection containing sceneCollections	
<b>sceneCollection</b>	oic.wk.scenecollection	Description of zero or more scenes	
<b>sceneMember</b>	oic.wk.scenemember	Description of mappings for each specific resource part of the sceneCollection	

#### 11.6.3 Security considerations

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

See OCF Security for details on the use of ACLs and also the mechanisms around Device Authentication that are necessary to ensure that the correct permissions exist for the Client to access the scene member resource(s) on the Server.

### 11.7 Icons

#### 11.7.1 Overview

Icons are a primitive that are needed by various OCF subsystems, such as bridging. An optional Resource Type of “oic.r.icon” has been defined to provide a common representation of an icon Resource that can be used by Devices.

## 11.7.2 Resource

The icon Resource is as defined in Table 28.

**Table 28. Optional Icon Core Resource**

URI	Resource Type Title	Resource Type ID ("rt" value)	Interfaces	Description	Related Functional Interaction
<code>"/example/oic/icon"</code>	Icon	<code>"oic.r.icon"</code>	<code>"oic.if.r"</code>	The Resource through which the Device can obtain icon images. The Resource properties exposed by <code>"/example/oic/mnt"</code> are listed in Table 29.	Icon

Table 29 defines the details for the `"oic.r.icon"` Resource Type.

**Table 29. "oic.r.icon" Resource Type definition**

Property title	Property name	Value type	Value rule	Unit	Access mode	Mandatory	Description
<b>Mime Type</b>	mimetype	string			R	yes	Specifies the format (media type) of the icon. It should be a template string as specified in IANA Media Types Assignment
<b>Width</b>	width	integer	$\geq 1$		R	yes	Width of the icon in pixels greater than or equal to 1.
<b>Height</b>	height	integer	$\geq 1$		R	yes	Height of the icon in pixels greater than or equal to 1.
<b>Icon</b>	media	uri			R	yes	URI to the location of the icon image.

## 11.8 Introspection

### 11.8.1 Overview

Introspection is a mechanism to announce the capabilities of Resources hosted on the Device.

The intended usage of the Introspection Device Data is to enable dynamic clients. E.g. clients that can use the Introspection Device Data to generate dynamically an UI or dynamically create translations of the hosted Resources to another eco-system. Other usages of the Introspection is that the information can be used to generate client code. The Introspection Device Data is designed to augment the existing data already on the wire. This means that existing mechanism needs to be used to get a full overview of what is implemented in the Device. For example the Introspection Device Data does not convey information about observe, since that is already conveyed with the "p" property on the links in `"/oic/res"` (see section 7.8.2.1.2).

The Introspection Device Data is recommended to be conveyed as "static" data. Meaning that the data does not change during the uptime of a Device. However when the data is not static the Introspection Resource shall indicate to be observable and the url property value of `"oic.wk.introspection"` Resource shall change to indicate that the Introspection Device Data is changed.

The Introspection Device Data describes the Resources that make up the Device. For the complete list of included Resources Table 13. The Introspection Device Data is described as a swagger2.0

3115 in JSON format file. The swagger2.0 file will contain the description of the Resources as defined  
3116 below: All Resources with the next remarks:

- 3117 • The URLs of the Resources in the Introspection Device Data shall be without the endpoint  
3118 description, e.g. it shall not be a full URL but only the relative path from the endpoint. The  
3119 relative path shall be the same as being conveyed by “/oic/res”.
- 3120 • “/oic/res” Resource shall not be listed in the Introspection Device Data.
- 3121 • The Resources “/oic/d”, “/oic/p” and the security Resources are allowed to be present in the  
3122 Introspection Device Data, but are not required. The “/oic/d”, “/oic/p”, “/oic/res” and the security  
3123 Resources shall be included when vendor defined or optional properties are implemented.
- 3124 • All other Resources are required to be listed in the Introspection Device Data.
- 3125 • Per Resource it will include:
  - 3126 ○ All Implemented Methods
  - 3127 ○ Per Supported Method:
    - 3128 ▪ Implemented queryParameters per Method.
      - 3129 • This includes the supported interfaces ("if") as enum value.
    - 3130 ▪ Schemas of the payload for the request and response bodies of the Method
    - 3131 ▪ The schema data shall be conveyed by the swagger schema object as  
3132 defined in the parameters section.
    - 3133 ▪ The swagger2.0 schema object shall comply with:
      - 3134 • The schemas shall be fully resolved, e.g. no references shall exist  
3135 outside the swagger file.
      - 3136 • The schemas shall list which interfaces are supported on the method.
      - 3137 • The schemas shall list if a property is optional or required.
      - 3138 • The schemas shall indicate if a property is read only or read-write
        - 3139 ○ By means of the readOnly schema tag belonging to the  
3140 property
      - 3141 • The default value of the “rt” property shall be used to indicate the  
3142 supported Resource Types.
      - 3143 • oneOf and anyOf constructs are allowed to be used as part of an  
3144 swagger2.0 schema object.

3145 Dynamic Resources (e.g. Resources that can be created up on a request by a Client) shall have  
3146 an URL definition which contains a URL identifier (e.g. using the {} syntax). An URL with {} identifies  
3147 that the Resource definition applies to the whole group of Resources that can be created. The  
3148 actual path can contain the collection node that links to the Resource.

3149 Example of an URL with identifiers:

3150 /SceneListResURI/{SceneCollectionResURI}/{SceneMemberResURI}:

3151 When different Resource Types are allowed to be created in a collection, then the different  
3152 schemas for the create method shall define all possible Resource Types that can be created. The  
3153 schema construct oneOf allows the definition of a schema with selectable Resources. The oneOf  
3154 construct allows the integration of all schemas and that only one existing sub schemas shall be  
3155 used to indicate the definition of the Resource that can be created.

3156 Example usage of oneOf JSON schema construct:



```

3157 {
3158   "oneOf":[
3159     { <<subschema 1 definition>> },
3160     { <<sub schema 2 definition >> }
3161   ...
3162   ]
3163 }

```

3164

3165 A Client using the Introspection Device Data of a Device should check the version of the supported  
 3166 Introspection Device Data of the Device. The swagger version is indicated in each file with the tag  
 3167 "swagger". Example of the 2.0 supported version of the tag is: "swagger": "2.0". Later versions of  
 3168 the spec may reference newer versions of the swagger specification, for example 3.0.

3169 A Server shall support one Resource with a Resource Type of “oic.wk.introspection” as defined in  
 3170 Table 30. The Resource with a Resource Type of “oic.wk.introspection” shall be included in the  
 3171 Resource “/oic/res”.

3172 **Table 30. Introspection Resource**

Pre-defined URI	Resource Type Title	Resource Type ID (“rt” value)	Interfaces	Description	Related Functional Interaction
none	Introspection	oic.wk.introspection	“oic.if.r”	The Resource that announces the URL of the Introspection file.	Introspection

3173

3174 Table 31 defines “oic.wk.introspection” Resource Type.

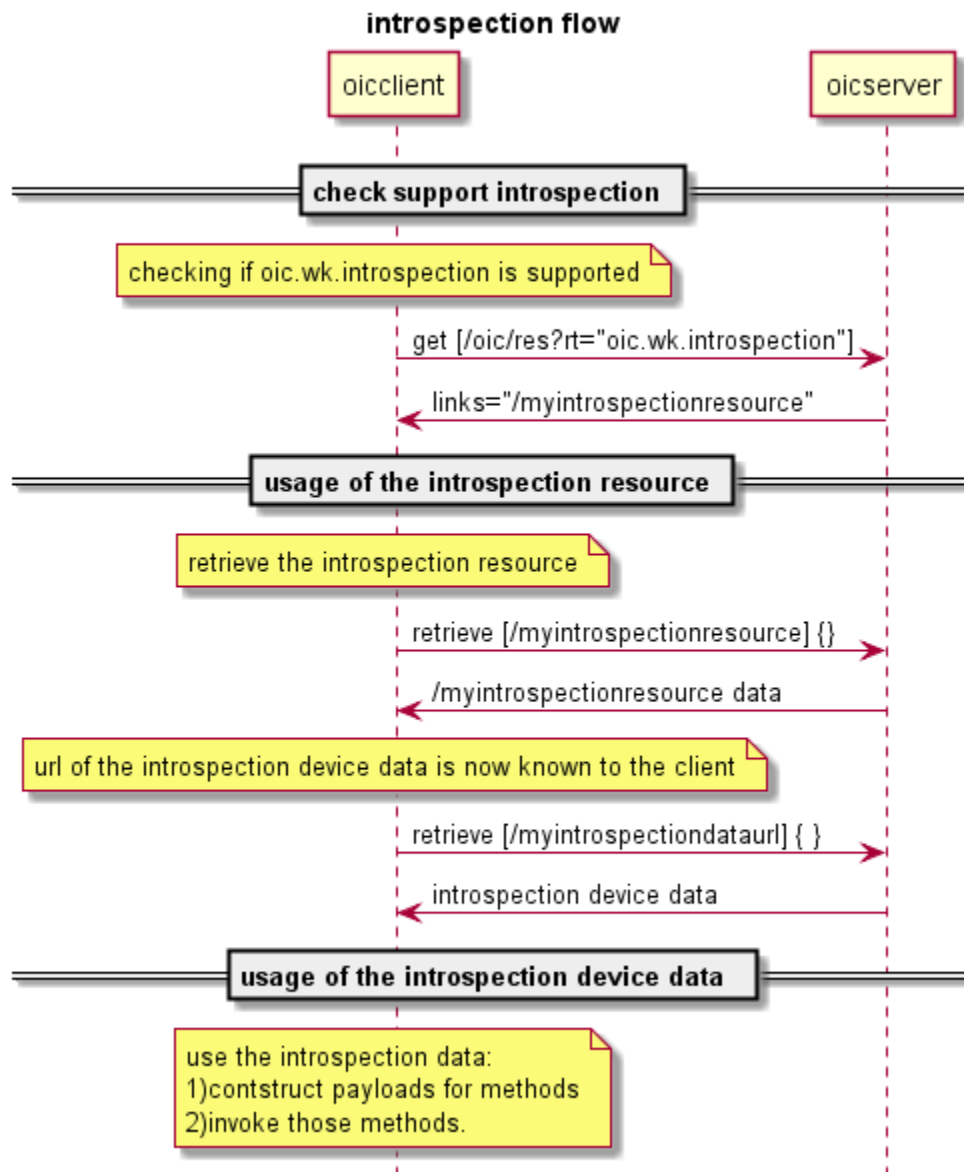
3175 **Table 31. “oic.wk.introspection” Resource Type definition**

Property title	Property name	Value type	Value rule	Unit	Access mode	Mandatory	Description
urlInfo	urlInfo	array			R	yes	array of objects
url	url	string	uri		R	yes	URL to the hosted payload
protocol	protocol	string	enum		R	yes	Protocol definition to retrieve the Introspection Device Data from the url.
content-type	content-type	string	enum		R	no	content type of the url.
version	version	integer	enum		R	no	Version of the Introspection protocol, indicates which rules are applied on the Introspection Device Data regarding the content of the RAML file. Current value is 1.

## 3176 11.8.2 Usage of introspection

3177 The Introspection Device Data is retrieved in the following steps:

- 3178 1) Check if the Introspection Resource is supported and retrieve the URL of the Resource.  
 3179 2) Retrieve the contents of the Introspection Resource  
 3180 3) Download the Introspection Device Data from the URL specified the Introspection Resource.  
 3181 4) Usage of the Introspection Device Data by the Client



3182  
 3183 **Figure 37 Interactions to check Introspection support and download the Introspection**  
 3184 **Device Data.**

## 3185 12 Messaging

### 3186 12.1 Introduction

3187 This section specifies the protocol messaging mapping to the CRUDN messaging operations  
 3188 (Section 8) for each messaging protocol specified (e.g., CoAP.). Mapping to additional protocols  
 3189 is expected in later version of this specification. All the property information from the resource

model shall be carried within the message payload. This payload shall be generated in the resource model layer and shall be encapsulated in the data connectivity layer. The message header shall only be used to describe the message payload (e.g., verb, mime-type, message payload format), in addition to the mandatory header fields defined in messaging protocol (e.g., CoAP) specification. If the message header does not support this, then this information shall also be carried in the message payload. Resource model information shall not be included in the message header structure unless the message header field is mandatory in the messaging protocol specification.

## 12.2 Mapping of CRUDN to CoAP

### 12.2.1 Overview

A Device implementing CoAP shall conform to IETF RFC 7252 for the methods specified in section 12.2.3. A Device implementing CoAP shall conform to IETF RFC 7641 to implement the CoAP Observe option. Support for CoAP block transfer when the payload is larger than the MTU is defined in section 12.2.8.

### 12.2.2 URIs

An OCF: URI is mapped to a coap: URI by replacing the scheme name 'oic' with 'coap' if unsecure or 'coaps' if secure before sending over the network by the requestor. Similarly on the receiver side, the scheme name is replaced with 'oic'.

### 12.2.3 CoAP method with request and response

#### 12.2.3.1 Overview

Every request has a CoAP method that realizes the request. The primary methods and their meanings are shown in Table 32, which provides the mapping of GET/PUT/POST/DELETE methods to CREATE, RETRIEVE, UPDATE, and DELETE operations. The associated text provides the generic behaviours when using these methods, however resource interfaces may modify these generic semantics.

**Table 32. CoAP request and response**

Method for CRUDN	(mandatory) Request data	(mandatory) Response data
<b>GET for RETRIEVE</b>	<ul style="list-style-type: none"> <li>- <b>Method code:</b> GET (0.01)</li> <li>- <b>Request URI:</b> an existing URI for the Resource to be retrieved</li> </ul>	<ul style="list-style-type: none"> <li>- <b>Response code:</b> success (2.xx) or error (4.xx or 5.xx)</li> <li>- <b>Payload:</b> Resource representation of the target Resource (when successful)</li> </ul>
<b>POST for CREATE</b>	<ul style="list-style-type: none"> <li>- <b>Method code:</b> POST (0.02)</li> <li>- <b>Request URI:</b> an existing URI for the Resource responsible for the creation</li> <li>- <b>Payload:</b> Resource presentation of the Resource to be created</li> </ul>	<ul style="list-style-type: none"> <li>- <b>Response code:</b> success (2.xx) or error (4.xx or 5.xx)</li> <li>- <b>Payload:</b> the URI of the newly created Resource (when successful).</li> </ul>
<b>PUT for CREATE</b>	<ul style="list-style-type: none"> <li>- <b>Method code:</b> PUT (0.03)</li> <li>- <b>Request URI:</b> a new URI for the Resource to be created.</li> <li>- <b>Payload:</b> Resource presentation of the Resource to be created.</li> </ul>	<ul style="list-style-type: none"> <li>- <b>Response code:</b> success (2.xx) or error (4.xx or 5.xx)</li> </ul>
<b>POST for UPDATE</b>	<ul style="list-style-type: none"> <li>- <b>Method code:</b> POST (0.02)</li> <li>- <b>Request URI:</b> an existing URI for the Resource to be updated.</li> <li>- <b>Payload:</b> representation of the Resource to be updated.</li> </ul>	<ul style="list-style-type: none"> <li>- <b>Response Code:</b> success (2.xx) or error (4.xx or 5.xx)</li> </ul>

<b>DELETE for DELETE</b>	<b>- Method code:</b> DELETE (0.04) <b>- Request URI:</b> an existing URI for the Resource to be deleted.	<b>- Response code:</b> success (2.xx) or error (4.xx or 5.xx)
--------------------------	--	--

3216

## 3217 **12.2.3.2 CREATE with POST or PUT**

### 3218 **12.2.3.2.1 With POST**

3219 POST shall be used only in situations where the request URI is valid, that is it is the URI of an  
3220 existing Resource on the Server that is processing the request. If no such Resource is present,  
3221 the Server shall respond with an error response code of 4.xx. The use of POST for CREATE shall  
3222 use an existing request URI which identifies the Resource on the Server responsible for creation.  
3223 The URI of the created Resource is determined by the Server and provided to the Client in the  
3224 response.

3225 A Client shall include the representation of the new Resource in the request payload. The new  
3226 resource representation in the payload shall have all the necessary properties to create a valid  
3227 Resource instance, i.e. the created Resource should be able to properly respond to the valid  
3228 Request with mandatory Interface (e.g., "GET with ?if=oic.if.baseline").

3229 Upon receiving the POST request, the Server shall either

- 3230 • create the new Resource with a new URI, respond with the new URI for the newly created  
3231 Resource and a success response code (2.xx); or
- 3232 • respond with an error response code (4.xx or 5.xx).

3233 POST is unsafe and is the supported method when idempotent behaviour cannot be expected or  
3234 guaranteed.

### 3235 **12.2.3.2.2 With PUT**

3236 PUT shall be used to create a new Resource or completely replace the entire representation of an  
3237 existing Resource. The resource representation in the payload of the PUT request shall be the  
3238 complete representation. PUT for CREATE shall use a new request URI identifying the new  
3239 Resource to be created.

3240 The new resource representation in the payload shall have all the necessary properties to create  
3241 a valid Resource instance, i.e. the created Resource should be able to properly respond to the  
3242 valid Request with mandatory Interface (e.g. "GET with ?if=oic.if.baseline").

3243 Upon receiving the PUT request, the Server shall either

- 3244 • create the new Resource with the request URI provided in the PUT request and send back a  
3245 response with a success response code (2.xx); or
- 3246 • respond with an error response code (4.xx or 5.xx).

3247 PUT is an unsafe method but it is idempotent, thus when a PUT request is repeated the outcome  
3248 is the same each time.

## 3249 **12.2.3.3 RETRIEVE with GET**

3250 GET shall be used for the RETRIEVE operation. The GET method retrieves the representation of  
3251 the target Resource identified by the request URI.

3252 Upon receiving the GET request, the Server shall either

- 3253 • send back the response with the representation of the target Resource with a success response  
3254 code (2.xx); or

3255 • respond with an error response code (4.xx or 5.xx) or ignore it (e.g. non-applicable multicast  
3256 GET).

3257 GET is a safe method and is idempotent.

3258 **12.2.3.4 UPDATE with POST**

3259 POST shall be used only in situations where the request URI is valid, that is it is the URI of an  
3260 existing Resource on the Server that is processing the request. If no such Resource is present,  
3261 the Server shall respond with an error response code of 4.xx. A client shall use POST to UPDATE  
3262 Property values of an existing Resource (see Sections 3.1.32 and 8.4.2).

3263 Upon receiving the request, the Server shall either

- 3264 • apply the request to the Resource identified by the request URI in accordance with the applied  
3265 interface (i.e. POST for non-existent Properties is ignored) and send back a response with a  
3266 success response code (2.xx); or
- 3267 • respond with an error response code (4.xx or 5.xx). Note that if the representation in the  
3268 payload is incompatible with the target Resource for POST using the applied interface (i.e. the  
3269 "overwrite" semantic cannot be honored because of read-only property in the payload), then  
3270 the error response code 4.xx shall be returned.

3271 POST is unsafe and is the supported method when idempotent behaviour cannot be expected or  
3272 guaranteed.

3273 **12.2.3.5 DELETE with DELETE**

3274 DELETE shall be used for DELETE operation. The DELETE method requests that the resource  
3275 identified by the request URI be deleted.

3276 Upon receiving the DELETE request, the Server shall either

- 3277 • delete the target Resource and send back a response with a success response code (2.xx); or
- 3278 • respond with an error response code (4.xx or 5.xx).

3279 DELETE is unsafe but idempotent (unless URIs are recycled for new instances).

3280  
3281

3282 **12.2.4 Content-Format negotiation**

3283 The OCF Framework mandates support of CBOR, however it allows for negotiation of the payload  
3284 body if more than one Content-Format (e.g. CBOR and JSON) is supported by an implementation.  
3285 In this case the Accept Option defined in section 5.10.4 of IETF RFC 7252 shall be used to indicate  
3286 which Content-Format (e.g. JSON) is requested by the Client.

3287 The Content-Formats supported are shown in Table 33.

3288

**Table 33. OCF Content-Formats**

Media Type	ID
"application/cbor"	60
"application/vnd.ocf+cbor"	10000

3289 Clients shall include a Content-Format Option in every message that contains a payload. Servers  
3290 shall include a Content-Format Option for all success (2.xx) responses with a payload body. Per  
3291 IETF RFC 7252 section 5.5.1, Servers shall include a Content-Format Option for all error (4.xx or

5.xx) responses with a payload body unless they include a Diagnostic Payload; error responses with a Diagnostic Payload do not include a Content-Format Option. The Content-Format Option shall use the ID column numeric value from Table 33. An OCF vertical may mandate a specific Content-Format Option.

Clients shall also include an Accept Option in every request message. The Accept Option shall indicate the required Content-Format as defined in Table 33 for response messages. The Server shall return the required Content-Format if available. If the required Content-Format cannot be returned, then the Server shall respond with an appropriate error message.

## 12.2.5 OCF-Content-Format-Version information

Servers and Clients shall include the OCF-Content-Format-Version in both request and response messages with a payload. Clients shall include the OCF-Accept-Content-Format-Version in request messages. The OCF-Content-Format-Version and OCF-Accept-Content-Format-Version are specified as Option Numbers in the CoAP header as shown in Table 34.

**Table 34. OCF-Content-Format-Version and OCF-Accept-Content-Format-Version Option Numbers**

CoAP Option Number	Name	Format	Length (bytes)
2049	OCF-Accept-Content-Format-Version	uint	2
2053	OCF-Content-Format-Version	uint	2

The value of the OCF-Accept-Content-Format-Version and the OCF-Content-Format-Version is a two-byte unsigned integer that is used to define the major, minor and sub versions. The major and minor versions are represented by 5 bits and the sub version is represented by 6 bits as shown in Table 35.

**Table 35. OCF-Accept-Content-Format-Version and the OCF-Content-Format-Version Representation**

Major Version						Minor Version						Sub Version					
Bit	15	14	13	12	11	10	9	8	7	6		5	4	3	2	1	0

Table 36 illustrates several examples:

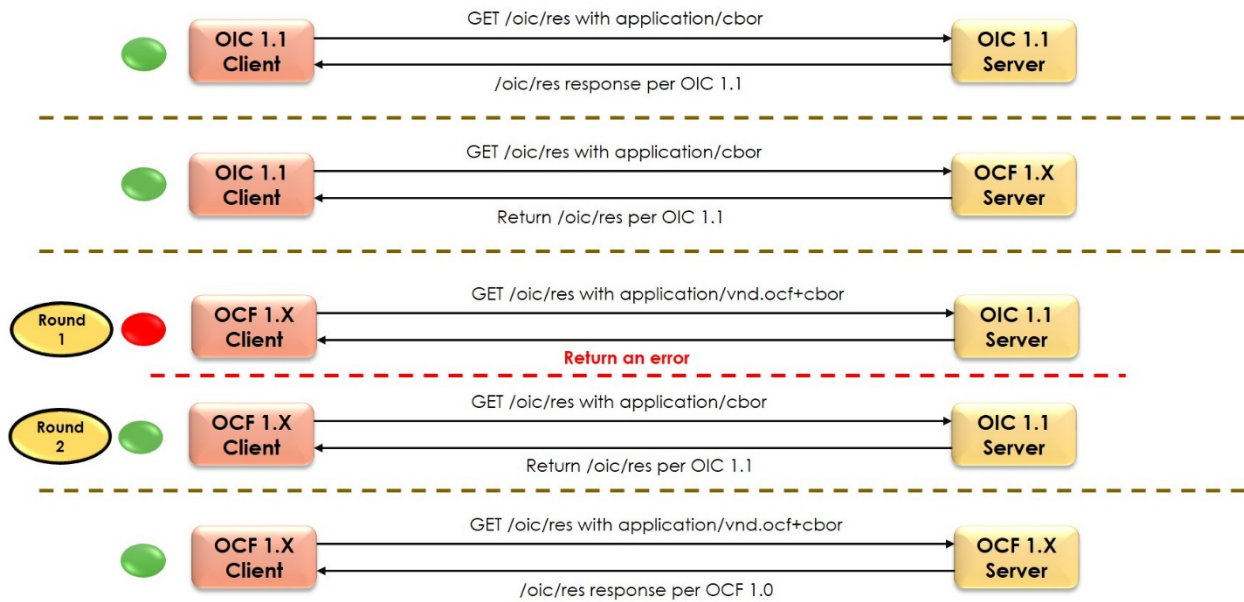
**Table 36. Examples of OCF-Content-Format-Version and OCF-Accept-Content-Format-Version Representation**

OCF version	Binary representation	Integer value
1.0.0	0000 1000 0000 0000	2048
1.1.0	0000 1000 0100 0000	2112

The OCF-Accept-Content-Format-Version and OCF-Content-Format-Version for this version of the specification shall be 1.0.0 (i.e. 0b0000 1000 0000 0000).

## 12.2.6 Content-Format policy

To maintain compatibility between devices implemented to different versions of this specification, Devices shall follow the policy as described in Figure 38.



**Figure 38 Content-Format Policy**

All Devices shall support the current and all previous Content-Format Option and Versions. Clients shall send discovery request messages with the current and all previous Content-Format and Versions until it discovers all Servers in the network.

### 12.2.7 CRUDN to CoAP response codes

The mapping of CRUDN operations response codes to CoAP response codes are identical to the response codes defined in IETF RFC 7252.

### 12.2.8 CoAP block transfer

Basic CoAP messages work well for the small payloads typical of light-weight, constrained IoT devices. However scenarios can be envisioned in which an application needs to transfer larger payloads.

CoAP block-wise transfer as defined in <https://tools.ietf.org/html/rfc7721>

IETF RFC 7959 shall be used by all Servers which generate a content payload that would exceed the size of a CoAP datagram as the result of handling any defined CRUDN operation.

Similarly, CoAP block-wise transfer as defined in <https://tools.ietf.org/html/rfc7721>

IETF RFC 7959 shall be supported by all Clients. The use of block-wise transfer is applied to both the reception of payloads as well as transmission of payloads that would exceed the size of a CoAP datagram.

All blocks that are sent using this mechanism for a single instance of a transfer shall all have the same reliability setting (i.e. all confirmable or all non-confirmable).

A Client may support both the block1 (as descriptive) and block2 (as control) options as described by IETF RFC 7959 A Server may support both the block1 (as control) and block2 (as descriptive) options as described by <https://tools.ietf.org/html/rfc7721>

IETF RFC 7959.

## 12.3 CoAP serialization over TCP

### 12.3.1.1 Introduction

In environments where TCP is already available, CoAP can take advantage of it to provide reliability. Also in some environments UDP traffic is blocked, so deployments may use TCP. For example, consider a cloud application acting as a Client and the Server is located at the user's home. The Server which already support CoAP as a messaging protocol (e.g., Smart Home vertical profile) could easily support CoAP serialization over TCP rather than adding another messaging protocol. A Device implementing CoAP Serialization over TCP should conform to IETF draft-ietf-core-coap-tcp-tls-07.

### 12.3.1.2 Indication of support

If UDP is blocked, clients depend on the pre-configured details on the device to find support for CoAP over TCP. If UDP is not-blocked, a Device which supports CoAP serialization over TCP shall populate the Messaging Protocol ("mpro") property in "/oic/res" with the value "coap+tcp" or "coaps+tcp" to indicate that the device supports messaging protocol as specified by section 11.3.4.

### 12.3.1.3 Message type and header

The message type transported between Client and Server shall be a non-confirmable message (NON). The protocol stack used in this scenario should be as described in section 3 in IETF draft-ietf-core-coap-tcp-tls-07.

The CoAP header as described in figure 6 in IETF draft-ietf-core-coap-tcp-tls-07 should be used for messages transmitted between a Client and a Server. A Device should use "Alternative L3" as defined in IETF draft-ietf-core-coap-tcp-tls-07.

### 12.3.1.4 URI scheme

The URI scheme used shall be as defined in section 6 in IETF draft-ietf-core-coap-tcp-tls-07.

For the "coaps+tcp" URI scheme the "TLS Application Layer Protocol Negotiation Extension" IETF RFC 7301 shall be used.

### 12.3.1.5 KeepAlive

#### 12.3.1.5.1 Overview

In order to ensure that the connection between a Device is maintained, when using CoAP serialization over TCP, a Device that initiated the connection should send application layer KeepAlive messages. The reasons to support application layer KeepAlive are as follows:

- TCP KeepAlive only guarantees that a connection is alive at the network layer, but not at the application layer
- Interval of TCP KeepAlive is configurable only using kernel parameters, and is OS dependent (e.g., 2 hours by default in Linux)

#### 12.3.1.5.2 KeepAlive Mechanism

Devices supporting CoAP over TCP shall use the following KeepAlive mechanism. A Server shall support the "oic.wk.ping" Resource Type as defined in Table 37.

**Table 37. Ping resource**

Pre-defined URI	Resource Type Title	Resource Type ID ("rt" value)	Interfaces	Description	Related Functional Interaction
/oic/ping	Ping	oic.wk.ping	"oic.if.rw"	The resource using which a Client keeps its Connection with a Server active.	KeepAlive



				The resource properties exposed by "/oic/ping" are listed in Table 38.	
--	--	--	--	--	--

Table 38 defines "oic.wk.ping" Resource Type.

**Table 38. "oic.wk.ping" Resource Type definition**

Property title	Property name	Value type	Value rule	Unit	Access mode	Mandatory	Description
Interval	in	integer	minutes		R,W	yes	The time interval for which connection shall be kept alive and not closed. Default value is 0.

The following steps detail the KeepAlive mechanisms for a Client and Server:

- 1) A Client which wants to keep the connection with a Server alive shall send a POST request to "/oic/ping" resource on the Server updating its connection Interval.
  - a. This time interval shall start from 2 minutes and increases in multiples of 2 up to a maximum of 64 minutes. It stays at 64 minutes from that point.
- 5) A Server receiving this ping request shall respond within 1 minute.
- 6) If a Client does not receive the response within 1 minute, it shall terminate the connection.
- 7) If a Server does not receive a POST request to ping resource within the specified "interval" time, the Server shall terminate the connection.

An example of the KeepAlive mechanism is as follows:

- Client → Server: "POST/oic/ping {interval: 2}"
- Server → Client: 2.03 valid

## 12.4 Payload Encoding in CBOR

OCF implementations shall perform the conversion to CBOR from JSON defined schemas and to JSON from CBOR in accordance with IETF RFC 7049 section 4 unless otherwise specified in this section.

Properties defined as a JSON integer shall be encoded in CBOR as an integer (CBOR major types 0 and 1). Properties defined as a JSON number shall be encoded as an integer, single- or double-precision floating point (CBOR major type 7, sub-types 26 and 27); the choice is implementation dependent. Half-precision floating point (CBOR major 7, sub-type 25) shall not be used. Integer numbers shall be within the closed interval  $[-2^{53}, 2^{53}]$ . Properties defined as a JSON number should be encoded as integers whenever possible; if this is not possible Properties defined as a JSON number should use single-precision if the loss of precision does not affect the quality of service, otherwise the Property shall use double-precision.

On receipt of a CBOR payload, an implementation shall be able to interpret CBOR integer values in any position. If a property defined as a JSON integer is received encoded other than as an integer, the implementation may reject this encoding using a final response as appropriate for the underlying transport (e.g. 4.00 for CoAP) and thus optimise for the integer case. If a property is defined as a JSON number an implementation shall accept integers, single- and double-precision floating point.

## 13 Security

The details for handling security and privacy are specified in [OCF Security].



## Annex A (informative)

### Operation Examples

#### A.1 Introduction

This section describes some example scenarios using sequence of operations between the entities involved. In all the examples below “Light” is a Server and “Smartphone” is a Client. In one of the scenario “Garage” additionally acts as a Server. All the examples are based on the following example resource definitions:

rt=oic.example.light with Resource Type definition as illustration in Table 39.

**Table 39. oic.example.light Resource Type definition**

Property title	Property name	Value type	Value rule	Unit	Access mode	Mandatory	Description
<b>Name</b>	n	string			R, W	no	
<b>on-off</b>	of	boolean			R, W	yes	On/Off Control: 0 = Off 1 = On
<b>dim</b>	dm	integer	0-255		R, W	yes	Resource which can take a range of values minimum being 0 and maximum being 255

rt=oic.example.garagedoor with Resource Type definition as illustration in Table 40.

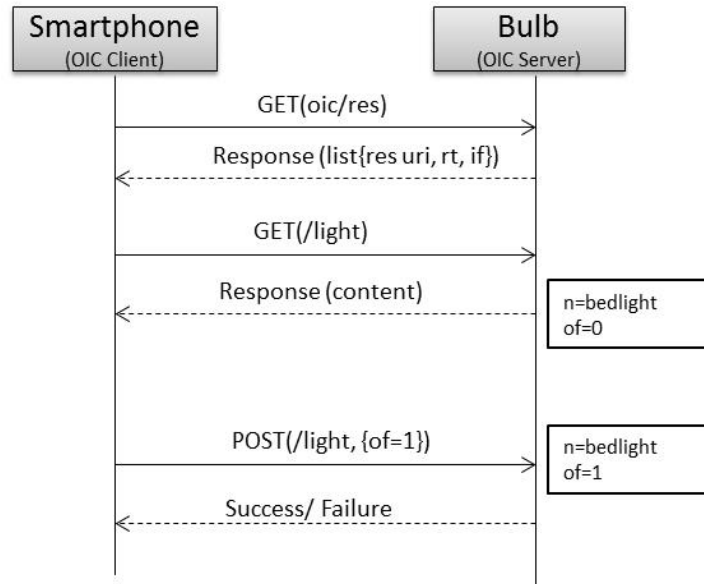
**Table 40. oic.example.garagedoor Resource Type definition**

Property title	Property name	Value type	Value rule	Unit	Access mode	Mandatory	Description
<b>Name</b>	n	string			R, W	no	
<b>open-close</b>	oc	boolean			R, W	yes	Open/Close Control: 0 = Open 1 = Close

“/oic/mnt” (“rt=oic.wk.mnt”) used in below examples is defined in section 11.5.2.

#### A.2 When at home: From smartphone turn on a single light

This sequence highlights (Figure 39) the discovery and control of an OCF light resource from an OCF smartphone.



**Figure 39. When at home: from smartphone turn on a single light**

Discovery request can be sent to “All OCF Nodes” Multicast address FF0X::158 or can be sent directly to the IP address of device hosting the light resource.

- 1) Smartphone sends a GET request to “/oic/res” resource to discover all resources hosted on targeted endpoint
- 2) The endpoint (bulb) responds with the list of Resource URI, Resource Type and Interfaces supported on the endpoint (one of the resource is ‘/light’ whose rt=oic.example.light)
- 3) Smartphone sends a GET request to ‘/light’ resource to know its current state
- 4) The endpoint responds with representation of light resource ({n=bedlight;of=0})
- 5) Smartphone changes the ‘of’ property of the light resource by sending a POST request to ‘/light’ resource ({of=1})
- 6) On Successful execution of the request, the endpoint responds with the changed resource representation. Else, error code is returned. Details of the error codes are defined in section 12.2.7.

### A.3 GroupAction execution

This example will be added when groups feature is added in later version of specification

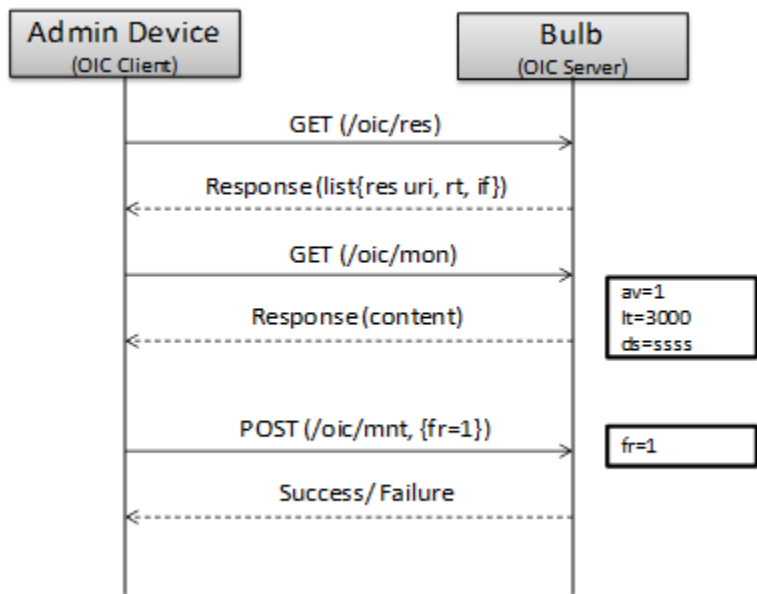
### A.4 When garage door opens, turn on lights in hall; also notify smartphone

This example will be added when scripts feature is added in later version of specification

### A.5 Device management

This sequence highlights (Figure 40) the device management function of maintenance.

3462



3463

3464

**Figure 40. Device management (maintenance)**

3465

**Pre-Condition:** Admin device has different security permissions and hence can perform device management operations on the Device

3466

3467

1) Admin device sends a GET request to “/oic/res” resource to discover all resources hosted on a targeted end point (in this case Bulb)

3468

3469

2) The end point (bulb) responds with the list of Resource URI, Resource Type and Interfaces supported on the end point (one of the resources is “/oic/mnt” whose “rt=oic.wk.mnt”)

3470

3471

3) Admin Device changes the ‘fr’ property of the maintenance resource by sending a POST request to “/oic/mnt” resource ({fr=1}). This triggers a factory reset of the end point (bulb)

3472

3473

4) On successful execution of the request, the end point responds with the changed resource representation. Else, error code is returned. Details of the error codes are defined in section 12.2.7.

3474

3475

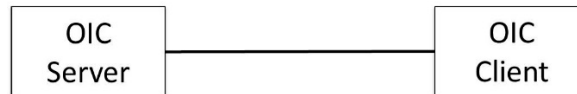
3476 **Annex B**  
3477 (informative)

3478 **OCF interaction scenarios and deployment models**  
3479

3480 **B.1 OCF interaction scenarios**

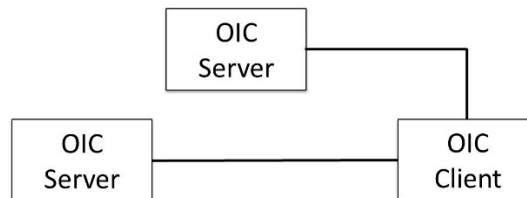
3481 A Client connects to one or multiple Servers in order to access the resources provided by those  
3482 Servers. The following are scenarios representing possible interactions among Roles:

- 3483 • Direct interaction between Client and Server (Figure 41). In this scenario the Client and the  
3484 Server directly communicate without involvement of any other Device. A smartphone which  
3485 controls an actuator directly uses this scenario.



3486  
3487 **Figure 41. Direct interaction between Server and Client**

- 3488 • Interaction between Client and Server using another server (Figure 42). In this scenario,  
3489 another Server provides the support needed for the Client to directly access the desired  
3490 resource on a specific Server. This scenario is used for example, when a smartphone first  
3491 accesses a discovery server to find the addressing information of a specific appliance, and  
3492 then directly accesses the appliance to control it.



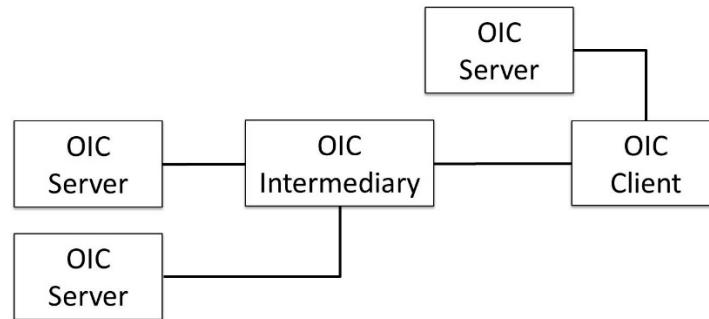
3493  
3494 **Figure 42. Interaction between Client and Server using another Server**

- 3495 • Interaction between Client and Server using Intermediary (Figure 43). In this scenario an  
3496 Intermediary facilitates the interaction between the Client and the Server. A smartphone which  
3497 controls appliances in a smart home via MQTT broker uses this scenario.



3498  
3499 **Figure 43. Interaction between Client and Server using Intermediary**

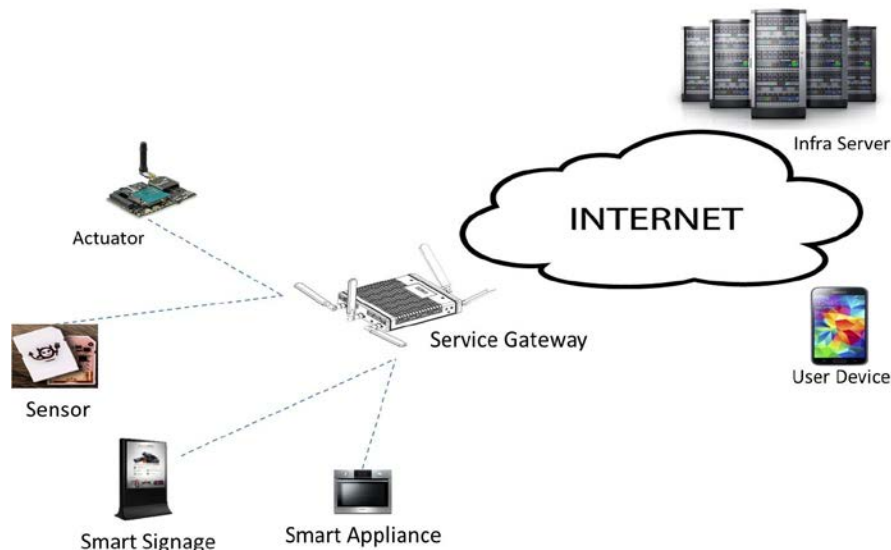
- 3500 • Interaction between Client and Server using support from multiple Servers and intermediary  
3501 (Figure 44). In this scenario, both Server and Intermediary roles are present to facilitate the  
3502 transaction between the Client and a specific Server. An example scenario is when a  
3503 smartphone first accesses a Resource Directory (RD) server to find the address to a specific  
3504 appliance, then utilizes MQTT broker to deliver a command message to the appliance. The  
3505 smartphone can utilize the mechanisms defined in CoRE Resource Directory such as default  
3506 location, anycast address or DHCP to discover the Resource Directory information.



**Figure 44. Interaction between Client and Server using support from multiple Servers and Intermediary**

## B.2 Deployment model

In deployment, Devices are deployed and interact via either wired or wireless connections. Devices are the physical entities that may host resources and play one or more Roles. There is no constraint on the structure of a deployment or number of Devices in it. Architecture is flexible and scalable and capable of addressing large number of devices with different device capabilities, including constrained devices which have limited memory and capabilities. Constrained devices are defined and categorized in [TCNN].



**Figure 45. Example of Devices**

Figure 45 depicts a typical deployment and set of Devices, which may be divided in the following categories:

- **Things:** Networked devices which are able to interface with physical environments. Things are the devices which are primarily controlled and monitored. Examples include smart appliances, sensors, and actuators. Things mostly take the role of Server but they may also take the role of Client, for example in machine-to-machine communications.
- **User Devices:** Devices employed by the users enabling the users to access resources and services. Examples include smart phones, tablets, and wearable devices. User Devices mainly take the role of Client, but may also take the role of Server or Intermediary.

- 3528 • **Service Gateways:** Network equipment which take the role of Intermediary. Examples are  
3529 home gateways.
- 3530 • **Infra Servers:** Data centers residing in cloud infrastructure, which facilitate the interaction  
3531 among Devices by providing network services such as AAA, NAT traversal or discovery. It can  
3532 also play the role of Client or Intermediary



## Annex C (informative)

### Other Resource Models and OCF Mapping

#### C.1 Multiple resource models

RESTful interactions are defined dependent on the resource model; hence, Devices require a common understanding of the resource model for interoperability.

There are multiple resource models defined by different organizations including OCF, IPSO Alliance and oneM2M, and used in the industry, which may restrict interoperability among respective ecosystems. The main differences from Resource model are as follows:

- **Resource structure:** Resources may be defined to have properties (e.g., oneM2M defined resources), or may be defined as an atomic entity and not be decomposable into properties (e.g., IPSO alliance defined resources). For example, a smart light may be represented as a resource with an on-off property or a resource collection containing an on-off resource. In the former, on-off property doesn't have a URI of its own and can only be accessed indirectly via the resource. In the latter, being a resource itself, on-off resource is assigned its own URI and can be directly manipulated.
- **Resource name & type:** Resources may be allowed to be named freely and have their characteristics indicated using a Resource Type property (e.g., as defined in oneM2M). Alternatively, the name of resources may be defined a priori in a way that the name by itself is indicative of its characteristic (e.g., as defined by IPSO alliance). For example, in oneM2M resource model, a smart light can be named with no restrictions, such as 'LivingRoomLight\_1' but in IPSO alliance resource model it is required to have the fixed Object name with numerical Object ID of "IPSO Light Control (3311)". Consequently, it's likely that in the former case the data path in URI is freely defined and in the latter case it is predetermined.
- **Resource hierarchy:** Resources may be allowed to be organized in hierarchy where a resource contains another resource with a parent-child relationship (e.g., in oneM2M definition of resource model). Resources may also be required to have a flat structure and associate with other resources only by referencing their links.

In addition to the above, different organizations use different syntax and define different features (e.g., resource interface), which preclude interoperability.

#### C.2 OCF approach for support of multiple resource models

In order to expand the IoT ecosystem the Framework takes an inclusive approach for interworking with existing resource models. Specifically, the Framework defines a resource model while providing a mechanism to easily map to other models. By embracing existing resource models OCF is inclusive of existing ecosystems while allowing for the transition toward definition of a comprehensive resource model integrating all ecosystems.

The following OCF characteristics enable support of other resource models:

- **resource model is the superset of multiple models:** the resource model is defined as the superset of existing resource models. In other words, any existing resource model can be mapped to a subset of resource model concepts.
- **Framework may allow for resource model negotiation:** the Client and Server exchange the information about what resource model(s) each supports. Based on the exchanged information, the Client and Server choose a resource model to perform RESTful interactions or to perform translation. This feature is out of scope of the current version of this specification, however, the following is a high level description for resource model negotiation.

### C.3 Resource model indication

The Client and server exchange the information about what resource model(s) each supports. Based on the exchanged information, the Client and Server choose a resource model to perform RESTful interactions or to perform translation. The exchange could be part of discovery and negotiation. Based on the exchange, the Client and Server follow a procedure to ensure interoperability among them. They may choose a common resource model or execute translation between resource models.

- **Resource model schema exchange:** The Client and Server may share the resource model information when they initiate a RESTful interaction. They may exchange the information about which resource model they support as part of session establishment procedures. Alternatively, each request or response message may carry the indication of which resource model it is using. For example, [COAP] defines “Content-Format option” to indicate the “representation format” such as “application/json”. It’s possible to extend the Content-Format Option to indicate the resource model used with the representation format such as “application/ipsoson”.
- **Ensuing procedures:** After the Client and Server exchange the resource model information, they perform a suitable procedure to ensure interoperability among them. The simplest way is to choose a resource model supported by both the Client and Server. In case there is no common resource model, the Client and Server may interact through a 3rd party.

In addition to translation which can be resource intensive, a method based on profiles can be used in which an OCF implementation can accommodate multiple profiles and hence multiple ecosystems.

- **Resource Model Profile:** the Framework defines resource model profiles and implementers or users choose the active profile. The chosen profile constraints the Device to strict rules in how resources are defined, instantiated and interacted with. This would allow for interoperability with devices from the ecosystem identified by the profile (e.g., IPSO, OneM2M etc.). Although this enables a Device to participate in and be part of any given ecosystem, this scheme does not allow for generic interoperability at runtime. While this approach may be suitable for resource constrained devices, more resource capable devices are expected to support more than one profile.

### C.4 An Example Profile (IPSO profile)

IPSO defines smart objects that have specific resources and they take values determined by the data type of that resource. The smart object specification defines a category of such objects. Each resource represents a characteristic of the smart object being modelled.

While the terms may be different, there are equivalent concepts in OCF to represent these terms. This section provides the equivalent OCF terms and then frames the IPSO smart object in OCF terms.

The IPSO object Light Control defined in Section 16 of the IPSO Smart Objects 1.0 is used as the reference example.

#### C.4.1 Conceptual equivalence

The IPSO smart object definition is equivalent to an Resource Type definition which defines the relevant characteristics of an entity being modelled. The specific IPSO Resource is equivalent to a Property that like an IPSO Resource has a defined data type, enumeration of acceptable values, units, a general description and access modes (based on the Interface).

The general method for developing the equivalent Resource Type from an IPSO Smart Object definition is to ignore the Object ID and replace the Object URN with an OCF ‘.’ (dot) separated name that incorporates the IPSO object. Alternatively the Object URN can be used as the Resource

Type ID as is (as long as the URN does not contain any ‘.’ (dots)) – using the same Object URN as the Resource Type ID allows for compatibility when interacting with an IPSO compliant device. The object URN based naming does not have any bearing for OCF to OCF interoperability and so the OCF format is preferred – for OCF to OCF interoperability only the data model consistency is required.

Two models are available to render IPSO objects into OCF.

1) One is where the IPSO Smart Object represents a Resource. In this case, the IP Smart Object is regarded as a resource with the Resource Type matching the description of the Smart Object. Furthermore, each resource in the IPSO definition is represented as a Property in the Resource Type (the IPSO Resource ID is replaced with a string representing the Property). This is the preferred approach when the IPSO Data Model is expressed in the Resource Model.

8) The other approach is to model an IPSO Smart Object as a Collection. Each IPSO Resource is then modelled as a Resource with an Resource Type that matches the definition of the IPSO Resource. Each of these resource instances are then bound to the Collection that represents this IPSO Smart Object.

Below is an example showing how an IPSO LightControl Object is modelled as a Resource.

### Resource Type: Light Control

Description: This Object is used to control a light source, such as a LED or other light. It allows a light to be turned on or off and its dimmer setting to be controlled as a percentage value between 0 and 100. An optional colour setting enables a string to be used to indicate the desired colour. Table 41 and Table 42 define the Resource Type and its properties, respectively.

**Table 41. Light control Resource Type definition**

Resource Type	Resource Type ID	Multiple Instances	Description
Light Control	"oic.light.control" or "urn:oma:lwm2m:ext:3311"	Yes	Light control object with on/off and optional dimming and energy monitor

**Table 42. Light control Resource Type definition**

Property title	Property name	Value type	Value rule	Unit	Access mode	Mandatory	Description
<b>On/Off</b>	"on-off"	boolean			R, W	yes	On/Of Control: 0 = Off 1 = On
<b>Dimmer</b>	"dim"	integer		%	R, W	no	Proportional Control, integer value between 0 and 100 as percentage
<b>Color</b>	"color"	string	0 – 100	Defined by "units" property	R, W	no	String representing some value in color space
<b>Units</b>	"units"	string			R	no	Measurement Units Definition e.g., "Cel" for Temperature in Celsius.
<b>On Time</b>	"ontime"	integer		s	R, W	no	The time in seconds that the light has been on.

							Writing a value of 0 resets the counter
<b>Cumulative active power</b>	"cumap"	float		Wh	R	no	The cumulative active power since the last cumulative energy reset or device start
<b>Power Factor</b>	"powfact"	float			R	no	The power factor of the load

3650

3651

**Annex D**  
(normative)

**Resource Type definitions**

**D.1 List of Resource Type definitions**

All the sections in Annex D and Annex E describe the Resource Types with a restful API definition language. The Resource Type definitions presented in Annex D and Annex E are formatted for readability, and so may appear to have extra line breaks. The contents of the Resource Types without the extra line breaks are available in OCF Resource Type Definitions.

Table 43 contains the list of defined core resources in this specification.

**Table 43. Alphabetized list of core resources**

<b>Friendly Name (informative)</b>	<b>Resource Type (rt)</b>	<b>Section</b>
<b>Collections</b>	"oic.wk.col"	D.2
<b>Device Configuration</b>	"oic.wk.con"	D.3
<b>Platform Configuration</b>	"oic.wk.con.p"	D.4
<b>Device</b>	"oic.wk.d"	D.5
<b>Discoverable Resources, baseline interface</b>	"oic.wk.res"	D.9
<b>Discoverable Resources, link list interface</b>	"oic.wk.res"	D.10
<b>Icon</b>	"oic.r.icon"	D.15
<b>Introspection</b>	"oic.wk.introspection"	D.16
<b>Maintenance</b>	"oic.wk.mnt"	D.6
<b>Platform</b>	"oic.wk.p"	D.7
<b>Ping</b>	"oic.wk.ping"	D.8
<b>Resource Directory</b>	"oic.wk.rd"	D.14
<b>Scenes (Top Level)</b>	"oic.wk.scenelist"	D.11

<b>Scenes Collections</b>	"oic.wk.scenecollection"	D.12
<b>Scenes Member</b>	"oic.wk.scenemember"	D.13

3663

## 3664 D.2 OCF Collection

### 3665 D.2.1 Introduction

3666 OCF Collection Resource Type contains properties and links. The oic.if.baseline interface exposes  
3667 a representation of the links and the properties of the collection resource itself

### 3668 D.2.2 Example URI

3669 /CollectionBaselineInterfaceURI

### 3670 D.2.3 Resource Type

3671 The resource type (rt) is defined as: oic.wk.col.

### 3672 D.2.4 RAML Definition

```

3673 #%RAML 0.8
3674 title: Collections
3675 version: 1.0
3676 traits:
3677   - interface-ll :
3678     queryParameters:
3679       if:
3680         enum: ["oic.if.ll"]
3681   - interface-b :
3682     queryParameters:
3683       if:
3684         enum: ["oic.if.b"]
3685   - interface-baseline :
3686     queryParameters:
3687       if:
3688         enum: ["oic.if.baseline"]
3689   - interface-all :
3690     queryParameters:
3691       if:
3692         enum: ["oic.if.ll", "oic.if.baseline", "oic.if.b"]
3693
3694 /CollectionBaselineInterfaceURI:
3695   description: |
3696     OCF Collection Resource Type contains properties and links.
3697     The oic.if.baseline interface exposes a representation of
3698     the links and the properties of the collection resource itself
3699
3700   is : ['interface-baseline']
3701   get:
3702     description: |
3703       Retrieve on Baseline Interface
3704
3705   responses :
3706     200:
3707     body:
```

```

3708     application/json:
3709         schema: /
3710             {
3711                 "$schema": "http://json-schema.org/draft-04/schema#",
3712                 "description" : "Copyright (c) 2016 Open Connectivity Foundation, Inc. All rights
3713 reserved.",
3714                 "id": "http://www.openconnectivity.org/ocf-apis/core/schemas/oic.collection-
3715 schema.json#",
3716                 "title": "Collection",
3717                 "definitions": {
3718                     "oic.collection.setoflinks": {
3719                         "description": "A set (array) of simple or individual OIC Links. In
3720 addition to properties required for an OIC Link, the identifier for that link in this set is also
3721 required",
3722                         "type": "array",
3723                         "items": {
3724                             "$ref": "oic.oic-link-schema.json#/definitions/oic.oic-link"
3725                         }
3726                     },
3727                     "oic.collection.alllinks": {
3728                         "description": "All forms of links in a collection",
3729                         "oneOf": [
3730                             {
3731                                 "$ref": "#/definitions/oic.collection.setoflinks"
3732                             }
3733                         ]
3734                     },
3735                     "oic.collection": {
3736                         "type": "object",
3737                         "description": "A collection is a set (array) of tagged-link or set
3738 (array) of simple links along with additional properties to describe the collection itself",
3739                         "properties": {
3740                             "id": {
3741                                 "anyOf": [
3742                                     {
3743                                         "type": "integer",
3744                                         "description": "A number that is unique to that
3745 collection; like an ordinal number that is not repeated"
3746                                     },
3747                                     {
3748                                         "type": "string",
3749                                         "description": "A unique string that could be a hash or
3750 similarly unique"
3751                                     },
3752                                     {
3753                                         "$ref": "oic.types-schema.json#/definitions/uuid",
3754                                         "description": "A unique string that could be a UUIDv4"
3755                                     }
3756                                 ],
3757                                 "description": "ID for the collection. Can be an value that is
3758 unique to the use context or a UUIDv4"
3759                             },
3760                             "di": {
3761                                 "$ref": "oic.types-schema.json#/definitions/uuid",
3762                                 "description": "The device ID which is an UUIDv4 string; used for
3763 backward compatibility with Spec A definition of /oic/res"
3764                             },
3765                             "rts": {
3766                                 "$ref": "oic.core-
3767 schema.json#/definitions/oic.core/properties/rt",
3768                                 "description": "Defines the list of allowable resource types (for
3769 Target and anchors) in links included in the collection; new links being created can only be from
3770 this list"
3771                             },
3772                             "drel": {
3773                                 "type": "string",
3774                                 "description": "When specified this is the default relationship
3775 to use when an OIC Link does not specify an explicit relationship with *rel* parameter"
3776                             },
3777                             "links": {
3778                                 "$ref": "#/definitions/oic.collection.alllinks"
3779                             }
3780                         }
3781                     }
3782             }

```

```

3778         }
3779     }
3780 }
3781 },
3782 "type": "object",
3783 "allOf": [
3784     {"$ref": "oic.core-schema.json#/definitions/oic.core"},
3785     {"$ref": "#/definitions/oic.collection"}
3786 ]
3787 }
3788
3789 example: /
3790 {
3791     "rt": ["oic.wk.col"],
3792     "id": "unique_example_id",
3793     "rts": [ "oic.r.switch.binary", "oic.r.airflow" ],
3794     "links": [
3795         {
3796             "href": "switch",
3797             "rt": [ "oic.r.switch.binary" ],
3798             "if": [ "oic.if.a", "oic.if.baseline" ],
3799             "eps": [
3800                 { "ep": "coap://[fe80::b1d6]:1111", "pri": 2 },
3801                 { "ep": "coaps://[fe80::b1d6]:1122" },
3802                 { "ep": "coap+tcp://[2001:db8:a::123]:2222", "pri": 3 }
3803             ]
3804         },
3805         {
3806             "href": "airFlow",
3807             "rt": [ "oic.r.airflow" ],
3808             "if": [ "oic.if.a", "oic.if.baseline" ],
3809             "eps": [
3810                 { "ep": "coap://[fe80::b1d6]:1111", "pri": 2 },
3811                 { "ep": "coaps://[fe80::b1d6]:1122" },
3812                 { "ep": "coap+tcp://[2001:db8:a::123]:2222", "pri": 3 }
3813             ]
3814         }
3815     ]
3816 }
3817
3818 post:
3819     description: |
3820         Update on Baseline Interface
3821
3822     body:
3823         application/json:
3824             schema: /
3825                 {
3826                     "$schema": "http://json-schema.org/draft-04/schema#",
3827                     "description": "Copyright (c) 2016 Open Connectivity Foundation, Inc. All rights
3828 reserved.",
3829                     "id": "http://www.openconnectivity.org/ocf-apis/core/schemas/oic.collection-
3830 schema.json#",
3831                     "title": "Collection",
3832                     "definitions": {
3833                         "oic.collection.setoflinks": {
3834                             "description": "A set (array) of simple or individual OIC Links. In addition
3835 to properties required for an OIC Link, the identifier for that link in this set is also required",
3836                             "type": "array",
3837                             "items": {
3838                                 "$ref": "oic.oic-link-schema.json#/definitions/oic.oic-link"
3839                             }
3840                         },
3841                         "oic.collection.alllinks": {
3842                             "description": "All forms of links in a collection",
3843                             "oneOf": [
3844                                 {

```



```

3845         "$ref": "#/definitions/oic.collection.setoflinks"
3846     }
3847 ]
3848 },
3849 "oic.collection": {
3850     "type": "object",
3851     "description": "A collection is a set (array) of tagged-link or set (array)
3852 of simple links along with additional properties to describe the collection itself",
3853     "properties": {
3854         "id": {
3855             "anyOf": [
3856                 {
3857                     "type": "integer",
3858                     "description": "A number that is unique to that collection;
3859 like an ordinal number that is not repeated"
3860                 },
3861                 {
3862                     "type": "string",
3863                     "description": "A unique string that could be a hash or
3864 similarly unique"
3865                 },
3866                 {
3867                     "$ref": "oic.types-schema.json#/definitions/uuid",
3868                     "description": "A unique string that could be a UUIDv4"
3869                 }
3870             ],
3871             "description": "ID for the collection. Can be an value that is unique
3872 to the use context or a UUIDv4"
3873         },
3874         "di": {
3875             "$ref": "oic.types-schema.json#/definitions/uuid",
3876             "description": "The device ID which is an UUIDv4 string; used for
3877 backward compatibility with Spec A definition of /oic/res"
3878         },
3879         "rts": {
3880             "$ref": "oic.core-schema.json#/definitions/oic.core.properties.rts",
3881             "description": "Defines the list of allowable resource types (for
3882 Target and anchors) in links included in the collection; new links being created can only be from
3883 this list"
3884         },
3885         "drel": {
3886             "type": "string",
3887             "description": "When specified this is the default relationship to
3888 use when an OIC Link does not specify an explicit relationship with *rel* parameter"
3889         },
3890         "links": {
3891             "$ref": "#/definitions/oic.collection.alllinks"
3892         }
3893     },
3894 },
3895 "type": "object",
3896 "allOf": [
3897     {"$ref": "oic.core-schema.json#/definitions/oic.core"},
3898     {"$ref": "#/definitions/oic.collection"}
3899 ]
3900 }
3901
3902 responses :
3903 200:
3904     body:
3905         application/json:
3906             schema: /
3907             {
3908                 "$schema": "http://json-schema.org/draft-04/schema#",
3909                 "description" : "Copyright (c) 2016 Open Connectivity Foundation, Inc. All rights
3910 reserved.",
3911                 "id": "http://www.openconnectivity.org/ocf-apis/core/schemas/oic.collection-
3912 schema.json#",

```

```

3913         "title": "Collection",
3914         "definitions": {
3915             "oic.collection.setoflinks": {
3916                 "description": "A set (array) of simple or individual OIC Links. In
3917 addition to properties required for an OIC Link, the identifier for that link in this set is also
3918 required",
3919                 "type": "array",
3920                 "items": {
3921                     "$ref": "oic.oic-link-schema.json#/definitions/oic.oic-link"
3922                 }
3923             },
3924             "oic.collection.alllinks": {
3925                 "description": "All forms of links in a collection",
3926                 "oneOf": [
3927                     {
3928                         "$ref": "#/definitions/oic.collection.setoflinks"
3929                     }
3930                 ]
3931             },
3932             "oic.collection": {
3933                 "type": "object",
3934                 "description": "A collection is a set (array) of tagged-link or set
3935 (array) of simple links along with additional properties to describe the collection itself",
3936                 "properties": {
3937                     "id": {
3938                         "anyOf": [
3939                             {
3940                                 "type": "integer",
3941                                 "description": "A number that is unique to that
3942 collection; like an ordinal number that is not repeated"
3943                             },
3944                             {
3945                                 "type": "string",
3946                                 "description": "A unique string that could be a hash or
3947 similarly unique"
3948                             },
3949                             {
3950                                 "$ref": "oic.types-schema.json#/definitions/uuid",
3951                                 "description": "A unique string that could be a UUIDv4"
3952                             }
3953                         ],
3954                         "description": "ID for the collection. Can be an value that is
3955 unique to the use context or a UUIDv4"
3956                     },
3957                     "di": {
3958                         "$ref": "oic.types-schema.json#/definitions/uuid",
3959                         "description": "The device ID which is an UUIDv4 string; used for
3960 backward compatibility with Spec A definition of /oic/res"
3961                     },
3962                     "rts": {
3963                         "$ref": "oic.core-
3964 schema.json#/definitions/oic.core/properties/rt",
3965                         "description": "Defines the list of allowable resource types (for
3966 Target and anchors) in links included in the collection; new links being created can only be from
3967 this list"
3968                     },
3969                     "drel": {
3970                         "type": "string",
3971                         "description": "When specified this is the default relationship
3972 to use when an OIC Link does not specify an explicit relationship with *rel* parameter"
3973                     },
3974                     "links": {
3975                         "$ref": "#/definitions/oic.collection.alllinks"
3976                     }
3977                 }
3978             },
3979             "type": "object",
3980             "allOf": [
3981                 {"$ref": "oic.core-schema.json#/definitions/oic.core"},
3982                 {"$ref": "#/definitions/oic.collection"}
3983             ]
3984         }

```

3984 }  
 3985  
 3986

## D.2.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
rt	array: see schema	yes		Resource Type
di	multiple types: see schema			Unique identifier for device (UUID)
title	string			A title for the link relation. Can be used by the UI to provide a context
eps	array: see schema			the Endpoint information of the target Resource
pri (eps)	integer			The priority among multiple Endpoints as specified in 10.2.3
ep (eps)	string			URI with Transport Protocol Suites + Endpoint Locator as specified in 10.2.1
ins	multiple types: see schema			The instance identifier for this web link in an array of web links - used in collections
p	object: see schema			Specifies the framework policies on the Resource referenced by the target URI
bm (p)	integer	yes		Specifies the framework policies on the Resource referenced by the target URI for e.g. observable and discoverable
href	string	yes		This is the target URI, it can be specified as a Relative Reference or fully-qualified URI. Relative Reference

				should be used along with the di parameter to make it unique.
rel	multiple types: see schema			The relation of the target URI referenced by the link to the context URI
type	array: see schema			A hint at the representation of the resource referenced by the target URI. This represents the media types that are used for both accepting and emitting
anchor	string			This is used to override the context URI e.g. override the URI of the containing collection
if	array: see schema	yes		The interface set supported by this resource

## D.2.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/CollectionBaselineInterfaceURI		get	post		

## D.2.7 Referenced JSON schemas

### D.2.8 oic.oic-link-schema.json

```

{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "description": "Copyright (c) 2016, 2017 Open Connectivity Foundation, Inc. All rights
reserved.",
  "id": "http://www.openconnectivity.org/ocf-apis/core/schemas/oic.oic-link-schema.json#",
  "definitions": {
    "oic.oic-link": {
      "type": "object",
      "properties": {
        "href": {
          "type": "string",
          "maxLength": 256,
          "description": "This is the target URI, it can be specified as a Relative Reference or
fully-qualified URI. Relative Reference should be used along with the di parameter to make it
unique.",
          "format": "uri"
        },
        "rel": {
          "oneOf": [
            {
              "type": "array",
              "items": {
                "type": "string",
                "maxLength": 64
              },
              "minItems": 1,

```

```

4016         "default": ["hosts"]
4017     },
4018     {
4019         "type": "string",
4020         "maxLength": 64,
4021         "default": "hosts"
4022     }
4023 ],
4024 "description": "The relation of the target URI referenced by the link to the context URI"
4025 },
4026 "rt": {
4027     "type": "array",
4028     "items": {
4029         "type": "string",
4030         "maxLength": 64
4031     },
4032     "minItems": 1,
4033     "description": "Resource Type"
4034 },
4035 "if": {
4036     "type": "array",
4037     "items": {
4038         "type": "string",
4039         "enum": ["oic.if.baseline", "oic.if.ll", "oic.if.b", "oic.if.rw", "oic.if.r",
4040 "oic.if.a", "oic.if.s" ]
4041     },
4042     "minItems": 1,
4043     "description": "The interface set supported by this resource"
4044 },
4045 "di": {
4046     "$ref": "oic.types-schema.json#/definitions/uuid",
4047     "description": "Unique identifier for device (UUID)"
4048 },
4049 "p": {
4050     "description": "Specifies the framework policies on the Resource referenced by the target
4051 URI",
4052     "type": "object",
4053     "properties": {
4054         "bm": {
4055             "description": "Specifies the framework policies on the Resource referenced by the
4056 target URI for e.g. observable and discoverable",
4057             "type": "integer"
4058         }
4059     },
4060     "required": ["bm"]
4061 },
4062 "title": {
4063     "type": "string",
4064     "maxLength": 64,
4065     "description": "A title for the link relation. Can be used by the UI to provide a
4066 context"
4067 },
4068 "anchor": {
4069     "type": "string",
4070     "maxLength": 256,
4071     "description": "This is used to override the context URI e.g. override the URI of the
4072 containing collection",
4073     "format": "uri"
4074 },
4075 "ins": {
4076     "oneOf": [
4077         {
4078             "type": "integer",
4079             "description": "An ordinal number that is not repeated - must be unique in the
4080 collection context"
4081         },
4082         {
4083             "type": "string",
4084             "maxLength": 256,
4085             "format": "uri",
4086             "description": "Any unique string including a URI"

```

```

4087         },
4088         {
4089             "$ref": "oic.types-schema.json#/definitions/uuid",
4090             "description": "Unique identifier (UUID)"
4091         }
4092     ],
4093     "description": "The instance identifier for this web link in an array of web links - used
4094 in collections"
4095 },
4096 "type": {
4097     "type": "array",
4098     "description": "A hint at the representation of the resource referenced by the target
4099 URI. This represents the media types that are used for both accepting and emitting",
4100     "items": {
4101         "type": "string",
4102         "maxLength": 64
4103     },
4104     "minItems": 1,
4105     "default": "application/cbor"
4106 },
4107 "eps": {
4108     "type": "array",
4109     "description": "the Endpoint information of the target Resource",
4110     "items": {
4111         "type": "object",
4112         "properties": {
4113             "ep": {
4114                 "type": "string",
4115                 "format": "uri",
4116                 "description": "URI with Transport Protocol Suites + Endpoint Locator as specified
4117 in 10.2.1"
4118             },
4119             "pri": {
4120                 "type": "integer",
4121                 "minimum": 1,
4122                 "description": "The priority among multiple Endpoints as specified in 10.2.3"
4123             }
4124         }
4125     }
4126 },
4127 },
4128 "required": [ "href", "rt", "if" ]
4129 },
4130 },
4131 "type": "object",
4132 "allOf": [
4133     { "$ref": "#/definitions/oic.oic-link" }
4134 ]
4135 }
4136

```

## 4137 D.3 Device Configuration

### 4138 D.3.1 Introduction

4139 Resource that allows for Device specific information to be configured.

### 4140 D.3.2 Example URI

4141 /example/DeviceConfigurationResURI

### 4142 D.3.3 Resource Type

4143 The resource type (rt) is defined as: oic.wk.con.

### 4144 D.3.4 RAML Definition

```

4145 #%RAML 0.8
4146 title: OCF Configuration
4147 version: v1-20160622
4148 traits:

```

```

4149 - interface-rw :
4150     queryParameters:
4151         if:
4152             enum: ["oic.if.rw"]
4153 - interface-all :
4154     queryParameters:
4155         if:
4156             enum: ["oic.if.rw", "oic.if.baseline"]
4157
4158 /example/DeviceConfigurationResURI:
4159     description: |
4160         Resource that allows for Device specific information to be configured.
4161
4162     get:
4163         description: |
4164             Retrieves the current Device configuration settings
4165
4166         is : ['interface-all']
4167         responses :
4168             200:
4169                 body:
4170                     application/json:
4171                         schema: /
4172                             {
4173                                 "id": "http://www.openconnectivity.org/ocf-apis/core/schemas/oic.wk.con-
4174 schema.json#",
4175                                 "$schema": "http://json-schema.org/draft-04/schema#",
4176                                 "description": "Copyright (c) 2016, 2017 Open Connectivity Foundation, Inc. All
4177 rights reserved.",
4178                                 "definitions": {
4179                                     "oic.wk.con": {
4180                                         "type": "object",
4181                                         "properties": {
4182                                             "loc": {
4183                                                 "type": "array",
4184                                                 "description": "Location information",
4185                                                 "items": {
4186                                                     "type": "number"
4187                                                 },
4188                                                 "minItems": 2,
4189                                                 "maxItems": 2
4190                                             },
4191                                             "locn": {
4192                                                 "type": "string",
4193                                                 "maxLength": 64,
4194                                                 "description": "Human Friendly Name for location"
4195                                             },
4196                                             "c": {
4197                                                 "type": "string",
4198                                                 "maxLength": 64,
4199                                                 "description": "Currency"
4200                                             },
4201                                             "r": {
4202                                                 "type": "string",
4203                                                 "maxLength": 64,
4204                                                 "description": "Region"
4205                                             },
4206                                             "ln": {
4207                                                 "type": "array",
4208                                                 "items" :
4209                                                     {
4210                                                         "type": "object",
4211                                                         "properties": {

```

```

4212         "language": {
4213             "$ref": "oic.types-schema.json#/definitions/language-tag",
4214             "description": "An RFC 5646 language tag."
4215         },
4216         "value": {
4217             "type": "string",
4218             "maxLength": 64,
4219             "description": "Device description in the indicated language."
4220         }
4221     },
4222     },
4223     "minItems": 1,
4224     "description": "Localized names"
4225 },
4226 "dl": {
4227     "$ref": "oic.types-schema.json#/definitions/language-tag",
4228     "description": "Default Language"
4229 }
4230 }
4231 }
4232 },
4233 "type": "object",
4234 "allOf": [
4235     { "$ref": "oic.core-schema.json#/definitions/oic.core" },
4236     { "$ref": "#/definitions/oic.wk.con" }
4237 ],
4238 "required": ["n"]
4239 }
4240
4241 example: /
4242 {
4243     "n": "My Friendly Device Name",
4244     "rt": ["oic.wk.con"],
4245     "loc": [32.777,-96.797],
4246     "locn": "My Location Name",
4247     "c": "USD",
4248     "r": "MyRegion",
4249     "dl": "en"
4250 }
4251
4252 post:
4253     description: |
4254         Update the information about the Device
4255
4256     is : ['interface-rw']
4257     body:
4258         application/json:
4259             schema: /
4260                 {
4261                     "id": "http://www.openconnectivity.org/ocf-apis/core/schemas/oic.wk.con-Update-
4262 schema.json#",
4263                     "$schema": "http://json-schema.org/draft-04/schema#",
4264                     "description": "Copyright (c) 2016 Open Connectivity Foundation, Inc. All rights
4265 reserved.",
4266                     "definitions": {
4267                         "oic.wk.con": {
4268                             "type": "object",
4269                             "properties": {
4270                                 "loc": {
4271                                     "type": "array",
4272                                     "description": "Location information",
4273                                     "items": {
4274                                         "type": "number"
4275                                     },
4276                                     "minItems": 2,
4277                                     "maxItems": 2
4278                                 },

```



```

4279         "locn": {
4280             "type": "string",
4281             "maxLength": 64,
4282             "description": "Human Friendly Name for location"
4283         },
4284         "c": {
4285             "type": "string",
4286             "maxLength": 64,
4287             "description": "Currency"
4288         },
4289         "r": {
4290             "type": "string",
4291             "maxLength": 64,
4292             "description": "Region"
4293         },
4294         "ln": {
4295             "type": "array",
4296             "items":
4297             {
4298                 "type": "object",
4299                 "properties": {
4300                     "language": {
4301                         "$ref": "oic.types-schema.json#/definitions/language-tag",
4302                         "description": "An RFC 5646 language tag."
4303                     },
4304                     "value": {
4305                         "type": "string",
4306                         "maxLength": 64,
4307                         "description": "Device description in the indicated language."
4308                     }
4309                 }
4310             },
4311             "minItems": 1,
4312             "description": "Localized names"
4313         },
4314         "dl": {
4315             "$ref": "oic.types-schema.json#/definitions/language-tag",
4316             "description": "Default Language"
4317         }
4318     }
4319 },
4320 {
4321     "type": "object",
4322     "allOf": [
4323         { "$ref": "oic.core-schema.rw.json#/definitions/oic.core" },
4324         { "$ref": "#/definitions/oic.wk.con" }
4325     ],
4326     "required": ["n"]
4327 }
4328
4329 example: /
4330 {
4331     "n": "Nuevo Nombre Amistoso",
4332     "r": "MyNewRegion",
4333     "ln": [ { "language": "es", "value": "Nuevo Nombre Amistoso" } ],
4334     "dl": "es"
4335 }
4336
4337 responses :
4338 200:
4339     body:
4340         application/json:
4341             schema: /
4342             {
4343                 "id": "http://www.openconnectivity.org/ocf-apis/core/schemas/oic.wk.con-Update-
4344 schema.json#",
4345                 "$schema": "http://json-schema.org/draft-04/schema#",

```

```

4346         "description" : "Copyright (c) 2016 Open Connectivity Foundation, Inc. All rights
4347 reserved.",
4348         "definitions": {
4349             "oic.wk.con": {
4350                 "type": "object",
4351                 "properties": {
4352                     "loc": {
4353                         "type": "array",
4354                         "description": "Location information",
4355                         "items": {
4356                             "type": "number"
4357                         },
4358                         "minItems": 2,
4359                         "maxItems": 2
4360                     },
4361                     "locn": {
4362                         "type": "string",
4363                         "maxLength": 64,
4364                         "description": "Human Friendly Name for location"
4365                     },
4366                     "c": {
4367                         "type": "string",
4368                         "maxLength": 64,
4369                         "description": "Currency"
4370                     },
4371                     "r": {
4372                         "type": "string",
4373                         "maxLength": 64,
4374                         "description": "Region"
4375                     },
4376                     "ln": {
4377                         "type": "array",
4378                         "items" :
4379                         {
4380                             "type": "object",
4381                             "properties": {
4382                                 "language": {
4383                                     "$ref": "oic.types-schema.json#/definitions/language-tag",
4384                                     "description": "An RFC 5646 language tag."
4385                                 },
4386                                 "value": {
4387                                     "type": "string",
4388                                     "maxLength": 64,
4389                                     "description": "Device description in the indicated language."
4390                                 }
4391                             }
4392                         },
4393                         "minItems" : 1,
4394                         "description": "Localized names"
4395                     },
4396                     "dl": {
4397                         "$ref": "oic.types-schema.json#/definitions/language-tag",
4398                         "description": "Default Language"
4399                     }
4400                 }
4401             }
4402         },
4403         "type": "object",
4404         "allOf": [
4405             { "$ref": "oic.core-schema.rw.json#/definitions/oic.core"},
4406             { "$ref": "#/definitions/oic.wk.con" }
4407         ],
4408         "required": ["n"]
4409     }
4410
4411     example: /
4412     {
4413         "n": "Nuevo Nombre Amistoso",
4414         "r": "MyNewRegion",
4415         "ln": [ { "language": "es", "value": "Nuevo Nombre Amistoso" } ],

```

4416 "dl" : "es"  
4417 }  
4418

4419 **D.3.5 Property Definition**

Property name	Value type	Mandatory	Access mode	Description
loc	array: see schema			Location information
c	string			Currency
ln	array: see schema			Localized names
value (ln)	string			Device description in the indicated language.
language (ln)	multiple types: see schema			An RFC 5646 language tag.
locn	string			Human Friendly Name for location
dl	multiple types: see schema			Default Language
r	string			Region

4420 **D.3.6 CRUDN behavior**

Resource	Create	Read	Update	Delete	Notify
/example/DeviceConfigurationResURI		get	post		

4421 **D.4 Platform Configuration**

4422 **D.4.1 Introduction**

4423 Resource that allows for platform specific information to be configured.

4424 **D.4.2 Example URI**

4425 /example/PlatformConfigurationResURI

4426 **D.4.3 Resource Type**

4427 The resource type (rt) is defined as: oic.wk.con.p.

4428 **D.4.4 RAML Definition**

4429 `##RAML 0.8`  
4430 `title: OCF Platform Configuration`  
4431 `version: v1-20160622`  
4432 `traits:`  
4433 `- interface-rw :`  
4434  `queryParameters:`  
4435  `if:`  
4436  `enum: ["oic.if.rw"]`  
4437 `- interface-all :`  
4438  `queryParameters:`  
4439  `if:`  
4440  `enum: ["oic.if.rw", "oic.if.baseline"]`  
4441  
4442 `/example/PlatformConfigurationResURI:`  
4443  `description: |`

```

4444     Resource that allows for platform specific information to be configured.
4445
4446 get:
4447     description: |
4448         Retrieves the current platform configuration settings
4449
4450     is : ['interface-all']
4451
4452     responses :
4453         200:
4454             body:
4455                 application/json:
4456                     schema: /
4457                         {
4458 schema.json#",
4459                 "$schema": "http://json-schema.org/draft-04/schema#",
4460                 "description" : "Copyright (c) 2017 Open Connectivity Foundation, Inc. All rights
4461 reserved.",
4462                 "definitions": {
4463                     "oic.wk.con.p": {
4464                         "type": "object",
4465                         "properties": {
4466                             "mnpn": {
4467                                 "type": "array",
4468                                 "items" :
4469                                     {
4470                                         "type": "object",
4471                                         "properties": {
4472                                             "language": {
4473                                                 "$ref": "oic.types-schema.json#/definitions/language-tag",
4474                                                 "description": "An RFC 5646 language tag."
4475                                             },
4476                                             "value": {
4477                                                 "type": "string",
4478                                                 "maxLength": 64,
4479                                                 "description": "Platform description in the indicated language."
4480                                             }
4481                                         }
4482                                     },
4483                                     "minItems" : 1,
4484                                     "description": "Platform names"
4485                                 }
4486                             }
4487                         },
4488                         "type": "object",
4489                         "allOf": [
4490                             { "$ref": "oic.core-schema.json#/definitions/oic.core" },
4491                             { "$ref": "#/definitions/oic.wk.con.p" }
4492                         ]
4493                     }
4494                 }
4495
4496             example: /
4497                 {
4498                     "rt":    ["oic.wk.con.p"],
4499                     "mnpn": [ { "language": "en", "value": "My Friendly Device Name" } ]
4500                 }
4501
4502 post:
4503     description: |
4504         Update the information about the platform
4505
4506     is : ['interface-rw']

```

```

4507     body:
4508         application/json:
4509             schema: /
4510                 {
4511                     "id": "http://www.openconnectivity.org/ocf-apis/core/schemas/oic.wk.con.p-Update-
4512 schema.json#",
4513                     "$schema": "http://json-schema.org/draft-04/schema#",
4514                     "description" : "Copyright (c) 2017 Open Connectivity Foundation, Inc. All rights
4515 reserved.",
4516                     "definitions": {
4517                         "oic.wk.con.p": {
4518                             "type": "object",
4519                             "properties": {
4520                                 "mnpn": {
4521                                     "type": "array",
4522                                     "items" :
4523                                         {
4524                                             "type": "object",
4525                                             "properties": {
4526                                                 "language": {
4527                                                     "$ref": "oic.types-schema.json#/definitions/language-tag",
4528                                                     "description": "An RFC 5646 language tag."
4529                                                 },
4530                                                 "value": {
4531                                                     "type": "string",
4532                                                     "maxLength": 64,
4533                                                     "description": "Platform description in the indicated language."
4534                                                 }
4535                                             }
4536                                         },
4537                                     "minItems" : 1,
4538                                     "description": "Platform names"
4539                                 }
4540                             }
4541                         },
4542                     },
4543                     "type": "object",
4544                     "allOf": [
4545                         { "$ref": "oic.core-schema.rw.json#/definitions/oic.core" },
4546                         { "$ref": "#/definitions/oic.wk.con.p" }
4547                     ],
4548                     "required": [ "mnpn" ]
4549                 }
4550
4551     example: /
4552         {
4553             "n": "Nuevo nombre",
4554             "mnpn": [ { "language": "es", "value": "Nuevo nombre de Plataforma Amigable" } ]
4555         }
4556
4557     responses :
4558         200:
4559             body:
4560                 application/json:
4561                     schema: /
4562                         {
4563                             "id": "http://www.openconnectivity.org/ocf-apis/core/schemas/oic.wk.con.p-Update-
4564 schema.json#",
4565                             "$schema": "http://json-schema.org/draft-04/schema#",
4566                             "description" : "Copyright (c) 2017 Open Connectivity Foundation, Inc. All rights
4567 reserved.",
4568                             "definitions": {
4569                                 "oic.wk.con.p": {
4570                                     "type": "object",
4571                                     "properties": {
4572                                         "mnpn": {

```

```

4573         "type": "array",
4574         "items" :
4575         {
4576             "type": "object",
4577             "properties": {
4578                 "language": {
4579                     "$ref": "oic.types-schema.json#/definitions/language-tag",
4580                     "description": "An RFC 5646 language tag."
4581                 },
4582                 "value": {
4583                     "type": "string",
4584                     "maxLength": 64,
4585                     "description": "Platform description in the indicated language."
4586                 }
4587             }
4588         },
4589         "minItems" : 1,
4590         "description": "Platform names"
4591     }
4592 }
4593 }
4594 },
4595 "type": "object",
4596 "allof": [
4597     { "$ref": "oic.core-schema.rw.json#/definitions/oic.core"},
4598     { "$ref": "#/definitions/oic.wk.con.p" }
4599 ],
4600 "required": ["mnpn"]
4601 }
4602
4603 example: /
4604 {
4605     "n": "Nuevo nombre",
4606     "mnpn": [ { "language": "es", "value": "Nuevo nombre de Plataforma Amigable" } ]
4607 }
4608

```

#### D.4.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
mnpn	array: see schema			Platform names
value (mnpn)	string			Platform description in the indicated language.
language (mnpn)	multiple types: see schema			An RFC 5646 language tag.

#### D.4.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/example/PlatformConfigurationResURI		get	post		

### D.5 Device

#### D.5.1 Introduction

Known resource that is hosted by every Server. Allows for logical device specific information to be discovered.

#### D.5.2 Wellknown URI

/oic/d

#### D.5.3 Resource Type

The resource type (rt) is defined as: oic.wk.d.

## D.5.4 RAML Definition

```
4619  #%RAML 0.8
4620  title: OIC Root Device
4621  version: v1-20160622
4622
4623  traits:
4624    - interface :
4625      queryParameters:
4626        if:
4627          enum: ["oic.if.r", "oic.if.baseline"]
4628
4629  /oic/d:
4630    description: |
4631      Known resource that is hosted by every Server.
4632      Allows for logical device specific information to be discovered.
4633
4634    is : ['interface']
4635
4636    get:
4637      description: |
4638        Retrieve the information about the Device
4639
4640      responses :
4641        200:
4642          body:
4643            application/json:
4644              schema: /
4645                {
4646                  "$schema": "http://json-schemas.org/draft-04/schema#",
4647                  "description" : "Copyright (c) 2016, 2017 Open Connectivity Foundation, Inc. All
4648  rights reserved.",
4649                  "id": "http://www.openconnectivity.org/ocf-apis/core/schemas/oic.wk.d-
4650  schema.json#",
4651                  "definitions": {
4652                    "oic.wk.d": {
4653                      "type": "object",
4654                      "properties": {
4655                        "di": {
4656                          "$ref": "oic.types-schema.json#/definitions/uuid",
4657                          "readOnly": true,
4658                          "description": "Unique identifier for device (UUID)"
4659                        },
4660                        "icv": {
4661                          "type": "string",
4662                          "maxLength": 64,
4663                          "readOnly": true,
4664                          "description": "The version of the OIC Server"
4665                        },
4666                        "dmv": {
4667                          "type": "string",
4668                          "maxLength": 256,
4669                          "readOnly": true,
4670                          "description": "Spec versions of the Resource and Device Specifications to
4671  which this device data model is implemented"
4672                        },
4673                        "ld": {
4674                          "type": "array",
4675                          "items" :
4676                          {
4677                            "type": "object",
4678                            "properties": {
4679                              "language": {
4680                                "$ref": "oic.types-schema.json#/definitions/language-tag",
4681                                "readOnly": true,
```

```

4681         "description": "An RFC 5646 language tag."
4682     },
4683     "value": {
4684         "type": "string",
4685         "maxLength": 64,
4686         "readOnly": true,
4687         "description": "Device description in the indicated language."
4688     }
4689 },
4690 },
4691 "minItems" : 1,
4692 "readOnly": true,
4693 "description": "Localized Description."
4694 },
4695 "sv": {
4696     "type": "string",
4697     "maxLength": 64,
4698     "readOnly": true,
4699     "description": "Software version."
4700 },
4701 "dmn": {
4702     "type": "array",
4703     "items" :
4704     {
4705         "type": "object",
4706         "properties": {
4707             "language": {
4708                 "$ref": "oic.types-schema.json#/definitions/language-tag",
4709                 "readOnly": true,
4710                 "description": "An RFC 5646 language tag."
4711             },
4712             "value": {
4713                 "type": "string",
4714                 "maxLength": 64,
4715                 "readOnly": true,
4716                 "description": "Manufacturer name in the indicated language."
4717             }
4718         }
4719     },
4720     "minItems" : 1,
4721     "readOnly": true,
4722     "description": "Manufacturer Name."
4723 },
4724 "dmno": {
4725     "type": "string",
4726     "maxLength": 64,
4727     "readOnly": true,
4728     "description": "Model number as designated by manufacturer."
4729 },
4730 "piid": {
4731     "$ref": "oic.types-schema.json#/definitions/uuid",
4732     "readOnly": true,
4733     "description": "Protocol independent unique identifier for device (UUID)
4734 that is immutable."
4735 },
4736 }
4737 }
4738 },
4739 "type": "object",
4740 "allOf": [
4741     { "$ref": "oic.core-schema.json#/definitions/oic.core" },
4742     { "$ref": "#/definitions/oic.wk.d" }
4743 ],
4744 "required": [ "n", "di", "icv", "dmv", "piid" ]
4745 }
4746
4747 example: /
4748 {
4749     "n":      "Device 1",
4750     "rt":     ["oic.wk.d"],

```



```

4751         "di": "54919CA5-4101-4AE4-595B-353C51AA983C",
4752         "icv": "ocf.1.0.0",
4753         "dmv": "ocf.res.1.0.0, ocf.sh.1.0.0",
4754         "piid": "6F0AAC04-2BB0-468D-B57C-16570A26AE48"
4755     }
4756
4757

```

### D.5.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
Id	array: see schema		Read Only	Localized Description.
value (Id)	string		Read Only	Device description in the indicated language.
language (Id)	multiple types: see schema		Read Only	An RFC 5646 language tag.
piid	multiple types: see schema	yes	Read Only	Protocol independent unique identifier for device (UUID) that is immutable.
di	multiple types: see schema	yes	Read Only	Unique identifier for device (UUID)
dmno	string		Read Only	Model number as designated by manufacturer.
sv	string		Read Only	Software version.
dmn	array: see schema		Read Only	Manufacturer Name.
value (dmn)	string		Read Only	Manufacturer name in the indicated language.
language (dmn)	multiple types: see schema		Read Only	An RFC 5646 language tag.
dmv	string	yes	Read Only	Spec versions of the Resource and Device Specifications to which this device data model is implemented
icv	string	yes	Read Only	The version of the OIC Server

### D.5.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/oic/d		get			

## D.6 Maintenance

### D.6.1 Introduction

The resource through which a Device is maintained and can be used for diagnostic purposes. fr (Factory Reset) is a boolean. The value 0 means No action (Default), the value 1 means Start Factory Reset After factory reset, this value shall be changed back to the default value rb (Reboot) is a boolean. The value 0 means No action (Default), the value 1 means Start Reboot After Reboot, this value shall be changed back to the default value

### D.6.2 Wellknown URI

/oic/mnt

### D.6.3 Resource Type

The resource type (rt) is defined as: oic.wk.mnt.

### D.6.4 RAML Definition

```
##RAML 0.8
title: Maintenance
version: v1-20160622

traits:
- interface-rw :
  queryParameters:
    if:
      enum: ["oic.if.rw", "oic.if.baseline"]
- interface-all :
  queryParameters:
    if:
      enum: ["oic.if.rw", "oic.if.r", "oic.if.baseline"]

/oic/mnt:
  description: |
    The resource through which a Device is maintained and can be used for diagnostic purposes.
    fr (Factory Reset) is a boolean.
    The value 0 means No action (Default), the value 1 means Start Factory Reset
    After factory reset, this value shall be changed back to the default value
    rb (Reboot) is a boolean.
    The value 0 means No action (Default), the value 1 means Start Reboot
    After Reboot, this value shall be changed back to the default value

  get:
    is : ['interface-all']
    description: |
      Retrieve the maintenance action status

  responses :
    200:
      body:
        application/json:
          schema: /
            {
              "$schema": "http://json-schemas.org/draft-04/schema#",
              "description" : "Copyright (c) 2016, 2017 Open Connectivity Foundation, Inc. All
rights reserved.",
              "id": "http://www.openconnectivity.org/ocf-apis/core/schemas/oic.wk.mnt-
schema.json#",
              "definitions": {
                "oic.wk.mnt": {
                  "type": "object",
```

```

4813         "anyOf": [
4814             {"required": ["fr"]},
4815             {"required": ["rb"]}
4816         ],
4817         "properties": {
4818             "fr": {
4819                 "type": "boolean",
4820                 "description": "Factory Reset"
4821             },
4822             "rb": {
4823                 "type": "boolean",
4824                 "description": "Reboot Action"
4825             }
4826         }
4827     },
4828     "type": "object",
4829     "allOf": [
4830         {"$ref": "oic.core-schema.json#/definitions/oic.core"},
4831         {"$ref": "#/definitions/oic.wk.mnt" }
4832     ]
4833 }
4834
4835
4836 example: /
4837 {
4838     "rt":    ["oic.wk.mnt"],
4839     "fr":    false,
4840     "rb":    false
4841 }
4842
4843 post:
4844     is : ['interface-rw']
4845     description: |
4846         Set the maintenance action(s)
4847
4848     body:
4849         application/json:
4850             schema: /
4851                 {
4852                     "$schema": "http://json-schemas.org/draft-04/schema#",
4853                     "description" : "Copyright (c) 2016, 2017 Open Connectivity Foundation, Inc. All rights
4854 reserved.",
4855                     "id": "http://www.openconnectivity.org/ocf-apis/core/schemas/oic.wk.mnt-schema.json#",
4856                     "definitions": {
4857                         "oic.wk.mnt": {
4858                             "type": "object",
4859                             "anyOf": [
4860                                 {"required": ["fr"]},
4861                                 {"required": ["rb"]}
4862                             ],
4863                             "properties": {
4864                                 "fr": {
4865                                     "type": "boolean",
4866                                     "description": "Factory Reset"
4867                                 },
4868                                 "rb": {
4869                                     "type": "boolean",
4870                                     "description": "Reboot Action"
4871                                 }
4872                             }
4873                         }
4874                     },
4875                     "type": "object",
4876                     "allOf": [
4877                         {"$ref": "oic.core-schema.json#/definitions/oic.core"},
4878                         {"$ref": "#/definitions/oic.wk.mnt" }
4879                     ]

```

```

4880     }
4881
4882     example: /
4883     {
4884         "fr":    false,
4885         "rb":    false
4886     }
4887
4888     responses :
4889         200:
4890             body:
4891                 application/json:
4892                     schema: /
4893                     {
4894                         "$schema": "http://json-schemas.org/draft-04/schema#",
4895                         "description" : "Copyright (c) 2016, 2017 Open Connectivity Foundation, Inc. All
4896 rights reserved.",
4897                         "id": "http://www.openconnectivity.org/ocf-apis/core/schemas/oic.wk.mnt-
4898 schema.json#",
4899                         "definitions": {
4900                             "oic.wk.mnt": {
4901                                 "type": "object",
4902                                 "anyOf": [
4903                                     {"required": ["fr"]},
4904                                     {"required": ["rb"]}
4905                                 ],
4906                                 "properties": {
4907                                     "fr": {
4908                                         "type": "boolean",
4909                                         "description": "Factory Reset"
4910                                     },
4911                                     "rb": {
4912                                         "type": "boolean",
4913                                         "description": "Reboot Action"
4914                                     }
4915                                 }
4916                             }
4917                         },
4918                         "type": "object",
4919                         "allOf": [
4920                             {"$ref": "oic.core-schema.json#/definitions/oic.core"},
4921                             {"$ref": "#/definitions/oic.wk.mnt" }
4922                         ]
4923                     }
4924
4925     example: /
4926     {
4927         "fr":    false,
4928         "rb":    false
4929     }
4930

```

## D.6.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
fr	boolean	yes		Factory Reset
rb	boolean	yes		Reboot Action

## D.6.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/oic/mnt		get	post		

## 4933 D.7 Platform

### 4934 D.7.1 Introduction

4935 Known resource that is defines the platform on which an Server is hosted. Allows for platform  
4936 specific information to be discovered.

### 4937 D.7.2 Wellknown URI

4938 /oic/p

### 4939 D.7.3 Resource Type

4940 The resource type (rt) is defined as: oic.wk.p.

### 4941 D.7.4 RAML Definition

4942 `##RAML 0.8`

4943 `title: Platform`

4944 `version: v1-20160622`

4945 `traits:`

4946 `- interface :`

4947  `queryParameters:`

4948  `if:`

4949  `enum: ["oic.if.r", "oic.if.baseline"]`

4950

4951 `/oic/p:`

4952  `description: |`

4953  `Known resource that is defines the platform on which an Server is hosted.`

4954  `Allows for platform specific information to be discovered.`

4955

4956  `is : ['interface']`

4957  `get:`

4958  `description: |`

4959  `Retrieve the information about the Platform`

4960

4961  `responses :`

4962  `200:`

4963  `body:`

4964  `application/json:`

4965  `schema: /`

4966  `{`  
4967  `"$schema": "http://json-schemas.org/draft-04/schema#",`  
4968  `"description" : "Copyright (c) 2016, 2017 Open Connectivity Foundation, Inc. All`  
4969 `rights reserved.",`

4970  `"id": "http://www.openconnectivity.org/ocf-apis/core/schemas/oic.wk.p-`  
4971 `schema.json#",`

4972  `"definitions": {`

4973  `"oic.wk.p": {`

4974  `"type": "object",`

4975  `"properties": {`

4976  `"pi": {`

4977  `"$ref": "oic.types-schema.json#/definitions/uuid",`

4978  `"readOnly": true,`

4979  `"description": "Platform Identifier as a UUID"`

4980  `},`

4981  `"mnmn": {`

4982  `"type": "string",`

4983  `"readOnly": true,`

4984  `"description": "Manufacturer Name",`

4985  `"maxLength": 64`

4986  `},`

4987  `"mnm1": {`

```

4988         "type": "string",
4989         "readOnly": true,
4990         "description": "Manufacturer's URL",
4991         "maxLength": 256,
4992         "format": "uri"
4993     },
4994     "mnmo": {
4995         "type": "string",
4996         "maxLength": 64,
4997         "readOnly": true,
4998         "description": "Model number as designated by manufacturer"
4999     },
5000     "mndt": {
5001         "$ref": "oic.types-schema.json#/definitions/date",
5002         "readOnly": true,
5003         "description": "Manufacturing Date."
5004     },
5005     "mnpv": {
5006         "type": "string",
5007         "maxLength": 64,
5008         "readOnly": true,
5009         "description": "Platform Version"
5010     },
5011     "mnos": {
5012         "type": "string",
5013         "maxLength": 64,
5014         "readOnly": true,
5015         "description": "Platform Resident OS Version"
5016     },
5017     "mnhw": {
5018         "type": "string",
5019         "maxLength": 64,
5020         "readOnly": true,
5021         "description": "Platform Hardware Version"
5022     },
5023     "mnfv": {
5024         "type": "string",
5025         "maxLength": 64,
5026         "readOnly": true,
5027         "description": "Manufacturer's firmware version"
5028     },
5029     "mnsi": {
5030         "type": "string",
5031         "readOnly": true,
5032         "description": "Manufacturer's Support Information URL",
5033         "maxLength": 256,
5034         "format": "uri"
5035     },
5036     "st": {
5037         "type": "string",
5038         "readOnly": true,
5039         "description": "Reference time for the device as defined in ISO 8601, where
5040 concatenation of 'date' and 'time' with the 'T' as a delimiter between 'date' and 'time'.",
5041         "format": "date-time"
5042     },
5043     "vid": {
5044         "type": "string",
5045         "maxLength": 64,
5046         "readOnly": true,
5047         "description": "Manufacturer's defined string for the platform. The string
5048 is freeform and up to the manufacturer on what text to populate it"
5049     }
5050 }
5051 }
5052 },
5053 "type": "object",
5054 "allOf": [
5055     { "$ref": "oic.core-schema.json#/definitions/oic.core" },
5056     { "$ref": "oic.wk.p" }
5057 ],
5058 "required": [ "pi", "mnmn" ]

```

```

5059     }
5060
5061     example: /
5062     {
5063         "pi": "54919CA5-4101-4AE4-595B-353C51AA983C",
5064         "rt": ["oic.wk.p"],
5065         "mnmn": "Acme, Inc"
5066     }
5067

```

## D.7.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
mnfv	string		Read Only	Manufacturer's firmware version
vid	string		Read Only	Manufacturer's defined string for the platform. The string is freeform and up to the manufacturer on what text to populate it
mnmn	string	yes	Read Only	Manufacturer Name
mnmo	string		Read Only	Model number as designated by manufacturer
mnml	string		Read Only	Manufacturer's URL
mnos	string		Read Only	Platform Resident OS Version
mndt	multiple types: see schema		Read Only	Manufacturing Date.
st	string		Read Only	Reference time for the device as defined in ISO 8601, where concatenation of 'date' and 'time' with the 'T' as a delimiter between 'date' and 'time'.
mnsi	string		Read Only	Manufacturer's Support Information URL
mpv	string		Read Only	Platform Version
pi	multiple types: see schema	yes	Read Only	Platform Identifier as a UUID
mnhw	string		Read Only	Platform Hardware Version

## D.7.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/oic/p		get			

## D.8 Ping

### D.8.1 Introduction

The resource using which an Client keeps its Connection with an Server active.

### D.8.2 Wellknown URI

/oic/ping

### D.8.3 Resource Type

The resource type (rt) is defined as: oic.wk.ping.

### D.8.4 RAML Definition

```
##RAML 0.8
title: Ping
version: v1-20160622

traits:
  - interface :
      queryParameters:
        if:
          enum: ["oic.if.rw", "oic.if.baseline"]

/oic/ping:
  description: |
    The resource using which an Client keeps its Connection with an Server active.

  is : ['interface']

  get:
    description: |
      Retrieve the ping information

    responses :
      200:
        body:
          application/json:
            schema: /
              {
                "$schema": "http://json-schemas.org/draft-04/schema#",
                "description" : "Copyright (c) 2016, 2017 Open Connectivity Foundation, Inc. All
rights reserved.",
                "id": "http://www.openconnectivity.org/ocf-apis/core/schemas/oic.wk.ping-
schema.json#",
                "definitions": {
                  "oic.wk.ping": {
                    "type": "object",
                    "properties": {
                      "in": {
                        "type": "integer",
                        "readOnly": false,
                        "description": "Indicates the interval for which connection shall be kept
alive"
                      }
                    }
                  }
                }
              },
```



```

5120         "type": "object",
5121         "allof": [
5122             { "$ref": "oic.core-schema.json#/definitions/oic.core"},
5123             { "$ref": "#/definitions/oic.wk.ping"}
5124         ],
5125         "required": [
5126             "in"
5127         ]
5128     }
5129
5130     example: /
5131     {
5132         "rt": ["oic.wk.ping"],
5133         "n": "Ping Information",
5134         "in": 16
5135     }
5136
5137     post:
5138         description: |
5139             Update or reset the alive interval
5140
5141         body:
5142             application/json:
5143                 schema: /
5144                 {
5145                     "$schema": "http://json-schemas.org/draft-04/schema#",
5146                     "description": "Copyright (c) 2016, 2017 Open Connectivity Foundation, Inc. All rights
5147 reserved.",
5148                     "id": "http://www.openconnectivity.org/ocf-apis/core/schemas/oic.wk.ping-schema.json#",
5149                     "definitions": {
5150                         "oic.wk.ping": {
5151                             "type": "object",
5152                             "properties": {
5153                                 "in": {
5154                                     "type": "integer",
5155                                     "readOnly": false,
5156                                     "description": "Indicates the interval for which connection shall be kept
5157 alive"
5158                                 }
5159                             }
5160                         },
5161                     },
5162                     "type": "object",
5163                     "allof": [
5164                         { "$ref": "oic.core-schema.json#/definitions/oic.core"},
5165                         { "$ref": "#/definitions/oic.wk.ping"}
5166                     ],
5167                     "required": [
5168                         "in"
5169                     ]
5170                 }
5171
5172     example: /
5173     {
5174         "in": 16
5175     }
5176
5177     responses :
5178         203:
5179             description: |
5180                 Successfully updated & restarted alive interval timer.
5181
5182         body:
5183             application/json:

```

```

5183         schema: /
5184         {
5185             "$schema": "http://json-schemas.org/draft-04/schema#",
5186             "description" : "Copyright (c) 2016, 2017 Open Connectivity Foundation, Inc. All
5187 rights reserved.",
5188             "id": "http://www.openconnectivity.org/ocf-apis/core/schemas/oic.wk.ping-
5189 schema.json#",
5190             "definitions": {
5191                 "oic.wk.ping": {
5192                     "type": "object",
5193                     "properties": {
5194                         "in": {
5195                             "type": "integer",
5196                             "readOnly": false,
5197                             "description": "Indicates the interval for which connection shall be kept
5198 alive"
5199                         }
5200                     }
5201                 },
5202             },
5203             "type": "object",
5204             "allOf": [
5205                 { "$ref": "oic.core-schema.json#/definitions/oic.core" },
5206                 { "$ref": "#/definitions/oic.wk.ping" }
5207             ],
5208             "required": [
5209                 "in"
5210             ]
5211         }
5212
5213         example: /
5214         {
5215             "in": 16
5216         }
5217

```

## 5218 D.8.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
in	integer		Read Write	Indicates the interval for which connection shall be kept alive

## 5219 D.8.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/oic/ping		get	post		

## 5220 D.9 Discoverable Resources Baseline Interface

### 5221 D.9.1 Introduction

5222 Baseline representation of /oic/res; list of discoverable resources

### 5223 D.9.2 Wellknown URI

5224 /oic/res

### 5225 D.9.3 Resource Type

5226 The resource type (rt) is defined as: oic.wk.res.

### 5227 D.9.4 RAML Definition

5228 `##RAML 0.8`

5229 `title: Discoverable Resources`

5230 `version: v1-20160622`

```

5231 traits:
5232   - interface-11 :
5233     queryParameters:
5234       if:
5235         enum: ["oic.if.11"]
5236   - interface-baseline :
5237     queryParameters:
5238       if:
5239         enum: ["oic.if.baseline"]
5240
5241 /oic-res-BaselineInterfaceURI:
5242   description: |
5243     Baseline representation of /oic/res; list of discoverable resources
5244
5245   is : ['interface-baseline']
5246   get:
5247     description: |
5248       Retrieve the discoverable resource set, baseline interface
5249
5250   responses :
5251     200:
5252       body:
5253         application/json:
5254           schema: /
5255             {
5256               "$schema": "http://json-schema.org/draft-v4/schema#",
5257               "description" : "Copyright (c) 2016, 2017 Open Connectivity Foundation, Inc. All
rights reserved.",
5258               "id": "http://www.openconnectivity.org/ocf-apis/core/schemas/oic.wk.res-
schema.json#",
5259               "definitions": {
5260                 "oic.res-baseline": {
5261                   "type": "object",
5262                   "properties": {
5263                     "rt": {
5264                       "type": "array",
5265                       "items" : {
5266                         "type" : "string",
5267                         "maxLength": 64
5268                       },
5269                       "minItems" : 1,
5270                       "readOnly": true,
5271                       "description": "Resource Type"
5272                     },
5273                     "if": {
5274                       "type": "array",
5275                       "items": {
5276                         "type" : "string",
5277                         "enum" : ["oic.if.baseline", "oic.if.11"]
5278                       },
5279                       "minItems": 1,
5280                       "readOnly": true,
5281                       "description": "The interface set supported by this resource"
5282                     },
5283                     "n": {
5284                       "type": "string",
5285                       "maxLength": 64,
5286                       "readOnly": true,
5287                       "description": "Human friendly name"
5288                     },
5289                     "mpro": {
5290                       "readOnly": true,
5291                       "description": "Supported messaging protocols",
5292

```

```

5294         "type": "string",
5295         "maxLength": 64
5296     },
5297     "links": {
5298         "type": "array",
5299         "items": {
5300             "$ref": "oic.oic-link-schema.json#/definitions/oic.oic-link"
5301         }
5302     },
5303 },
5304 "required": ["rt", "if", "links"]
5305 }
5306 },
5307 "description": "The list of resources expressed as OIC links",
5308 "type": "array",
5309 "items": {
5310     "$ref": "#/definitions/oic.res-baseline"
5311 }
5312 }
5313
5314 example: /
5315 [
5316     {
5317         "rt": ["oic.wk.res"],
5318         "if": ["oic.if.baseline", "oic.if.ll" ],
5319         "links":
5320         [
5321             {
5322                 "href": "/humidity",
5323                 "rt": ["oic.r.humidity"],
5324                 "if": ["oic.if.s"],
5325                 "p": {"bm": 3},
5326                 "eps": [
5327                     {"ep": "coaps://[fe80::b1d6]:1111", "pri": 2},
5328                     {"ep": "coaps://[fe80::b1d6]:1122"},
5329                     {"ep": "coap+tcp://[2001:db8:a::123]:2222", "pri": 3}
5330                 ]
5331             },
5332             {
5333                 "href": "/temperature",
5334                 "rt": ["oic.r.temperature"],
5335                 "if": ["oic.if.s"],
5336                 "p": {"bm": 3},
5337                 "eps": [
5338                     {"ep": "coaps://[2001:db8:a::123]:2222"}
5339                 ]
5340             }
5341         ]
5342     }
5343 ]
5344

```

## D.9.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
rt	array: see schema	yes	Read Only	Resource Type
n	string		Read Only	Human friendly name
links	array: see schema	yes		
mpro	string		Read Only	Supported messaging protocols
if	array: see schema	yes	Read Only	The interface set supported by this resource

5346 **D.9.6 CRUDN behavior**

Resource	Create	Read	Update	Delete	Notify
/oic/res		get			

5347 **D.10 Discoverable Resources Link List interface**

5348 **D.10.1 Introduction**

5349 Link list representation of /oic/res; list of discoverable resources

5350 **D.10.2 Wellknown URI**

5351 /oic/res

5352 **D.10.3 Resource Type**

5353 The resource type (rt) is defined as: oic.wk.res.

5354 **D.10.4 RAML Definition**

```
5355 #%RAML 0.8
5356 title: Discoverable Resources
5357 version: v1-20160622
5358 traits:
5359   - interface-ll :
5360       queryParameters:
5361         if:
5362           enum: ["oic.if.ll"]
5363   - interface-baseline :
5364       queryParameters:
5365         if:
5366           enum: ["oic.if.baseline"]
5367
5368 /oic-res-llInterfaceURI:
5369   description: |
5370     Link list representation of /oic/res; list of discoverable resources
5371
5372   is : ['interface-ll']
5373   get:
5374     description: |
5375       Retrieve the discoverable resource set, link list interface
5376
5377   responses :
5378     200:
5379       body:
5380         application/json:
5381           schema: /
5382             {
5383               "$schema": "http://json-schema.org/draft-v4/schema#",
5384               "description" : "Copyright (c) 2016, 2017 Open Connectivity Foundation, Inc. All
rights reserved.",
5385               "id": "http://www.openconnectivity.org/ocf-apis/core/schemas/oic.wk.res-schema-
ll.json#",
5386               "description": "The list of resources expressed as OCF links without di",
5387               "definitions": {
5388                 "oic.res-ll": {
5389                   "$ref": "oic.oic-link-schema.json#/definitions/oic.oic-link"
5390                 }
5391             },
5392             "type": "array",
5393             "items": {
```

```

5396         "$ref": "#/definitions/oic.res-ll"
5397     }
5398 }
5399
5400 example: /
5401 [
5402     {
5403         "href": "/humidity",
5404         "rt": ["oic.r.humidity"],
5405         "if": ["oic.if.s"],
5406         "p": {"bm": 3},
5407         "eps": [
5408             {"ep": "coaps://[fe80::bld6]:1111", "pri": 2},
5409             {"ep": "coaps://[fe80::bld6]:1122"},
5410             {"ep": "coaps+tcp://[2001:db8:a::123]:2222", "pri": 3}
5411         ]
5412     },
5413     {
5414         "href": "/temperature",
5415         "rt": ["oic.r.temperature"],
5416         "if": ["oic.if.s"],
5417         "p": {"bm": 3},
5418         "eps": [
5419             {"ep": "coaps://[2001:db8:a::123]:2222"}
5420         ]
5421     }
5422 ]
5423

```

#### D.10.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
rt	array: see schema	yes		Resource Type
di	multiple types: see schema			Unique identifier for device (UUID)
title	string			A title for the link relation. Can be used by the UI to provide a context
eps	array: see schema			the Endpoint information of the target Resource
pri (eps)	integer			The priority among multiple Endpoints as specified in 10.2.3
ep (eps)	string			URI with Transport Protocol Suites + Endpoint Locator as specified in 10.2.1
ins	multiple types: see schema			The instance identifier for this web link in an array of web links - used in collections

p	object: schema see			Specifies the framework policies on the Resource referenced by the target URI
bm (p)	integer	yes		Specifies the framework policies on the Resource referenced by the target URI for e.g. observable and discoverable
href	string	yes		This is the target URI, it can be specified as a Relative Reference or fully-qualified URI. Relative Reference should be used along with the di parameter to make it unique.
rel	multiple types: see schema			The relation of the target URI referenced by the link to the context URI
type	array: schema see			A hint at the representation of the resource referenced by the target URI. This represents the media types that are used for both accepting and emitting
anchor	string			This is used to override the context URI e.g. override the URI of the containing collection
if	array: schema see	yes		The interface set supported by this resource

5425

#### D.10.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/oic/res		get			

## D.10.7 Referenced JSON schemas

### D.10.8 oic.oic-link-schema.json

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "description": "Copyright (c) 2016, 2017 Open Connectivity Foundation, Inc. All rights
reserved.",
  "id": "http://www.openconnectivity.org/ocf-apis/core/schemas/oic.oic-link-schema.json#",
  "definitions": {
    "oic.oic-link": {
      "type": "object",
      "properties": {
        "href": {
          "type": "string",
          "maxLength": 256,
          "description": "This is the target URI, it can be specified as a Relative Reference or
fully-qualified URI. Relative Reference should be used along with the di parameter to make it
unique.",
          "format": "uri"
        },
        "rel": {
          "oneOf": [
            {
              "type": "array",
              "items": {
                "type": "string",
                "maxLength": 64
              },
              "minItems": 1,
              "default": ["hosts"]
            },
            {
              "type": "string",
              "maxLength": 64,
              "default": "hosts"
            }
          ],
          "description": "The relation of the target URI referenced by the link to the context URI"
        },
        "rt": {
          "type": "array",
          "items": {
            "type": "string",
            "maxLength": 64
          },
          "minItems": 1,
          "description": "Resource Type"
        },
        "if": {
          "type": "array",
          "items": {
            "type": "string",
            "enum": ["oic.if.baseline", "oic.if.ll", "oic.if.b", "oic.if.rw", "oic.if.r",
"oic.if.a", "oic.if.s" ]
          },
          "minItems": 1,
          "description": "The interface set supported by this resource"
        },
        "di": {
          "$ref": "oic.types-schema.json#/definitions/uuid",
          "description": "Unique identifier for device (UUID)"
        },
        "p": {
          "description": "Specifies the framework policies on the Resource referenced by the target
URI",
          "type": "object",
          "properties": {
            "bm": {
              "description": "Specifies the framework policies on the Resource referenced by the
target URI for e.g. observable and discoverable",
```



```

5495         "type": "integer"
5496     },
5497 },
5498 "required" : ["bm"]
5499 },
5500 "title": {
5501     "type": "string",
5502     "maxLength": 64,
5503     "description": "A title for the link relation. Can be used by the UI to provide a
5504 context"
5505 },
5506 "anchor": {
5507     "type": "string",
5508     "maxLength": 256,
5509     "description": "This is used to override the context URI e.g. override the URI of the
5510 containing collection",
5511     "format": "uri"
5512 },
5513 "ins": {
5514     "oneOf": [
5515         {
5516             "type": "integer",
5517             "description": "An ordinal number that is not repeated - must be unique in the
5518 collection context"
5519         },
5520         {
5521             "type": "string",
5522             "maxLength": 256,
5523             "format": "uri",
5524             "description": "Any unique string including a URI"
5525         },
5526         {
5527             "$ref": "oic.types-schema.json#/definitions/uuid",
5528             "description": "Unique identifier (UUID)"
5529         }
5530     ],
5531     "description": "The instance identifier for this web link in an array of web links - used
5532 in collections"
5533 },
5534 "type": {
5535     "type": "array",
5536     "description": "A hint at the representation of the resource referenced by the target
5537 URI. This represents the media types that are used for both accepting and emitting",
5538     "items": {
5539         "type": "string",
5540         "maxLength": 64
5541     },
5542     "minItems": 1,
5543     "default": "application/cbor"
5544 },
5545 "eps": {
5546     "type": "array",
5547     "description": "the Endpoint information of the target Resource",
5548     "items": {
5549         "type": "object",
5550         "properties": {
5551             "ep": {
5552                 "type": "string",
5553                 "format": "uri",
5554                 "description": "URI with Transport Protocol Suites + Endpoint Locator as specified
5555 in 10.2.1"
5556             },
5557             "pri": {
5558                 "type": "integer",
5559                 "minimum": 1,
5560                 "description": "The priority among multiple Endpoints as specified in 10.2.3"
5561             }
5562         }
5563     }
5564 },
5565 },

```

```

5566         "required": [ "href", "rt", "if" ]
5567     }
5568 },
5569 "type": "object",
5570 "allOf": [
5571     { "$ref": "#/definitions/oic.oic-link" }
5572 ]
5573 }
5574

```

## 5575 **D.11 Scenes (Top level)**

### 5576 **D.11.1 Introduction**

5577 Toplevel Scene resource. This resource is a generic collection resource. The rts value shall contain  
5578 oic.wk.scenecollection resource types.

### 5579 **D.11.2 Example URI**

5580 /SceneListResURI

### 5581 **D.11.3 Resource Type**

5582 The resource type (rt) is defined as: oic.wk.scenelist.

### 5583 **D.11.4 RAML Definition**

```

5584 #%RAML 0.8
5585 title: Scene
5586 version: v1-20160622
5587 traits:
5588   - interface :
5589       queryParameters:
5590           if:
5591               enum: ["oic.if.a", "oic.if.ll", "oic.if.baseline"]
5592
5593 /SceneListResURI:
5594   description: |
5595     Toplevel Scene resource.
5596     This resource is a generic collection resource.
5597     The rts value shall contain oic.wk.scenecollection resource types.
5598
5599   get:
5600       description: |
5601         Provides the current list of web links pointing to scenes
5602
5603   responses :
5604     200:
5605       body:
5606         application/json:
5607           schema: /
5608             {
5609               "$schema": "http://json-schema.org/draft-04/schema#",
5610               "description" : "Copyright (c) 2016 Open Connectivity Foundation, Inc. All rights
5611 reserved.",
5612               "id": "http://www.openconnectivity.org/ocf-apis/core/schemas/oic.collection-
5613 schema.json#",
5614               "title": "Collection",
5615               "definitions": {
5616                 "oic.collection.setoflinks": {
5617                   "description": "A set (array) of simple or individual OIC Links. In
5618 addition to properties required for an OIC Link, the identifier for that link in this set is also
5619 required",
5620                   "type": "array",

```

```

5621         "items": {
5622             "$ref": "oic.oic-link-schema.json#/definitions/oic.oic-link"
5623         },
5624     },
5625     "oic.collection.alllinks": {
5626         "description": "All forms of links in a collection",
5627         "oneOf": [
5628             {
5629                 "$ref": "#/definitions/oic.collection.setoflinks"
5630             }
5631         ],
5632     },
5633     "oic.collection": {
5634         "type": "object",
5635         "description": "A collection is a set (array) of tagged-link or set
5636 (array) of simple links along with additional properties to describe the collection itself",
5637         "properties": {
5638             "id": {
5639                 "anyOf": [
5640                     {
5641                         "type": "integer",
5642                         "description": "A number that is unique to that
5643 collection; like an ordinal number that is not repeated"
5644                     },
5645                     {
5646                         "type": "string",
5647                         "description": "A unique string that could be a hash or
5648 similarly unique"
5649                     }
5650                 ],
5651                 "$ref": "oic.types-schema.json#/definitions/uuid",
5652                 "description": "A unique string that could be a UUIDv4"
5653             },
5654             "description": "ID for the collection. Can be an value that is
5655 unique to the use context or a UUIDv4"
5656         },
5657         "di": {
5658             "$ref": "oic.types-schema.json#/definitions/uuid",
5659             "description": "The device ID which is an UUIDv4 string; used for
5660 backward compatibility with Spec A definition of /oic/res"
5661         },
5662         "rts": {
5663             "$ref": "oic.core-
5664 schema.json#/definitions/oic.core/properties/rt",
5665             "description": "Defines the list of allowable resource types (for
5666 Target and anchors) in links included in the collection; new links being created can only be from
5667 this list"
5668         },
5669         "drel": {
5670             "type": "string",
5671             "description": "When specified this is the default relationship
5672 to use when an OIC Link does not specify an explicit relationship with *rel* parameter"
5673         },
5674         "links": {
5675             "$ref": "#/definitions/oic.collection.alllinks"
5676         }
5677     }
5678 },
5679 },
5680 "type": "object",
5681 "allOf": [
5682     {"$ref": "oic.core-schema.json#/definitions/oic.core"},
5683     {"$ref": "#/definitions/oic.collection"}
5684 ]
5685 }
5686
5687 example: /
5688 {
5689     "rt": ["oic.wk.scenelist"],
5690     "n": "list of scene Collections",

```

```

5691         "rts": ["oic.wk.scenecollection"],
5692         "links": [
5693             ]
5694     }
5695

```

#### D.11.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
drel	string			When specified this is the default relationship to use when an OIC Link does not specify an explicit relationship with *rel* parameter
links	multiple types: see schema			
id	multiple types: see schema			ID for the collection. Can be an value that is unique to the use context or a UUIDv4
rts	multiple types: see schema			Defines the list of allowable resource types (for Target and anchors) in links included in the collection; new links being created can only be from this list
di	multiple types: see schema			The device ID which is an UUIDv4 string; used for backward compatibility with Spec A definition of /oic/res

#### D.11.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/SceneListResURI		get			

### D.12 Scene Collections

#### D.12.1 Introduction

Collection that models a set of Scenes. This resource is a generic collection resource with additional parameters. The rts value shall contain oic.scenemember resource types. The additional parameters are lastScene, this is the scene value last set by any OCF Client sceneValues, this is the list of available scenes lastScene shall be listed in sceneValues.

## D.12.2 Example URI

/SceneCollectionResURI

## D.12.3 Resource Type

The resource type (rt) is defined as: oic.wk.scenecollection.

## D.12.4 RAML Definition

```
##RAML 0.8
title: Scene
version: v1-20160622

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

/SceneCollectionResURI:
  description: |
    Collection that models a set of Scenes.
    This resource is a generic collection resource with additional parameters.
    The rts value shall contain oic.scenemember resource types.
    The additional parameters are
    lastScene, this is the scene value last set by any OCF Client
    sceneValues, this is the list of available scenes
    lastScene shall be listed in sceneValues.

  get:
    description: |
      Provides the current list of web links pointing to scenes

    responses :
      200:
        body:
          application/json:
            schema: /
              {
                "$schema": "http://json-schema.org/draft-04/schema#",
                "description" : "Copyright (c) 2016, 2017 Open Connectivity Foundation, Inc. All
rights reserved.",
                "id": "http://www.openconnectivity.org/ocf-apis/core/schemas/oic.sceneCollection-
schema.json#",
                "title" : "Scene Collection",
                "definitions": {
                  "oic.sceneCollection": {
                    "type": "object",
                    "properties": {
                      "lastScene": {
                        "type": "string",
                        "description": "Last selected Scene, shall be part of sceneValues"
                      },
                      "sceneValues": {
                        "type": "string",
                        "readOnly": true,
                        "description": "All available scene values"
                      },
                      "n": {
                        "type": "string",
                        "description": "Used to name the Scene collection"
                      },
                      "id": {
                        "type": "string",
```

```

5763         "description" : "A unique string that could be a hash or
5764 similarly unique"
5765     },
5766     "rts": {
5767         "$ref": "oic.core-schema.json#/definitions/oic.core/properties/rt",
5768         "description": "Defines the list of allowable resource types in links
5769 included in the collection; new links being created can only be from this list"
5770     },
5771     "links": {
5772         "type": "array",
5773         "description": "Array of OIC web links that are reference from this
5774 collection",
5775         "items" : {
5776             "allOf": [
5777                 { "$ref": "oic.oic-link-schema.json#/definitions/oic.oic-link" },
5778                 { "required" : [ "ins" ] }
5779             ]
5780         }
5781     },
5782     "required": [ "lastScene", "sceneValues", "rts", "id" ]
5783 },
5784 },
5785 },
5786 "type": "object",
5787 "allOf" : [
5788     { "$ref": "oic.core-schema.json#/definitions/oic.core" },
5789     { "$ref": "#/definitions/oic.sceneCollection" }
5790 ]
5791 }
5792 }
5793
5794 example: /
5795 {
5796     "lastScene": "off",
5797     "sceneValues": "off,Reading,TVWatching",
5798     "rt":         ["oic.wk.sceneCollection"],
5799     "n":          "My Scenes for my living room",
5800     "id":         "0685B960-736F-46F7-BEC0-9E6CBD671ADC1",
5801     "rts":        ["oic.wk.sceneMember"],
5802     "links": [
5803     ]
5804 }
5805
5806 post:
5807     description: |
5808         Provides the action to change the last set scene selection.
5809         Calling this method shall update all scene members to the prescribed member value.
5810         When this method is called with the same value as the current lastScene value
5811         then all scene members shall be updated.
5812
5813     body:
5814         application/json:
5815             schema: /
5816                 {
5817                     "$schema": "http://json-schema.org/draft-04/schema#",
5818                     "description" : "Copyright (c) 2016, 2017 Open Connectivity Foundation, Inc. All rights
5819 reserved.",
5820                     "id": "http://www.openconnectivity.org/ocf-apis/core/schemas/oic.sceneCollection-
5821 schema.json#",
5822                     "title": "Scene Collection",
5823                     "definitions": {
5824                         "oic.sceneCollection": {
5825                             "type": "object",
5826                             "properties": {
5827                                 "lastScene": {
5828                                     "type": "string",
5829                                     "description": "Last selected Scene, shall be part of sceneValues"

```

```

5830         },
5831         "sceneValues": {
5832             "type": "string",
5833             "readOnly": true,
5834             "description": "All available scene values"
5835         },
5836         "n": {
5837             "type": "string",
5838             "description": "Used to name the Scene collection"
5839         },
5840         "id": {
5841             "type": "string",
5842             "description": "A unique string that could be a hash or
5843 similarly unique"
5844         },
5845         "rts": {
5846             "$ref": "oic.core-schema.json#/definitions/oic.core/properties/rt",
5847             "description": "Defines the list of allowable resource types in links included
5848 in the collection; new links being created can only be from this list"
5849         },
5850         "links": {
5851             "type": "array",
5852             "description": "Array of OIC web links that are reference from this
5853 collection",
5854             "items": {
5855                 "allOf": [
5856                     { "$ref": "oic.oic-link-schema.json#/definitions/oic.oic-link" },
5857                     { "required": [ "ins" ] }
5858                 ]
5859             }
5860         },
5861         "required": [ "lastScene" ]
5862     },
5863 },
5864 },
5865
5866     "type": "object",
5867     "allOf": [
5868         { "$ref": "oic.core-schema.json#/definitions/oic.core" },
5869         { "$ref": "#/definitions/oic.sceneCollection" }
5870     ]
5871 }
5872
5873 example: /
5874 {
5875     "lastScene": "Reading"
5876 }
5877
5878 responses :
5879 200:
5880     description: |
5881         Indicates that the value is changed.
5882         The changed properties are provided in the response.
5883
5884 body:
5885     application/json:
5886         schema: /
5887         {
5888             "$schema": "http://json-schema.org/draft-04/schema#",
5889             "description": "Copyright (c) 2016, 2017 Open Connectivity Foundation, Inc. All
5890 rights reserved.",
5891             "id": "http://www.openconnectivity.org/ocf-apis/core/schemas/oic.sceneCollection-
5892 schema.json#",
5893             "title": "Scene Collection",
5894             "definitions": {
5895                 "oic.sceneCollection": {
5896                     "type": "object",

```

```

5897         "properties": {
5898             "lastScene": {
5899                 "type": "string",
5900                 "description": "Last selected Scene, shall be part of sceneValues"
5901             },
5902             "sceneValues": {
5903                 "type": "string",
5904                 "readOnly": true,
5905                 "description": "All available scene values"
5906             },
5907             "n": {
5908                 "type": "string",
5909                 "description": "Used to name the Scene collection"
5910             },
5911             "id": {
5912                 "type": "string",
5913                 "description": "A unique string that could be a hash or
5914 similarly unique"
5915             },
5916             "rts": {
5917                 "$ref": "oic.core-schema.json#/definitions/oic.core/properties/rt",
5918                 "description": "Defines the list of allowable resource types in links
5919 included in the collection; new links being created can only be from this list"
5920             },
5921             "links": {
5922                 "type": "array",
5923                 "description": "Array of OIC web links that are reference from this
5924 collection",
5925                 "items": {
5926                     "allOf": [
5927                         { "$ref": "oic.oic-link-schema.json#/definitions/oic.oic-link" },
5928                         { "required": [ "ins" ] }
5929                     ]
5930                 }
5931             },
5932             "required": [ "lastScene" ]
5933         }
5934     },
5935     "type": "object",
5936     "allOf": [
5937         { "$ref": "oic.core-schema.json#/definitions/oic.core" },
5938         { "$ref": "oic.oic-link-schema.json#/definitions/oic.oic-link" },
5939         { "$ref": "oic.oic-link-schema.json#/definitions/oic.oic-link-collection" }
5940     ]
5941 }
5942
5943
5944 example: /
5945 {
5946     "lastScene": "Reading"
5947 }
5948

```

## D.12.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
lastScene	string	yes		Last selected Scene, shall be part of sceneValues
links	array: see schema			Array of OIC web links that are reference from this collection
sceneValues	string	yes	Read Only	All available scene values



n	string			Used to name the Scene collection
rts	multiple types: see schema	yes		Defines the list of allowable resource types in links included in the collection; new links being created can only be from this list
id	string	yes		A unique string that could be a hash or similarly unique

## D.12.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/SceneCollectionResURI		get	post		

## D.13 Scene Member

### D.13.1 Introduction

Collection that models a scene member.

### D.13.2 Example URI

/SceneMemberResURI

### D.13.3 Resource Type

The resource type (rt) is defined as: oic.wk.scenemember.

### D.13.4 RAML Definition

```

#%RAML 0.8
title: Scene
version: v1-20160622
traits:
  - interface :
      queryParameters:
        if:
          enum: ["oic.if.a", "oic.if.ll", "oic.if.baseline"]

/SceneMemberResURI:
  description: |
    Collection that models a scene member.

  get:
    description: |
      Provides the scene member

  responses :
    200:
      body:
        application/json:
          schema: /

```

```

5981     {
5982         "$schema": "http://json-schema.org/draft-04/schema#",
5983         "description": "Copyright (c) 2016, 2017 Open Connectivity Foundation, Inc. All
5984 rights reserved.",
5985         "id": "http://www.openconnectivity.org/ocf-apis/core/schemas/oic.sceneMember-
5986 schema.json#",
5987         "title": "Scene Member",
5988         "definitions": {
5989             "oic.sceneMember": {
5990                 "type": "object",
5991                 "properties": {
5992                     "n": {
5993                         "type": "string",
5994                         "description": "Used to name the Scene collection"
5995                     },
5996                     "id": {
5997                         "type": "string",
5998                         "description": "Can be an value that is unique to the use context or a
5999 UUIDv4"
6000                     },
6001                     "SceneMappings": {
6002                         "type": "array",
6003                         "description": "array of mappings per scene, can be 1",
6004                         "items": {
6005                             "type": "object",
6006                             "properties": {
6007                                 "scene": {
6008                                     "type": "string",
6009                                     "description": "Specifies a scene value that will acted upon"
6010                                 },
6011                                 "memberProperty": {
6012                                     "type": "string",
6013                                     "readOnly": true,
6014                                     "description": "property name that will be mapped"
6015                                 },
6016                                 "memberValue": {
6017                                     "type": "string",
6018                                     "readOnly": true,
6019                                     "description": "value of the Member Property"
6020                                 }
6021                             },
6022                             "required": [ "scene", "memberProperty", "memberValue" ]
6023                         }
6024                     },
6025                     "link": {
6026                         "type": "string",
6027                         "description": "web link that points at a resource",
6028                         "$ref": "oic.oic-link-schema.json#/definitions/oic.oic-link"
6029                     }
6030                 },
6031                 "required": [ "link" ]
6032             }
6033         },
6034         "type": "object",
6035         "allOf": [
6036             { "$ref": "oic.core-schema.json#/definitions/oic.core" },
6037             { "$ref": "#/definitions/oic.sceneMember" }
6038         ]
6039     }
6040 }
6041
6042
6043 example: /
6044 {
6045     "rt": ["oic.wk.scenemember"],
6046     "id": "0685B960-FFFF-46F7-BEC0-9E6234671ADC1",
6047     "n": "my binary switch (for light bulb) mappings",
6048     "link": {
6049         "href": "binarySwitch",
6050         "rt": [ "oic.r.switch.binary" ],

```

```

6051         "if": ["oic.if.a", "oic.if.baseline"],
6052         "eps": [
6053             { "ep": "coap://[fe80::b1d6]:1111", "pri": 2 },
6054             { "ep": "coaps://[fe80::b1d6]:1122" },
6055             { "ep": "coap+tcp://[2001:db8:a::123]:2222", "pri": 3 }
6056         ],
6057     },
6058     "sceneMappings": [
6059         {
6060             "scene": "off",
6061             "memberProperty": "value",
6062             "memberValue": true
6063         },
6064         {
6065             "scene": "Reading",
6066             "memberProperty": "value",
6067             "memberValue": false
6068         },
6069         {
6070             "scene": "TVWatching",
6071             "memberProperty": "value",
6072             "memberValue": true
6073         }
6074     ]
6075 }
6076

```

### D.13.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
SceneMappings	array: see schema			array of mappings per scene, can be 1
memberValue (SceneMappings)	string	yes	Read Only	value of the Member Property
memberProperty (SceneMappings)	string	yes	Read Only	property name that will be mapped
scene (SceneMappings)	string	yes		Specifies a scene value that will acted upon
link	string	yes		web link that points at a resource
id	string			Can be an value that is unique to the use context or a UUIDv4
n	string			Used to name the Scene collection

### D.13.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/SceneMemberResURI		get			

## D.14 Resource directory resource

### D.14.1 Introduction

Resource to be exposed by any Device that can act as a Resource Directory. 1) Provides selector criteria (e.g., integer) with GET request 2) Publish or Update a Link in /oic/res with POST request 3) Delete a Link in /oic/res with DELETE request

6084 **D.14.2 Wellknown URI**

6085 /oic/rd

6086 **D.14.3 Resource Type**

6087 The resource type (rt) is defined as: oic.wk.rd.

6088 **D.14.4 RAML Definition**

6089 `##RAML 0.8`

6090 `title: Resource Directory`

6091 `version: v1-20160622`

6092 `traits:`

6093 `- rddelete-di :`

6094  `queryParameters:`

6095  `di:`

6096  `description: This is used to determine which set of links to operata on. (Need`

6097 `authentication to ensure that there is no spoofing). If instance is ommitted then the entire set of`

6098 `links from this device ID is deleted`

6099 `Example: DELETE /oic/rd?di="0685B960-736F-46F7-BEC0-9E6CBD671ADC1"`

6100

6101 `- rddelete-ins :`

6102  `queryParameters:`

6103  `ins:`

6104  `description: Instance of the link to delete`

6105 `Value of parameter is a string where instance to be deleted are comma separated`

6106 `Example: DELETE /oic/rd?di="0685B960-736F-46F7-BEC0-9E6CBD671ADC1";ins="20"`

6107

6108 `- rdgetinterface :`

6109  `queryParameters:`

6110  `if:`

6111  `enum: ["oic.if.baseline"]`

6112  `description: Interface is optional since there is only one interface supported for the`

6113 `Resource Type`

6114 `Both for RD selectin and for publish.`

6115 `Example: GET /oic/rd?if=oic.if.baseline`

6116

6117 `- rdpostinterface :`

6118  `queryParameters:`

6119  `rt:`

6120  `enum: ["oic.wk.rdpub"]`

6121  `description: Used in POST request to ask the RD to add the Links in payload to /oic/res.`

6122 `Example: POST /oic/rd?rt=oic.wk.rdpub`

6123

6124

6125 `/oic/rd:`

6126  `description: |`

6127  `Resource to be exposed by any Device that can act as a Resource Directory.`

6128  `1) Provides selector criteria (e.g., integer) with GET request`

6129  `2) Publish or Update a Link in /oic/res with POST request`

6130  `3) Delete a Link in /oic/res with DELETE request`

6131

6132  `get:`

6133  `description: |`

6134  `Get the attributes of the Resource Directory for selection purposes.`

6135

6136  `is : ['rdgetinterface']`

6137  `responses :`

6138  `200:`

6139  `description: |`

```

6140         Respond with the selector criteria - either the set of attributes or the bias factor
6141
6142     body:
6143         application/json:
6144             schema: /
6145                 {
6146                     "$schema": "http://json-schema.org/draft-04/schema#",
6147                     "description": "Copyright (c) 2016, 2017 Open Connectivity Foundation, Inc. All
6148 rights reserved.",
6149                     "id": "http://www.openconnectivity.org/ocf-apis/core/schemas/oic.rd.selection-
6150 schema.json#",
6151                     "title": "RD Selection",
6152                     "definitions": {
6153                         "oic.rd.attributes": {
6154                             "type": "object",
6155                             "oneOf": [
6156                                 {
6157                                     "properties": {
6158                                         "sel": {
6159                                             "type": "integer",
6160                                             "minimum": 0,
6161                                             "maximum": 100,
6162                                             "description": "A bias factor calculated by the Resource directory -
6163 the value is in the range of 0 to 100 - 0 implies that RD is not to be selected. Client chooses RD
6164 with highest bias factor or randomly between RDs that have same bias factor"
6165                                         }
6166                                     },
6167                                     "required": ["sel"]
6168                                 },
6169                                 {
6170                                     "properties": {
6171                                         "sel": {
6172                                             "description": "Selection criteria that a device wanting to publish to
6173 any RD can use to choose this Resource Directory over others that are discovered",
6174                                             "type": "object",
6175                                             "properties": {
6176                                                 "pwr": {
6177                                                     "type": "string",
6178                                                     "enum": [ "ac", "batt", "safe" ],
6179                                                     "description": "A hint about how the RD is powered. If AC then this
6180 is stronger than battery powered. If source is reliable (safe) then appropriate mechanism for
6181 managing power failure exists"
6182                                                 },
6183                                                 "conn": {
6184                                                     "type": "string",
6185                                                     "enum": [ "wrld", "wrls" ],
6186                                                     "description": "A hint about the networking connectivity of the RD.
6187 *wrld* if wired connected and *wrls* if wireless connected."
6188                                                 },
6189                                                 "bw": {
6190                                                     "type": "string",
6191                                                     "description": "Qualitative bandwidth of the connection",
6192                                                     "enum": [ "high", "low", "lossy" ]
6193                                                 },
6194                                                 "mf": {
6195                                                     "type": "integer",
6196                                                     "description": "Memory factor - Ratio of available memory to total
6197 memory expressed as a percentage"
6198                                                 },
6199                                                 "load": {
6200                                                     "type": "array",
6201                                                     "items": {
6202                                                         "type": "number"
6203                                                     },
6204                                                     "minItems": 3,
6205                                                     "maxItems": 3,
6206                                                     "description": "Current load capacity of the RD. Expressed as a
6207 load factor 3-tuple (upto two decimal points each). Load factor is based on request processed in a
6208 1 minute, 5 minute window and 15 minute window"

```

```

6209         }
6210     }
6211 }
6212 },
6213 "required": ["sel"]
6214 }
6215 ]
6216 }
6217 },
6218 "type": "object",
6219 "allOf": [
6220 { "$ref": "oic.core-schema.json#/definitions/oic.core" },
6221 { "$ref": "#/definitions/oic.rd.attributes" }
6222 ]
6223 }
6224
6225 example: /
6226 {
6227     "rt": ["oic.wk.rd"],
6228     "if": ["oic.if.baseline"],
6229     "sel": 50
6230 }
6231
6232 post:
6233     description: |
6234         Publish the resource information for the first time or Update the existing one in /oic/res.
6235         Appropriates parts of the information, i.e., Links of the published Resources will be
6236 discovered through /oic/res.
6237         1) When a Device first publishes a Link, the request payload to RD may include the Links
6238 without "ins" Parameter.
6239         2) Upon granting the request, the RD assigns a unique instance value identifying the Link
6240 among all the Links it advertises
6241         and sends back the instance value in "ins" Parameter in the Link to the publishing Device.
6242         3) When later the publishing Device updates the existing Link, i.e., changing its Endpoint
6243 information,
6244         the request payload to RD needs to include the instance value in "ins" Parameter to
6245 identify the Link to update.
6246
6247     is : ['rdpostinterface']
6248     body:
6249         application/json:
6250             schema: /
6251                 {
6252                     "$schema": "http://json-schema.org/draft-04/schema#",
6253                     "description": "Copyright (c) 2016,2017 Open Connectivity Foundation, Inc. All rights
6254 reserved.",
6255                     "id": "http://www.openconnectivity.org/ocf-apis/core/schemas/oic.rd.publish-
6256 schema.json#",
6257                     "title": "RD Publish & Update",
6258                     "definitions": {
6259                         "oic.rd.publish": {
6260                             "description": "Publishes resources as OIC Links into the resource directory",
6261                             "properties": {
6262                                 "di": {
6263                                     "$ref": "oic.types-schema.json#/definitions/uuid",
6264                                     "description": "A unique identifier for the publishing Device, i.e., its device
6265 ID"
6266                                 },
6267                                 "links": {
6268                                     "$ref": "oic.collection-schema.json#/definitions/oic.collection.setoflinks"
6269                                 },
6270                                 "ttl": {
6271                                     "type": "integer",
6272                                     "description": "Time to indicate a RD, how long to keep this published item.
6273 After this time (in seconds) elapses, the RD invalidates the links. To keep link alive the
6274 publishing device updates the ttl using the update schema"
6275                                 }
6276                             }
6277                         }
6278                     }
6279                 }

```

```

6276         }
6277     }
6278 },
6279 "type": "object",
6280 "allOf": [
6281     {
6282         "$ref": "oic.core-schema.json#/definitions/oic.core"
6283     },
6284     {
6285         "$ref": "#/definitions/oic.rd.publish"
6286     }
6287 ],
6288 "required": [
6289     "di",
6290     "links",
6291     "ttl"
6292 ]
6293 }
6294
6295 example: /
6296 {
6297     "di": "e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",
6298     "links": [
6299         {
6300             "anchor": "ocf://e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",
6301             "href": "/myLightSwitch",
6302             "rt": ["oic.r.switch.binary"],
6303             "if": ["oic.if.a", "oic.if.baseline"],
6304             "p": {"bm": 3},
6305             "eps": [
6306                 {"ep": "coaps://[2001:db8:a::b1d6]:1111", "pri": 2},
6307                 {"ep": "coaps://[2001:db8:a::b1d6]:1122"},
6308                 {"ep": "coaps+tcp://[2001:db8:a::123]:2222", "pri": 3}
6309             ]
6310         },
6311         {
6312             "anchor": "ocf://e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",
6313             "href": "/myLightBrightness",
6314             "rt": ["oic.r.brightness"],
6315             "if": ["oic.if.a", "oic.if.baseline"],
6316             "p": {"bm": 3},
6317             "eps": [
6318                 {"ep": "coaps://[2001:db8:a::123]:2222"}
6319             ]
6320         }
6321     ],
6322     "ttl": 600
6323 }
6324
6325 responses :
6326 200:
6327     description: |
6328         Respond with the same schema as publish but, when a Link is first published,
6329         with the additional "ins" Parameter in the Link.
6330         This value is used by the receiver to manage that OCF Link instance.
6331
6332     body:
6333         application/json:
6334             schema: /
6335             {
6336                 "$schema": "http://json-schema.org/draft-04/schema#",
6337                 "description": "Copyright (c) 2016,2017 Open Connectivity Foundation, Inc. All
6338 rights reserved.",
6339                 "id": "http://www.openconnectivity.org/ocf-apis/core/schemas/oic.rd.publish-
6340 schema.json#",
6341                 "title": "RD Publish & Update",
6342                 "definitions": {

```

```

6343         "oic.rd.publish": {
6344             "description": "Publishes resources as OIC Links into the resource directory",
6345             "properties": {
6346                 "di": {
6347                     "$ref": "oic.types-schema.json#/definitions/uuid",
6348                     "description": "A unique identifier for the publishing Device, i.e., its
6349 device ID"
6350                 },
6351                 "links": {
6352                     "$ref": "oic.collection-schema.json#/definitions/oic.collection.setoflinks"
6353                 },
6354                 "ttl": {
6355                     "type": "integer",
6356                     "description": "Time to indicate a RD, how long to keep this published
6357 item. After this time (in seconds) elapses, the RD invalidates the links. To keep link alive the
6358 publishing device updates the ttl using the update schema"
6359                 }
6360             }
6361         },
6362     },
6363     "type": "object",
6364     "allOf": [
6365         {
6366             "$ref": "oic.core-schema.json#/definitions/oic.core"
6367         },
6368         {
6369             "$ref": "#/definitions/oic.rd.publish"
6370         }
6371     ],
6372     "required": [
6373         "di",
6374         "links",
6375         "ttl"
6376     ]
6377 }
6378
6379 example: /
6380 {
6381     "di": "e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",
6382     "links": [
6383         {
6384             "anchor": "ocf://e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",
6385             "href": "/myLightSwitch",
6386             "rt": ["oic.r.switch.binary"],
6387             "if": ["oic.if.a", "oic.if.baseline"],
6388             "p": {"bm": 3},
6389             "eps": [
6390                 {"ep": "coaps://[2001:db8:a::b1d6]:1111", "pri": 2},
6391                 {"ep": "coaps://[2001:db8:a::b1d6]:1122"},
6392                 {"ep": "coaps+tcp://[2001:db8:a::123]:2222", "pri": 3}
6393             ],
6394             "ins": "11235"
6395         },
6396         {
6397             "anchor": "ocf://e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",
6398             "href": "/myLightBrightness",
6399             "rt": ["oic.r.brightness"],
6400             "if": ["oic.if.a", "oic.if.baseline"],
6401             "p": {"bm": 3},
6402             "eps": [
6403                 {"ep": "coaps://[2001:db8:a::123]:2222"}
6404             ],
6405             "ins": "112358"
6406         }
6407     ],
6408     "ttl": 600
6409 }
6410
6411 delete:

```



```

6412     description: |
6413         Delete a particular OIC Link - the link may be a simple link or a link in a tagged set.
6414
6415     is : ['rddelete-di','rddelete-ins']
6416
6417     responses :
6418         200:
6419             description: |
6420                 The delete succeeded

```

#### D.14.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
sel	object: see schema	yes		Selection criteria that a device wanting to publish to any RD can use to choose this Resource Directory over others that are discovered
mf (sel)	integer			Memory factor - Ratio of available memory to total memory expressed as a percentage
load (sel)	array: see schema			Current load capacity of the RD. Expressed as a load factor 3-tuple (upto two decimal points each). Load factor is based on request processed in a 1 minute, 5 minute window and 15 minute window
bw (sel)	string			Qualitative bandwidth of the connection
pwr (sel)	string			A hint about how the RD is powered. If AC then this is stronger than battery powered. If source is reliable (safe) then appropriate mechanism for managing power failure exists

conn (sel)	string			A hint about the networking connectivity of the RD. *wrd* if wired connected and *wrls* if wireless connected.
---------------	--------	--	--	--

#### D.14.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/oic/rd		get	post	delete	

### D.15 Icon

#### D.15.1 Introduction

This resource describes the attributes associated with an Icon.

#### D.15.2 Example URI

/IconResURI

#### D.15.3 Resource Type

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

#### D.15.4 RAML Definition

```

#%RAML 0.8
title: OICIcon
version: v1.1.0-20161107

traits:
- interface :
    queryParameters:
        if:
            enum: ["oic.if.r", "oic.if.baseline"]

/IconResURI:
    description: |
        This resource describes the attributes associated with an Icon.

    is : ['interface']

    get:
        description: |
            Retrieves the current icon properties.

        responses :
            200:
                body:
                    application/json:
                        schema: /
                            {
                                "id": "http://www.openconnectivity.org/ocf-apis/core/schemas/oic.r.icon.json#",
                                "$schema": "http://json-schema.org/draft-04/schema#",
                                "description" : "Copyright (c) 2017 Open Connectivity Foundation, Inc. All rights
reserved.",
                                "title": "Icon",
                                "definitions": {
                                    "oic.r.icon": {

```

```

6462         "properties": {
6463             "mimetype": {
6464                 "type": "string",
6465                 "maxLength": 64,
6466                 "readOnly": true,
6467                 "description": "Specifies the format of the MIME Type"
6468             },
6469             "width": {
6470                 "type": "integer",
6471                 "minimum": 1,
6472                 "readOnly": true,
6473                 "description": "Specifies the width in pixels"
6474             },
6475             "height": {
6476                 "type": "integer",
6477                 "minimum": 1,
6478                 "readOnly": true,
6479                 "description": "Specifies the height in pixels"
6480             },
6481             "media": {
6482                 "type": "string",
6483                 "maxLength": 256,
6484                 "format": "uri",
6485                 "readOnly": true,
6486                 "description": "Specifies the media URL to icon"
6487             }
6488         }
6489     },
6490     "type": "object",
6491     "allof": [
6492         { "$ref": "oic.core-schema.json#/definitions/oic.core" },
6493         { "$ref": "#/definitions/oic.r.icon" }
6494     ],
6495     "required": ["mimetype", "width", "height", "media"]
6496 }
6497
6498
6499     example: /
6500     {
6501         "rt": ["oic.r.icon"],
6502         "id": "unique_example_id",
6503         "mimetype": "image/png",
6504         "width": 256,
6505         "height": 256,
6506         "media": "http://findbetter.ru/public/uploads/1481662800/2043.png"
6507     }
6508

```

### D.15.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
mimetype	string	yes	Read Only	Specifies the format of the MIME Type
width	integer	yes	Read Only	Specifies the width in pixels
media	string	yes	Read Only	Specifies the media URL to icon
height	integer	yes	Read Only	Specifies the height in pixels

### D.15.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/IconResURI		get			

## D.16 Introspection Resource

### D.16.1 Introduction

This resource provides the means to get the device introspection data specifying all the endpoints of the device. The url hosted by this resource is either a local or an external url.

### D.16.2 Example URI

/IntrospectionResURI

### D.16.3 Resource Type

The resource type (rt) is defined as: oic.wk.introspection.

### D.16.4 RAML Definition

```
##RAML 0.8
title: OICIntrospection
version: v1.0.0-20160707

traits:
  - interface :
      queryParameters:
        if:
          enum: ["oic.if.r", "oic.if.baseline"]

/IntrospectionResURI:
  description: |
    This resource provides the means to get the device introspection data specifying all the
    endpoints of the device.
    The url hosted by this resource is either a local or an external url.

  is : ['interface']
  get:
    responses :
      200:
        body:
          application/json:
            schema: /
              {
                "id": "http://www.openconnectivity.org/ocf-
apis/core/schemas/oic.wk.introspectionInfo.json#",
                "$schema": "http://json-schema.org/draft-04/schema#",
                "description" : "Copyright (c) 2017 Open Interconnect Consortium, Inc. All rights
reserved.",
                "title": "introspection resource",
                "definitions": {
                  "oic.wk.introspectionInfo": {
                    "type": "object",
                    "properties": {
                      "urlInfo": {
                        "type": "array",
                        "description": "The valid range for the value Property",
                        "readOnly": true,
                        "minItems": 1,
                        "items": {
                          "type" : "object",
                          "properties": {
                            "url": {
                              "type": "string",
                              "format": "uri",
                              "description" : "url to download the description"
                            },
                            "protocol": {
                              "type": "string",
```

```

6568         "enum": [ "coap", "coaps", "http", "https", "coap+tcp",
6569 "coaps+tcp" ],
6570         "description" : "protocol to be used to download the introspection"
6571     },
6572     "content-type": {
6573         "type": "string",
6574         "enum": [ "application/json", "application/cbor" ],
6575         "default" : "application/cbor",
6576         "description" : "content-type of the introspection data"
6577     },
6578     "version": {
6579         "type": "integer",
6580         "enum": [ 1 ],
6581         "default" : 1,
6582         "description" : "version the introspection data that can be
6583 downloaded"
6584     }
6585 },
6586 "required" : [ "url","protocol"]
6587 }
6588 }
6589 },
6590 "required" : ["urlInfo"]
6591 }
6592 },
6593 "type": "object",
6594 "allOf": [
6595     { "$ref": "#/definitions/oic.wk.introspectionInfo" },
6596     { "$ref": "oic.core-schema.json#/definitions/oic.core" }
6597 ]
6598 }
6599
6600 example: /
6601 {
6602     "rt" : ["oic.wk.introspection"],
6603     "urlInfo" : [
6604         {
6605             "content-type" : "application/cbor",
6606             "protocol" : "coap",
6607             "url" : "coap://[fe80::1]:1234/IntrospectionExampleURI"
6608         }
6609     ]
6610 }
6611

```

## D.16.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
urlInfo	array: see schema	yes	Read Only	The valid range for the value Property
url (urlInfo)	string	yes		url to download the description
content-type (urlInfo)	string			content-type of the introspection data
version (urlInfo)	integer			version the introspection data that can be downloaded
protocol (urlInfo)	string	yes		protocol to be used to download the introspection

6613 **D.16.6 CRUDN behavior**

Resource	Create	Read	Update	Delete	Notify
/IntrospectionResURI		get			

6614

## Annex E (informative)

### Swagger2.0 definitions

#### E.1 Icon

##### E.1.1 Introduction

This resource describes the attributes associated with an Icon.  
Retrieves the current icon properties.

##### E.1.2 Example URI

/IconResURI

##### E.1.3 Resource Type

The resource type (rt) is defined as: ['oic.r.icon'].

##### E.1.4 Swagger2.0 Definition

```
{
  "swagger": "2.0",
  "info": {
    "title": "Icon",
    "version": "v1.1.0-20161107",
    "license": {
      "name": "copyright 2016-2017 Open Connectivity Foundation, Inc. All rights reserved.",
      "x-description": "Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:\n      1.
Redistributions of source code must retain the above copyright notice, this list of conditions and
the following disclaimer.\n      2. Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer in the documentation and/or
other materials provided with the distribution.\n\n      THIS SOFTWARE IS PROVIDED BY THE Open
Connectivity Foundation, INC. \\"AS IS\\" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE OR
WARRANTIES OF NON-INFRINGEMENT, ARE DISCLAIMED.\n      IN NO EVENT SHALL THE Open Connectivity
Foundation, INC. OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)\n      HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR
OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY
OF SUCH DAMAGE.\n"
    }
  },
  "schemes": ["http"],
  "consumes": ["application/json"],
  "produces": ["application/json"],
  "paths": {
    "/IconResURI" : {
      "get": {
        "description": "This resource describes the attributes associated with an Icon.\nRetrieves
the current icon properties.\n",
        "parameters": [
        ],
        "responses": {
          "200": {
            "description": "",
            "x-example": {
              "rt": ["oic.r.icon"],
              "id": "unique_example_id",
              "mimetype": "image/png",
              "width": 256,
              "height": 256,
            }
          }
        }
      }
    }
  }
}
```

```

6674         "media": "http://findbetter.ru/public/uploads/1481662800/2043.png"
6675     }
6676     ,
6677     "schema": { "$ref": "#/definitions/Icon" }
6678 }
6679 }
6680 }
6681 }
6682 },
6683 "parameters": {
6684     "interface" : {
6685         "in" : "query",
6686         "name" : "if",
6687         "type" : "string",
6688         "enum" : ["oic.if.r", "oic.if.baseline"]
6689     }
6690 },
6691 "definitions": {
6692     "Icon" : {
6693         "properties": {
6694             "mimetype" :
6695                 {
6696                     "description": "The Media Type of the icon",
6697                     "maxLength": 64,
6698                     "readOnly": true,
6699                     "type": "string"
6700                 },
6701             "rt" :
6702                 {
6703                     "description": "Resource Type of the Resource",
6704                     "items": {
6705                         "maxLength": 64,
6706                         "type": "string"
6707                     },
6708                     "minItems": 1,
6709                     "readOnly": true,
6710                     "type": "array"
6711                 },
6712             "media" :
6713                 {
6714                     "description": "Specifies the URI to the icon",
6715                     "format": "uri",
6716                     "maxLength": 256,
6717                     "readOnly": true,
6718                     "type": "string"
6719                 },
6720             "n" :
6721                 {
6722                     "description": "Friendly name of the resource",
6723                     "maxLength": 64,
6724                     "readOnly": true,
6725                     "type": "string"
6726                 },
6727             "width" :
6728                 {
6729                     "description": "The width in pixels",
6730                     "minimum": 1,
6731                     "readOnly": true,
6732                     "type": "integer"
6733                 },
6734             "height" :
6735                 {
6736                     "description": "The height in pixels",
6737                     "minimum": 1,
6738                     "readOnly": true,
6739                     "type": "integer"
6740                 }
6741         }
6742     }
6743 }
6744

```



```

6745     },
6746
6747     "id" :
6748     {
6749         "description": "Instance ID of this specific resource",
6750         "maxLength": 64,
6751         "readOnly": true,
6752         "type": "string"
6753     },
6754
6755     "if" :
6756     {
6757         "description": "The interface set supported by this resource",
6758         "items": {
6759             "enum": [
6760                 "oic.if.baseline",
6761                 "oic.if.ll",
6762                 "oic.if.b",
6763                 "oic.if.lb",
6764                 "oic.if.rw",
6765                 "oic.if.r",
6766                 "oic.if.a",
6767                 "oic.if.s"
6768             ],
6769             "type": "string"
6770         },
6771         "minItems": 1,
6772         "readOnly": true,
6773         "type": "array"
6774     }
6775 },
6776 "required": ["mimetype", "width", "height", "media"]
6777 }
6778 }
6779 }
6780

```

### E.1.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
width	integer	yes	Read Only	Specifies the width in pixels
rt	array: see schema		Read Only	Resource Type
id	string		Read Only	Instance ID of this specific resource
height	integer	yes	Read Only	Specifies the height in pixels
mimetype	string	yes	Read Only	Specifies the format of the MIME Type
n	string		Read Only	Friendly name of the resource
if	array: see schema		Read Only	The interface set supported by this resource
media	string	yes	Read Only	Specifies the media URL to icon

### E.1.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/IconResURI		get			

## E.2 Introspection Resource

### E.2.1 Introduction

This resource provides the means to get the device introspection data specifying all the endpoints of the device. The url hosted by this resource is either a local or an external url.

### E.2.2 Example URI

/IntrospectionResURI

### E.2.3 Resource Type

The resource type (rt) is defined as: ['oic.wk.introspection'].

### E.2.4 Swagger2.0 Definition

```
{
  "swagger": "2.0",
  "info": {
    "title": "Introspection Resource",
    "version": "v1.0.0-20160707",
    "license": {
      "name": "copyright 2016-2017 Open Connectivity Foundation, Inc. All rights reserved.",
      "x-description": "Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:\n      1.
Redistributions of source code must retain the above copyright notice, this list of conditions and
the following disclaimer.\n      2. Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer in the documentation and/or
other materials provided with the distribution.\n      THIS SOFTWARE IS PROVIDED BY THE Open
Connectivity Foundation, INC. \"AS IS\" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE OR
WARRANTIES OF NON-INFRINGEMENT, ARE DISCLAIMED.\n      IN NO EVENT SHALL THE Open Connectivity
Foundation, INC. OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)\n      HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR
OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY
OF SUCH DAMAGE.\n"
    }
  },
  "schemes": ["http"],
  "consumes": ["application/json"],
  "produces": ["application/json"],
  "paths": {
    "/IntrospectionResURI" : {
      "get": {
        "description": "This resource provides the means to get the device introspection data
specifiying all the endpoints of the device.\nThe url hosted by this resource is either a local or
an external url.\n",
        "parameters": [
        ],
        "responses": {
          "200": {
            "description": "",
            "x-example": {
              "rt" : ["oic.wk.introspection"],
              "urlInfo" : [
                {
                  "content-type" : "application/cbor",
                  "protocol" : "coap",
                  "url" : "coap://[fe80::1]:1234/IntrospectionExampleURI"
                }
              ]
            }
          }
        }
      },
      "schema": { "$ref": "#/definitions/oic.wk.introspectionInfo" }
    }
  }
}
```

```

6846     }
6847   }
6848 }
6849 },
6850 "parameters": {
6851   "interface" : {
6852     "in" : "query",
6853     "name" : "if",
6854     "type" : "string",
6855     "enum" : ["oic.if.r", "oic.if.baseline"]
6856   }
6857 },
6858 "definitions": {
6859   "oic.wk.introspectionInfo" : {
6860     "properties": {
6861       "rt" :
6862         {
6863           "description": "Resource Type of the Resource",
6864           "items": {
6865             "maxLength": 64,
6866             "type": "string"
6867           },
6868           "minItems": 1,
6869           "readOnly": true,
6870           "type": "array"
6871         },
6872       "n" :
6873         {
6874           "description": "Friendly name of the resource",
6875           "maxLength": 64,
6876           "readOnly": true,
6877           "type": "string"
6878         },
6879     },
6880     "urlInfo" :
6881       {
6882         "description": "Information on the location of the introspection data.",
6883         "items": {
6884           "properties": {
6885             "content-type": {
6886               "default": "application/cbor",
6887               "description": "content-type of the introspection data",
6888               "enum": [
6889                 "application/json",
6890                 "application/cbor"
6891               ],
6892               "type": "string"
6893             },
6894           },
6895           "protocol": {
6896             "description": "Identifier for the protocol to be used to obtain the introspection
6897 information",
6898             "enum": [
6899               "coap",
6900               "coaps",
6901               "http",
6902               "https",
6903               "coap+tcp",
6904               "coaps+tcp"
6905             ],
6906             "type": "string"
6907           },
6908           "url": {
6909             "description": "The URL of the introspection information.",
6910             "format": "uri",
6911             "type": "string"
6912           },
6913           "version": {
6914             "default": 1,
6915             "description": "The version of the introspection data that can be downloaded",
6916             "enum": [

```

```

6917         1
6918     ],
6919     "type": "integer"
6920 },
6921 },
6922     "required": [
6923         "url",
6924         "protocol"
6925     ],
6926     "type": "object"
6927 },
6928     "minItems": 1,
6929     "readOnly": true,
6930     "type": "array"
6931 },
6932
6933     "id" :
6934     {
6935         "description": "Instance ID of this specific resource",
6936         "maxLength": 64,
6937         "readOnly": true,
6938         "type": "string"
6939     },
6940
6941     "if" :
6942     {
6943         "description": "The interface set supported by this resource",
6944         "items": {
6945             "enum": [
6946                 "oic.if.baseline",
6947                 "oic.if.ll",
6948                 "oic.if.b",
6949                 "oic.if.lb",
6950                 "oic.if.rw",
6951                 "oic.if.r",
6952                 "oic.if.a",
6953                 "oic.if.s"
6954             ],
6955             "type": "string"
6956         },
6957         "minItems": 1,
6958         "readOnly": true,
6959         "type": "array"
6960     }
6961 }
6962 }
6963 }
6964 }
6965 }

```

## E.2.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
if	array: see schema		Read Only	The interface set supported by this resource
id	string		Read Only	Instance ID of this specific resource
urlInfo	array: see schema	yes	Read Only	The valid range for the value Property
rt	array: see schema		Read Only	Resource Type
n	string		Read Only	Friendly name of the resource

## 6967 E.2.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/IntrospectionResURI		get			

## 6968 E.3 OCF Collection

### 6969 E.3.1 Introduction

6970 OCF Collection Resource Type contains properties and links.  
 6971 The oic.if.baseline interface exposes a representation of  
 6972 the links and the properties of the collection resource itself  
 6973 Retrieve on Baseline Interface  
 6974

### 6975 E.3.2 Example URI

6976 /CollectionBaselineInterfaceURI

### 6977 E.3.3 Resource Type

6978 The resource type (rt) is defined as: ['oic.wk.col'].

### 6979 E.3.4 Swagger2.0 Definition

```

6980 {
6981   "swagger": "2.0",
6982   "info": {
6983     "title": "OCF Collection",
6984     "version": "1.0",
6985     "license": {
6986       "name": "copyright 2016-2017 Open Connectivity Foundation, Inc. All rights reserved.",
6987       "x-description": "Redistribution and use in source and binary forms, with or without
6988 modification, are permitted provided that the following conditions are met:\n      1.
6989 Redistributions of source code must retain the above copyright notice, this list of conditions and
6990 the following disclaimer.\n      2. Redistributions in binary form must reproduce the above
6991 copyright notice, this list of conditions and the following disclaimer in the documentation and/or
6992 other materials provided with the distribution.\n\n      THIS SOFTWARE IS PROVIDED BY THE Open
6993 Connectivity Foundation, INC. \ "AS IS\ " AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
6994 LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE OR
6995 WARRANTIES OF NON-INFRINGEMENT, ARE DISCLAIMED.\n      IN NO EVENT SHALL THE Open Connectivity
6996 Foundation, INC. OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
6997 EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
6998 OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)\n      HOWEVER CAUSED AND
6999 ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR
7000 OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY
7001 OF SUCH DAMAGE.\n"
7002   },
7003 },
7004 "schemes": ["http"],
7005 "consumes": ["application/json"],
7006 "produces": ["application/json"],
7007 "paths": {
7008   "/CollectionBaselineInterfaceURI" : {
7009     "get": {
7010       "description": "OCF Collection Resource Type contains properties and links.\nThe
7011 oic.if.baseline interface exposes a representation of\nthe links and the properties of the
7012 collection resource itself\nRetrieve on Baseline Interface\n",
7013       "parameters": [
7014       ],
7015       "responses": {
7016         "200": {
7017           "description" : "",
7018           "x-example":
7019             {
7020               "rt": ["oic.wk.col"],
7021               "id": "unique_example_id",
7022               "rts": [ "oic.r.switch.binary", "oic.r.airflow" ],
7023               "links": [
7024                 {

```

```

7025         "href": "switch",
7026         "rt": ["oic.r.switch.binary"],
7027         "if": ["oic.if.a", "oic.if.baseline"],
7028         "eps": [
7029             { "ep": "coap://[fe80::b1d6]:1111", "pri": 2 },
7030             { "ep": "coaps://[fe80::b1d6]:1122", "pri": 3 },
7031             { "ep": "coap+tcp://[2001:db8:a::123]:2222", "pri": 3 }
7032         ],
7033     },
7034     {
7035         "href": "airFlow",
7036         "rt": ["oic.r.airflow"],
7037         "if": ["oic.if.a", "oic.if.baseline"],
7038         "eps": [
7039             { "ep": "coap://[fe80::b1d6]:1111", "pri": 2 },
7040             { "ep": "coaps://[fe80::b1d6]:1122", "pri": 3 },
7041             { "ep": "coap+tcp://[2001:db8:a::123]:2222", "pri": 3 }
7042         ]
7043     }
7044 ]
7045 },
7046 {
7047     "schema": { "$ref": "#/definitions/sbaseline" }
7048 },
7049 },
7050 },
7051 "post": {
7052     "description": "Update on Baseline Interface\n",
7053     "parameters": [
7054         {
7055             "name": "body",
7056             "in": "body",
7057             "required": true,
7058             "schema": { "$ref": "#/definitions/sbaseline" }
7059         }
7060     ],
7061     "responses": {
7062         "200": {
7063             "description": "",
7064             "schema": { "$ref": "#/definitions/sbaseline" }
7065         }
7066     }
7067 },
7068 },
7069 "/CollectionBatchInterfaceURI" : {
7070     "get": {
7071         "description": "OCF Collection Resource Type contains properties and links.\nThe oic.if.b
7072 interface exposes a composite representation of the\nresources pointed to by the links\nRetrieve
7073 on Batch Interface\n",
7074         "parameters": [
7075         ],
7076         "responses": {
7077             "200": {
7078                 "description": "All targets returned OK status (HTTP 200 or CoAP 2.05 Content)",
7079                 "x-example":
7080                 [
7081                     {
7082                         "href": "switch",
7083                         "rep":
7084                         {
7085                             "value": true
7086                         }
7087                     },
7088                     {
7089                         "href": "airFlow",
7090                         "rep":
7091                         {
7092                             "direction": "floor",
7093                             "speed": 3
7094                         }
7095                     }
7096                 ]
7097             }
7098         }
7099     }
7100 }

```

```

7096         ]
7097     ,
7098     "schema": { "$ref": "#/definitions/sbatch-retrieve" }
7099 },
7100 "404": {
7101     "description" : "One or more targets did not return an OK status, return a
7102 representation containing returned properties from the targets that returned OK",
7103     "x-example":
7104     [
7105         {
7106             "href": "switch",
7107             "rep":
7108             {
7109                 "value": true
7110             }
7111         }
7112     ]
7113     ,
7114     "schema": { "$ref": "#/definitions/sbatch-retrieve" }
7115 }
7116 },
7117 },
7118 "post": {
7119     "description": "Update on Batch Interface\n",
7120     "parameters": [
7121         {
7122             "name": "body",
7123             "in": "body",
7124             "required": true,
7125             "schema": { "$ref": "#/definitions/sbatch-update" },
7126             "x-example":
7127             [
7128                 {
7129                     "href": "switch",
7130                     "rep":
7131                     {
7132                         "value": true
7133                     }
7134                 },
7135                 {
7136                     "href": "airFlow",
7137                     "rep":
7138                     {
7139                         "direction": "floor",
7140                         "speed": 3
7141                     }
7142                 }
7143             ]
7144         }
7145     ],
7146     "responses": {
7147         "200": {
7148             "description" : "all targets returned OK status (HTTP 200 or CoAP 2.04 Changed)
7149 return a representation of the current state of all targets",
7150             "x-example":
7151             [
7152                 {
7153                     "href": "switch",
7154                     "rep":
7155                     {
7156                         "value": true
7157                     }
7158                 },
7159                 {
7160                     "href": "airFlow",
7161                     "rep":
7162                     {
7163                         "direction": "demist",
7164                         "speed": 5
7165                     }
7166                 }

```

```

7167         ],
7168         ,
7169         "schema": { "$ref": "#/definitions/sbatch-retrieve" }
7170     },
7171     "403": {
7172         "description": "one or more targets did not return OK status; return a retrieve
7173 representation of the current state of all targets in the batch",
7174         "x-example":
7175         [
7176             {
7177                 "href": "switch",
7178                 "rep":
7179                 {
7180                     "value": true
7181                 }
7182             },
7183             {
7184                 "href": "airFlow",
7185                 "rep":
7186                 {
7187                     "direction": "floor",
7188                     "speed": 3
7189                 }
7190             }
7191         ]
7192     },
7193     "schema": { "$ref": "#/definitions/sbatch-retrieve" }
7194 }
7195 }
7196 }
7197 },
7198 "/CollectionLinkListInterfaceURI" : {
7199     "get": {
7200         "description": "OCF Collection Resource Type contains properties and links.\nThe oic.if.ll
7201 interface exposes a representation of the links\nRetrieve on Link List Interface\n",
7202         "parameters": [
7203         ],
7204         "responses": {
7205             "200": {
7206                 "description": "",
7207                 "x-example":
7208                 [
7209                     {
7210                         "href": "switch",
7211                         "rt": ["oic.r.switch.binary"],
7212                         "if": ["oic.if.a", "oic.if.baseline"],
7213                         "eps": [
7214                             { "ep": "coap://[fe80::b1d6]:1111", "pri": 2 },
7215                             { "ep": "coaps://[fe80::b1d6]:1122" },
7216                             { "ep": "coap+tcp://[2001:db8:a::123]:2222", "pri": 3 }
7217                         ]
7218                     },
7219                     {
7220                         "href": "airFlow",
7221                         "rt": ["oic.r.airflow"],
7222                         "if": ["oic.if.a", "oic.if.baseline"],
7223                         "eps": [
7224                             { "ep": "coap://[fe80::b1d6]:1111", "pri": 2 },
7225                             { "ep": "coaps://[fe80::b1d6]:1122" },
7226                             { "ep": "coap+tcp://[2001:db8:a::123]:2222", "pri": 3 }
7227                         ]
7228                     }
7229                 ]
7230             },
7231             "schema": { "$ref": "#/definitions/slinks" }
7232         }
7233     }
7234 }
7235 },
7236 },
7237 "parameters": {

```



```

7238     "interface-ll" : {
7239         "in" : "query",
7240         "name" : "if",
7241         "type" : "string",
7242         "enum" : ["oic.if.ll"]
7243     },
7244     "interface-b" : {
7245         "in" : "query",
7246         "name" : "if",
7247         "type" : "string",
7248         "enum" : ["oic.if.b"]
7249     },
7250     "interface-baseline" : {
7251         "in" : "query",
7252         "name" : "if",
7253         "type" : "string",
7254         "enum" : ["oic.if.baseline"]
7255     },
7256     "interface-all" : {
7257         "in" : "query",
7258         "name" : "if",
7259         "type" : "string",
7260         "enum" : ["oic.if.ll", "oic.if.baseline", "oic.if.b"]
7261     }
7262 },
7263 "definitions": {
7264     "sbaseline" : {
7265         "properties": {
7266             "links" :
7267             {
7268                 "description": "A set of simple or individual OIC Links.",
7269                 "items": {
7270                     "$ref": "#/definitions/oic.oic-link"
7271                 },
7272                 "type": "array"
7273             }
7274         }
7275     }
7276 },
7277 ,
7278 "sbatch-retrieve" : {
7279     "title" :
7280         "Collection Batch Retrieve Format (auto merged)"
7281     ,
7282     "minItems" :
7283         1
7284     ,
7285     "items" :
7286     {
7287         "additionalProperties": true,
7288         "properties": {
7289             "href": {
7290                 "description": "URI of the target resource relative assuming the collection URI as
7291 anchor",
7292                 "format": "uri",
7293                 "maxLength": 256,
7294                 "type": "string"
7295             },
7296             "rep": {
7297                 "oneOf": [
7298                     {
7299                         "description": "The response payload from a single resource",
7300                         "type": "object"
7301                     },
7302                     {
7303                         "description": " The response payload from a collection (batch) resource",
7304                         "items": {
7305                             "properties": {
7306                                 "anchor": {
7307                                     "description": "This is used to override the context URI e.g. override the
7308 URI of the containing collection.",

```

```

7309         "format": "uri",
7310         "maxLength": 256,
7311         "type": "string"
7312     },
7313     "di": {
7314         "allOf": [
7315             {
7316                 "description": "Format pattern according to IETF RFC 4122.",
7317                 "pattern": "^[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-
7318 [a-fA-F0-9]{12}$",
7319                 "type": "string"
7320             },
7321             {
7322                 "description": "The device ID"
7323             }
7324         ]
7325     },
7326     "eps": {
7327         "description": "the Endpoint information of the target Resource",
7328         "items": {
7329             "properties": {
7330                 "ep": {
7331                     "description": "Transport Protocol Suite + Endpoint Locator",
7332                     "format": "uri",
7333                     "type": "string"
7334                 },
7335                 "pri": {
7336                     "description": "The priority among multiple Endpoints",
7337                     "minimum": 1,
7338                     "type": "integer"
7339                 }
7340             },
7341             "type": "object"
7342         },
7343         "type": "array"
7344     },
7345     "href": {
7346         "description": "This is the target URI, it can be specified as a Relative
7347 Reference or fully-qualified URI.",
7348         "format": "uri",
7349         "maxLength": 256,
7350         "type": "string"
7351     },
7352     "if": {
7353         "description": "The interface set supported by this resource",
7354         "items": {
7355             "enum": [
7356                 "oic.if.baseline",
7357                 "oic.if.ll",
7358                 "oic.if.b",
7359                 "oic.if.rw",
7360                 "oic.if.r",
7361                 "oic.if.a",
7362                 "oic.if.s"
7363             ],
7364             "type": "string"
7365         },
7366         "minItems": 1,
7367         "type": "array"
7368     },
7369     "ins": {
7370         "description": "The instance identifier for this web link in an array of web
7371 links - used in collections",
7372         "type": "integer"
7373     },
7374     "p": {
7375         "description": "Specifies the framework policies on the Resource referenced
7376 by the target URI",
7377         "properties": {
7378             "bm": {

```

```

7379         "description": "Specifies the framework policies on the Resource
7380 referenced by the target URI for e.g. observable and discoverable",
7381         "type": "integer"
7382     },
7383 },
7384     "required": [
7385         "bm"
7386     ],
7387     "type": "object"
7388 },
7389     "rel": {
7390         "description": "The relation of the target URI referenced by the link to the
7391 context URI",
7392         "oneOf": [
7393             {
7394                 "default": [
7395                     "hosts"
7396                 ],
7397                 "items": {
7398                     "maxLength": 64,
7399                     "type": "string"
7400                 },
7401                 "minItems": 1,
7402                 "type": "array"
7403             },
7404             {
7405                 "default": "hosts",
7406                 "maxLength": 64,
7407                 "type": "string"
7408             }
7409         ],
7410     },
7411     "rt": {
7412         "description": "Resource Type of the Resource",
7413         "items": {
7414             "maxLength": 64,
7415             "type": "string"
7416         },
7417         "minItems": 1,
7418         "type": "array"
7419     },
7420     "title": {
7421         "description": "A title for the link relation. Can be used by the UI to
7422 provide a context.",
7423         "maxLength": 64,
7424         "type": "string"
7425     },
7426     "type": {
7427         "default": "application/cbor",
7428         "description": "A hint at the representation of the resource referenced by
7429 the target URI. This represents the media types that are used for both accepting and emitting.",
7430         "items": {
7431             "maxLength": 64,
7432             "type": "string"
7433         },
7434         "minItems": 1,
7435         "type": "array"
7436     }
7437 },
7438     "required": [
7439         "href",
7440         "rt",
7441         "if"
7442     ],
7443     "type": "object"
7444 },
7445     "type": "array"
7446 }
7447 ]
7448 }
7449 },

```

```

7450         "required": [
7451             "href",
7452             "rep"
7453         ],
7454         "type": "object"
7455     }
7456
7457     , "type" :
7458         "array"
7459
7460 }
7461
7462 "sbatch-update" : {
7463     "title" :
7464         "Collection Batch Update Format (auto merged)"
7465
7466     , "minItems" :
7467         1
7468
7469     , "items" :
7470         {
7471             "$ref": "#/definitions/oic.batch-update.item"
7472         }
7473
7474     , "type" :
7475         "array"
7476
7477 }
7478
7479 "slinks" : {
7480     "type" :
7481         "array"
7482
7483     , "items" :
7484         {
7485             "$ref": "#/definitions/oic.oic-link"
7486         }
7487
7488 }
7489
7490 "oic.wk.col-batch-update" :
7491 {
7492     "description": "array of resource representations to apply to the batch collection, using
7493 href to indicate which resource(s) in the batch to update. If the href property is empty,
7494 effectively making the URI reference to the collection itself, the representation is to be applied
7495 to all resources in the batch",
7496     "items": {
7497         "$ref": "#/definitions/oic.batch-update.item"
7498     },
7499     "minItems": 1,
7500     "type": "array"
7501 }
7502
7503 , "uuid" :
7504 {
7505     "description": "Format pattern according to IETF RFC 4122.",
7506     "pattern": "^[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{12}$",
7507     "type": "string"
7508 }
7509
7510 "oic.collection.properties" :
7511 {
7512     "description": "A collection is a set of links along with additional properties to describe
7513 the collection itself",
7514     "properties": {
7515         "rts": {
7516             "$ref": "#/definitions/oic.core/properties/rt",
7517             "description": "The list of allowable resource types (for Target and anchors) in links
7518 included in the collection"
7519         }
7520     },
7521     "type": "object"

```

```

7521     }
7522
7523     , "oic.core" :
7524     {
7525         "properties": {
7526             "rt": {
7527                 "description": "Resource Type of the Resource",
7528                 "items": {
7529                     "maxLength": 64,
7530                     "type": "string"
7531                 },
7532                 "minItems": 1,
7533                 "readOnly": true,
7534                 "type": "array"
7535             }
7536         },
7537         "type": "object"
7538     }
7539
7540     , "oic.batch-update.item" :
7541     {
7542         "additionalProperties": true,
7543         "description": "array of resource representations to apply to the batch collection, using
7544 href to indicate which resource(s) in the batch to update. If the href property is empty,
7545 effectively making the URI reference to the collection itself, the representation is to be applied
7546 to all resources in the batch",
7547         "properties": {
7548             "href": {
7549                 "description": "URI of the target resource relative assuming the collection URI as
7550 anchor",
7551                 "format": "uri",
7552                 "maxLength": 256,
7553                 "type": "string"
7554             },
7555             "rep": {
7556                 "oneOf": [
7557                     {
7558                         "description": "The response payload from a single resource",
7559                         "type": "object"
7560                     },
7561                     {
7562                         "description": " The response payload from a collection (batch) resource",
7563                         "items": {
7564                             "$ref": "#/definitions/oic.oic-link"
7565                         },
7566                         "type": "array"
7567                     }
7568                 ]
7569             }
7570         },
7571         "required": [
7572             "href",
7573             "rep"
7574         ],
7575         "type": "object"
7576     }
7577
7578     , "oic.collection.linksexpanded" :
7579     {
7580         "properties": {
7581             "links": {
7582                 "description": "A set of simple or individual OIC Links.",
7583                 "items": {
7584                     "properties": {
7585                         "anchor": {
7586                             "description": "This is used to override the context URI e.g. override the URI of
7587 the containing collection.",
7588                             "format": "uri",
7589                             "maxLength": 256,
7590                             "type": "string"
7591                         },

```

```

7592     "di": {
7593         "description": "Format pattern according to IETF RFC 4122.",
7594         "pattern": "^[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{12}$",
7595     },
7596     "type": "string"
7597 },
7598 "eps": {
7599     "description": "the Endpoint information of the target Resource",
7600     "items": {
7601         "properties": {
7602             "ep": {
7603                 "description": "Transport Protocol Suite + Endpoint Locator",
7604                 "format": "uri",
7605                 "type": "string"
7606             },
7607             "pri": {
7608                 "description": "The priority among multiple Endpoints",
7609                 "minimum": 1,
7610                 "type": "integer"
7611             }
7612         },
7613         "type": "object"
7614     },
7615     "type": "array"
7616 },
7617 "href": {
7618     "description": "This is the target URI, it can be specified as a Relative Reference
7619 or fully-qualified URI.",
7620     "format": "uri",
7621     "maxLength": 256,
7622     "type": "string"
7623 },
7624 "if": {
7625     "description": "The interface set supported by this resource",
7626     "items": {
7627         "enum": [
7628             "oic.if.baseline",
7629             "oic.if.ll",
7630             "oic.if.b",
7631             "oic.if.rw",
7632             "oic.if.r",
7633             "oic.if.a",
7634             "oic.if.s"
7635         ],
7636         "type": "string"
7637     },
7638     "minItems": 1,
7639     "type": "array"
7640 },
7641 "ins": {
7642     "description": "The instance identifier for this web link in an array of web links
7643 - used in collections",
7644     "type": "integer"
7645 },
7646 "p": {
7647     "description": "Specifies the framework policies on the Resource referenced by the
7648 target URI",
7649     "properties": {
7650         "bm": {
7651             "description": "Specifies the framework policies on the Resource referenced by
7652 the target URI for e.g. observable and discoverable",
7653             "type": "integer"
7654         }
7655     },
7656     "required": [
7657         "bm"
7658     ],
7659     "type": "object"
7660 },
7661 "rel": {

```

```

7662         "description": "The relation of the target URI referenced by the link to the
7663 context URI",
7664         "oneOf": [
7665             {
7666                 "default": [
7667                     "hosts"
7668                 ],
7669                 "items": {
7670                     "maxLength": 64,
7671                     "type": "string"
7672                 },
7673                 "minItems": 1,
7674                 "type": "array"
7675             },
7676             {
7677                 "default": "hosts",
7678                 "maxLength": 64,
7679                 "type": "string"
7680             }
7681         ],
7682     },
7683     "rt": {
7684         "description": "Resource Type of the Resource",
7685         "items": {
7686             "maxLength": 64,
7687             "type": "string"
7688         },
7689         "minItems": 1,
7690         "type": "array"
7691     },
7692     "title": {
7693         "description": "A title for the link relation. Can be used by the UI to provide a
7694 context.",
7695         "maxLength": 64,
7696         "type": "string"
7697     },
7698     "type": {
7699         "default": "application/cbor",
7700         "description": "A hint at the representation of the resource referenced by the
7701 target URI. This represents the media types that are used for both accepting and emitting.",
7702         "items": {
7703             "maxLength": 64,
7704             "type": "string"
7705         },
7706         "minItems": 1,
7707         "type": "array"
7708     },
7709 },
7710 "required": [
7711     "href",
7712     "rt",
7713     "if"
7714 ],
7715 "type": "object"
7716 },
7717 "type": "array"
7718 }
7719 },
7720 "type": "object"
7721 }
7722 , "oic.collection.links" :
7723 {
7724     "properties": {
7725         "links": {
7726             "description": "A set of simple or individual OIC Links.",
7727             "items": {
7728                 "$ref": "#/definitions/oic.oic-link"
7729             },
7730         },
7731         "type": "array"
7732     }

```

```

7733     },
7734     "type": "object"
7735 }
7736
7737 , "oic.oic-link" :
7738 {
7739     "properties": {
7740         "anchor": {
7741             "description": "This is used to override the context URI e.g. override the URI of the
7742 containing collection.",
7743             "format": "uri",
7744             "maxLength": 256,
7745             "type": "string"
7746         },
7747         "di": {
7748             "$ref": "#/definitions/uuid",
7749             "description": "The device ID"
7750         },
7751         "eps": {
7752             "description": "the Endpoint information of the target Resource",
7753             "items": {
7754                 "properties": {
7755                     "ep": {
7756                         "description": "Transport Protocol Suite + Endpoint Locator",
7757                         "format": "uri",
7758                         "type": "string"
7759                     },
7760                     "pri": {
7761                         "description": "The priority among multiple Endpoints",
7762                         "minimum": 1,
7763                         "type": "integer"
7764                     }
7765                 },
7766                 "type": "object"
7767             },
7768             "type": "array"
7769         },
7770         "href": {
7771             "description": "This is the target URI, it can be specified as a Relative Reference or
7772 fully-qualified URI.",
7773             "format": "uri",
7774             "maxLength": 256,
7775             "type": "string"
7776         },
7777         "if": {
7778             "description": "The interface set supported by this resource",
7779             "items": {
7780                 "enum": [
7781                     "oic.if.baseline",
7782                     "oic.if.ll",
7783                     "oic.if.b",
7784                     "oic.if.rw",
7785                     "oic.if.r",
7786                     "oic.if.a",
7787                     "oic.if.s"
7788                 ],
7789                 "type": "string"
7790             },
7791             "minItems": 1,
7792             "type": "array"
7793         },
7794         "ins": {
7795             "description": "The instance identifier for this web link in an array of web links - used
7796 in collections",
7797             "type": "integer"
7798         },
7799         "p": {
7800             "description": "Specifies the framework policies on the Resource referenced by the target
7801 URI",
7802             "properties": {
7803                 "bm": {

```



```

7804         "description": "Specifies the framework policies on the Resource referenced by the
7805 target URI for e.g. observable and discoverable",
7806         "type": "integer"
7807     },
7808 },
7809 "required": [
7810     "bm"
7811 ],
7812 "type": "object"
7813 },
7814 "rel": {
7815     "description": "The relation of the target URI referenced by the link to the context
7816 URI",
7817     "oneOf": [
7818         {
7819             "default": [
7820                 "hosts"
7821             ],
7822             "items": {
7823                 "maxLength": 64,
7824                 "type": "string"
7825             },
7826             "minItems": 1,
7827             "type": "array"
7828         },
7829         {
7830             "default": "hosts",
7831             "maxLength": 64,
7832             "type": "string"
7833         }
7834     ],
7835 },
7836 "rt": {
7837     "description": "Resource Type of the Resource",
7838     "items": {
7839         "maxLength": 64,
7840         "type": "string"
7841     },
7842     "minItems": 1,
7843     "type": "array"
7844 },
7845 "title": {
7846     "description": "A title for the link relation. Can be used by the UI to provide a
7847 context.",
7848     "maxLength": 64,
7849     "type": "string"
7850 },
7851 "type": {
7852     "default": "application/cbor",
7853     "description": "A hint at the representation of the resource referenced by the target
7854 URI. This represents the media types that are used for both accepting and emitting.",
7855     "items": {
7856         "maxLength": 64,
7857         "type": "string"
7858     },
7859     "minItems": 1,
7860     "type": "array"
7861 },
7862 },
7863 "required": [
7864     "href",
7865     "rt",
7866     "if"
7867 ],
7868 "type": "object"
7869 }
7870 }
7871 }
7872 }

```

**E.3.5 Property Definition**

Property name	Value type	Mandatory	Access mode	Description
links	multiple types: see schema			All forms of links in a collection
anchor	string			This is used to override the context URI e.g. override the URI of the containing collection
rel	multiple types: see schema			The relation of the target URI referenced by the link to the context URI
type	array: see schema			A hint at the representation of the resource referenced by the target URI. This represents the media types that are used for both accepting and emitting
if	array: see schema	yes		The interface set supported by this resource
href	string	yes		This is the target URI, it can be specified as a Relative Reference or fully-qualified URI. Relative Reference should be used along with the di parameter to make it unique.
rt	array: see schema	yes		Resource Type
p	object: see schema			Specifies the framework policies on the Resource referenced by the target URI
ins	multiple types: see schema			The instance identifier for this web link in an array of web links - used in collections

n	string		Read Only	Friendly name of the resource
rts	array: see schema		Read Only	Defines the list of allowable resource types (for Target and anchors) in links included in the collection; new links being created can only be from this list
id	string		Read Only	Instance ID of this specific resource
di	string			Unique identifier for device (UUID)
drel	string			When specified this is the default relationship to use when an OIC Link does not specify an explicit relationship with *rel* parameter
eps	array: see schema			the Endpoint information of the target Resource
title	string			A title for the link relation. Can be used by the UI to provide a context
rep	multiple types: see schema	yes		
href	string	yes		URI of the target resource relative assuming the collection URI as anchor
title	string			A title for the link relation. Can be used by the UI to provide a context
anchor	string			This is used to override the context URI e.g. override the URI of the containing collection
rel	multiple types: see schema			The relation of the target URI referenced by

				the link to the context URI
type	array: see schema			A hint at the representation of the resource referenced by the target URI. This represents the media types that are used for both accepting and emitting
di	string			Unique identifier for device (UUID)
href	string	yes		This is the target URI, it can be specified as a Relative Reference or fully-qualified URI. Relative Reference should be used along with the di parameter to make it unique.
rt	array: see schema	yes		Resource Type
p	object: see schema			Specifies the framework policies on the Resource referenced by the target URI
ins	multiple types: see schema			The instance identifier for this web link in an array of web links - used in collections
eps	array: see schema			the Endpoint information of the target Resource
if	array: see schema	yes		The interface set supported by this resource
rep	multiple types: see schema	yes		
href	string	yes		URI of the target resource relative assuming the collection URI as anchor

### 7874 E.3.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/CollectionBaselineInterfaceURI		get	post		

## 7875 E.4 Platform Configuration

### 7876 E.4.1 Introduction

7877 Resource that allows for platform specific information to be configured.  
 7878 Retrieves the current platform configuration settings  
 7879

### 7880 E.4.2 Example URI

7881 /example/PlatformConfigurationResURI

### 7882 E.4.3 Resource Type

7883 The resource type (rt) is defined as: ['oic.wk.con.p'].

### 7884 E.4.4 Swagger2.0 Definition

```

7885 {
7886   "swagger": "2.0",
7887   "info": {
7888     "title": "Platform Configuration",
7889     "version": "v1-20160622",
7890     "license": {
7891       "name": "copyright 2016-2017 Open Connectivity Foundation, Inc. All rights reserved.",
7892       "x-description": "Redistribution and use in source and binary forms, with or without
7893 modification, are permitted provided that the following conditions are met:\n      1.
7894 Redistributions of source code must retain the above copyright notice, this list of conditions and
7895 the following disclaimer.\n      2. Redistributions in binary form must reproduce the above
7896 copyright notice, this list of conditions and the following disclaimer in the documentation and/or
7897 other materials provided with the distribution.\n\n      THIS SOFTWARE IS PROVIDED BY THE Open
7898 Connectivity Foundation, INC. \AS IS\ AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
7899 LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE OR
7900 WARRANTIES OF NON-INFRINGEMENT, ARE DISCLAIMED.\n      IN NO EVENT SHALL THE Open Connectivity
7901 Foundation, INC. OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
7902 EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
7903 OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)\n      HOWEVER CAUSED AND
7904 ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR
7905 OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY
7906 OF SUCH DAMAGE.\n"
7907     },
7908   },
7909   "schemes": ["http"],
7910   "consumes": ["application/json"],
7911   "produces": ["application/json"],
7912   "paths": {
7913     "/examplePlatformConfigurationResURI" : {
7914       "get": {
7915         "description": "Resource that allows for platform specific information to be
7916 configured.\nRetrieves the current platform configuration settings\n",
7917         "parameters": [
7918           { "$ref": "#/parameters/interface-all" }
7919         ],
7920         "responses": {
7921           "200": {
7922             "description": "",
7923             "x-example": {
7924               {
7925                 "rt": ["oic.wk.con.p"],
7926                 "mnpn": [ { "language": "en", "value": "My Friendly Device Name" } ]
7927               }
7928             },
7929             "schema": { "$ref": "#/definitions/Conf_Platform" }
7930           }
7931         }
7932       }
7933     }
  
```

```

7933     "post": {
7934         "description": "Update the information about the platform\n",
7935         "parameters": [
7936             { "$ref": "#/parameters/interface-rw" },
7937             {
7938                 "name": "body",
7939                 "in": "body",
7940                 "required": true,
7941                 "schema": { "$ref": "#/definitions/Update_Platform" },
7942                 "x-example":
7943                     {
7944                         "n": "Nuevo nombre",
7945                         "mnpn": [ { "language": "es", "value": "Nuevo nombre de Plataforma Amigable" } ]
7946                     }
7947             },
7948         ],
7949         "responses": {
7950             "200": {
7951                 "description": "",
7952                 "x-example":
7953                     {
7954                         "n": "Nuevo nombre",
7955                         "mnpn": [ { "language": "es", "value": "Nuevo nombre de Plataforma Amigable" } ]
7956                     },
7957                 "schema": { "$ref": "#/definitions/Update_Platform" }
7958             }
7959         }
7960     },
7961 },
7962 },
7963 },
7964 "parameters": {
7965     "interface-rw" : {
7966         "in" : "query",
7967         "name" : "if",
7968         "type" : "string",
7969         "enum" : ["oic.if.rw"]
7970     },
7971     "interface-all" : {
7972         "in" : "query",
7973         "name" : "if",
7974         "type" : "string",
7975         "enum" : ["oic.if.rw", "oic.if.baseline"]
7976     }
7977 },
7978 "definitions": {
7979     "Conf_Platform" : {
7980         "properties": {
7981             "rt" :
7982                 {
7983                     "description": "Resource Type of the Resource",
7984                     "items": {
7985                         "maxLength": 64,
7986                         "type": "string"
7987                     },
7988                     "minItems": 1,
7989                     "readOnly": true,
7990                     "type": "array"
7991                 },
7992             "n" :
7993                 {
7994                     "description": "Friendly name of the resource",
7995                     "maxLength": 64,
7996                     "readOnly": true,
7997                     "type": "string"
7998                 },
7999             "mnpn" :
8000                 {
8001                     "description": "Platform names",

```

```

8004     "items": {
8005         "properties": {
8006             "language": {
8007                 "allOf": [
8008                     {
8009                         "description": "Format pattern according to IETF RFC 5646 (language tag).",
8010                         "pattern": "^[A-Za-z]{1,8}(-[A-Za-z0-9]{1,8})*$",
8011                         "type": "string"
8012                     },
8013                     {
8014                         "description": "An RFC 5646 language tag."
8015                     }
8016                 ]
8017             },
8018             "value": {
8019                 "description": "The Platform description in the indicated language.",
8020                 "maxLength": 64,
8021                 "type": "string"
8022             }
8023         },
8024         "type": "object"
8025     },
8026     "minItems": 1,
8027     "type": "array"
8028 },
8029
8030 "id" :
8031 {
8032     "description": "Instance ID of this specific resource",
8033     "maxLength": 64,
8034     "readOnly": true,
8035     "type": "string"
8036 },
8037
8038 "if" :
8039 {
8040     "description": "The interface set supported by this resource",
8041     "items": {
8042         "enum": [
8043             "oic.if.baseline",
8044             "oic.if.ll",
8045             "oic.if.b",
8046             "oic.if.lb",
8047             "oic.if.rw",
8048             "oic.if.r",
8049             "oic.if.a",
8050             "oic.if.s"
8051         ],
8052         "type": "string"
8053     },
8054     "minItems": 1,
8055     "readOnly": true,
8056     "type": "array"
8057 }
8058
8059 }
8060
8061 ,
8062 "Update_Platform" : {
8063     "properties": {
8064         "rt" :
8065         {
8066             "description": "Resource Type of the Resource",
8067             "items": {
8068                 "maxLength": 64,
8069                 "type": "string"
8070             },
8071             "minItems": 1,
8072             "readOnly": true,
8073             "type": "array"
8074         },

```

```

8075
8076     "n" :
8077         {
8078             "description": "Friendly name of the resource",
8079             "maxLength": 64,
8080             "type": "string"
8081         },
8082
8083     "mnpn" :
8084         {
8085             "description": "Platform names",
8086             "items": {
8087                 "properties": {
8088                     "language": {
8089                         "allOf": [
8090                             {
8091                                 "description": "Format pattern according to IETF RFC 5646 (language tag).",
8092                                 "pattern": "^[A-Za-z]{1,8}(-[A-Za-z0-9]{1,8})*$",
8093                                 "type": "string"
8094                             },
8095                             {
8096                                 "description": "An RFC 5646 language tag."
8097                             }
8098                         ]
8099                     },
8100                     "value": {
8101                         "description": "The Platform description in the indicated language.",
8102                         "maxLength": 64,
8103                         "type": "string"
8104                     }
8105                 },
8106                 "type": "object"
8107             },
8108             "minItems": 1,
8109             "type": "array"
8110         },
8111
8112     "id" :
8113         {
8114             "anyOf": [
8115                 {
8116                     "maxLength": 64,
8117                     "type": "string"
8118                 },
8119                 {
8120                     "description": "Format pattern according to IETF RFC 4122.",
8121                     "pattern": "^[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{12}$",
8122                     "type": "string"
8123                 }
8124             ],
8125             "description": "Instance ID of this specific resource",
8126             "readOnly": true
8127         },
8128
8129     "if" :
8130         {
8131             "description": "The interface set supported by this resource",
8132             "items": {
8133                 "enum": [
8134                     "oic.if.baseline",
8135                     "oic.if.ll",
8136                     "oic.if.b",
8137                     "oic.if.lb",
8138                     "oic.if.rw",
8139                     "oic.if.r",
8140                     "oic.if.a",
8141                     "oic.if.s"
8142                 ],
8143                 "type": "string"
8144             },
8145         },

```



```

8146         "minItems": 1,
8147         "readOnly": true,
8148         "type": "array"
8149     }
8150
8151     },
8152     "required": ["mnpn"]
8153 }
8154 }
8155 }

```

#### E.4.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
mnpn	array: see schema			Platform names
if	array: see schema		Read Only	The interface set supported by this resource
rt	array: see schema		Read Only	Resource Type
id	string		Read Only	Instance ID of this specific resource
n	string		Read Only	Friendly name of the resource
mnpn	array: see schema	yes		Platform names
if	array: see schema		Read Only	The interface set supported by this resource
rt	array: see schema		Read Only	Resource Type
id	string		Read Only	Instance ID of this specific resource
n	string			Friendly name of the resource

#### E.4.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/example/PlatformConfigurationResURI		get	post		

### E.5 Device Configuration

#### E.5.1 Introduction

Resource that allows for Device specific information to be configured. Retrieves the current Device configuration settings

#### E.5.2 Example URI

/example/DeviceConfigurationResURI

#### E.5.3 Resource Type

The resource type (rt) is defined as: ['oic.wk.con'].

#### E.5.4 Swagger2.0 Definition

```

8168 {
8169     "swagger": "2.0",

```

```

8170     "info": {
8171         "title": "Device Configuration",
8172         "version": "v1-20160622",
8173         "license": {
8174             "name": "copyright 2016-2017 Open Connectivity Foundation, Inc. All rights reserved.",
8175             "x-description": "Redistribution and use in source and binary forms, with or without
8176 modification, are permitted provided that the following conditions are met:\n        1.
8177 Redistributions of source code must retain the above copyright notice, this list of conditions and
8178 the following disclaimer.\n        2. Redistributions in binary form must reproduce the above
8179 copyright notice, this list of conditions and the following disclaimer in the documentation and/or
8180 other materials provided with the distribution.\n\n        THIS SOFTWARE IS PROVIDED BY THE Open
8181 Connectivity Foundation, INC. \AS IS\ AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
8182 LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE OR
8183 WARRANTIES OF NON-INFRINGEMENT, ARE DISCLAIMED.\n        IN NO EVENT SHALL THE Open Connectivity
8184 Foundation, INC. OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
8185 EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
8186 OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)\n        HOWEVER CAUSED AND
8187 ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR
8188 OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY
8189 OF SUCH DAMAGE.\n"
8190     }
8191 },
8192 "schemes": ["http"],
8193 "consumes": ["application/json"],
8194 "produces": ["application/json"],
8195 "paths": {
8196     "/exampleDeviceConfigurationResURI" : {
8197         "get": {
8198             "description": "Resource that allows for Device specific information to be
8199 configured.\nRetrieves the current Device configuration settings\n",
8200             "parameters": [
8201                 { "$ref": "#/parameters/interface-all" }
8202             ],
8203             "responses": {
8204                 "200": {
8205                     "description": "",
8206                     "x-example":
8207                     {
8208                         "n": "My Friendly Device Name",
8209                         "rt": ["oic.wk.con"],
8210                         "loc": [32.777,-96.797],
8211                         "locn": "My Location Name",
8212                         "c": "USD",
8213                         "r": "MyRegion",
8214                         "dl": "en"
8215                     }
8216                 },
8217                 "schema": { "$ref": "#/definitions/Configuration" }
8218             }
8219         },
8220     },
8221     "post": {
8222         "description": "Update the information about the Device\n",
8223         "parameters": [
8224             { "$ref": "#/parameters/interface-rw" },
8225             {
8226                 "name": "body",
8227                 "in": "body",
8228                 "required": true,
8229                 "schema": { "$ref": "#/definitions/Update" },
8230                 "x-example":
8231                 {
8232                     "n": "Nuevo Nombre Amistoso",
8233                     "r": "MyNewRegion",
8234                     "ln": [ { "language": "es", "value": "Nuevo Nombre Amistoso" } ],
8235                     "dl": "es"
8236                 }
8237             }
8238         ],
8239         "responses": {
8240             "200": {

```

```

8241         "description" : "",
8242         "x-example":
8243         {
8244             "n": "Nuevo Nombre Amistoso",
8245             "r": "MyNewRegion",
8246             "ln": [ { "language": "es", "value": "Nuevo Nombre Amistoso" } ],
8247             "dl": "es"
8248         }
8249     },
8250     "schema": { "$ref": "#/definitions/Update" }
8251 }
8252 }
8253 }
8254 }
8255 },
8256 "parameters": {
8257     "interface-rw" : {
8258         "in" : "query",
8259         "name" : "if",
8260         "type" : "string",
8261         "enum" : ["oic.if.rw"]
8262     },
8263     "interface-all" : {
8264         "in" : "query",
8265         "name" : "if",
8266         "type" : "string",
8267         "enum" : ["oic.if.rw", "oic.if.baseline"]
8268     }
8269 },
8270 "definitions": {
8271     "Configuration" : {
8272         "properties": {
8273             "rt" :
8274             {
8275                 "description": "Resource Type of the Resource",
8276                 "items": {
8277                     "maxLength": 64,
8278                     "type": "string"
8279                 },
8280                 "minItems": 1,
8281                 "readOnly": true,
8282                 "type": "array"
8283             },
8284             "loc" :
8285             {
8286                 "description": "Location information (lat, long)",
8287                 "items": {
8288                     "type": "number"
8289                 },
8290                 "maxItems": 2,
8291                 "minItems": 2,
8292                 "type": "array"
8293             },
8294             "c" :
8295             {
8296                 "description": "Currency",
8297                 "maxLength": 64,
8298                 "type": "string"
8299             },
8300             "ln" :
8301             {
8302                 "description": "Localized names",
8303                 "items": {
8304                     "properties": {
8305                         "language": {
8306                             "allOf": [
8307                                 {
8308                                     "description": "Format pattern according to IETF RFC 5646 (language tag).",

```

```

8312         "pattern": "^[A-Za-z]{1,8}(-[A-Za-z0-9]{1,8})*$",
8313         "type": "string"
8314     },
8315     {
8316         "description": "An RFC 5646 language tag."
8317     }
8318 ]
8319 },
8320 "value": {
8321     "description": "The Device name in the indicated language.",
8322     "maxLength": 64,
8323     "type": "string"
8324 },
8325 },
8326 "type": "object"
8327 },
8328 "minItems": 1,
8329 "type": "array"
8330 },
8331 "locn" :
8332 {
8333     "description": "Human Friendly Name for location",
8334     "maxLength": 64,
8335     "type": "string"
8336 },
8337 },
8338 "dl" :
8339 {
8340     {
8341         "allof": [
8342             {
8343                 "description": "Format pattern according to IETF RFC 5646 (language tag).",
8344                 "pattern": "^[A-Za-z]{1,8}(-[A-Za-z0-9]{1,8})*$",
8345                 "type": "string"
8346             },
8347             {
8348                 "description": "Default Language as an RFC 5646 language tag."
8349             }
8350         ]
8351     },
8352     "n" :
8353     {
8354         "description": "Friendly name of the resource",
8355         "maxLength": 64,
8356         "readOnly": true,
8357         "type": "string"
8358     },
8359     "r" :
8360     {
8361         "description": "Region",
8362         "maxLength": 64,
8363         "type": "string"
8364     },
8365     "id" :
8366     {
8367         "description": "Instance ID of this specific resource",
8368         "maxLength": 64,
8369         "readOnly": true,
8370         "type": "string"
8371     },
8372     "if" :
8373     {
8374         "description": "The interface set supported by this resource",
8375         "items": {
8376             "enum": [
8377                 "oic.if.baseline",
8378                 "oic.if.ll",

```

```

8383         "oic.if.b",
8384         "oic.if.lb",
8385         "oic.if.rw",
8386         "oic.if.r",
8387         "oic.if.a",
8388         "oic.if.s"
8389     ],
8390     "type": "string"
8391 },
8392 "minItems": 1,
8393 "readOnly": true,
8394 "type": "array"
8395 }
8396 },
8397 "required": ["n"]
8398 }
8399
8400 'Update' : {
8401     "properties": {
8402         "rt" :
8403         {
8404             "description": "Resource Type of the Resource",
8405             "items": {
8406                 "maxLength": 64,
8407                 "type": "string"
8408             },
8409             "minItems": 1,
8410             "readOnly": true,
8411             "type": "array"
8412         },
8413     },
8414     "loc" :
8415     {
8416         "description": "Location information (lat, long)",
8417         "items": {
8418             "type": "number"
8419         },
8420         "maxItems": 2,
8421         "minItems": 2,
8422         "type": "array"
8423     },
8424     "c" :
8425     {
8426         "description": "Currency",
8427         "maxLength": 64,
8428         "type": "string"
8429     },
8430     "ln" :
8431     {
8432         "description": "Localized names",
8433         "items": {
8434             "properties": {
8435                 "language": {
8436                     "allOf": [
8437                         {
8438                             "description": "Format pattern according to IETF RFC 5646 (language tag).",
8439                             "pattern": "^[A-Za-z]{1,8}(-[A-Za-z0-9]{1,8})*$",
8440                             "type": "string"
8441                         },
8442                         {
8443                             "description": "An RFC 5646 language tag."
8444                         }
8445                     ]
8446                 }
8447             },
8448             "value": {
8449                 "description": "The Device name in the indicated language.",
8450                 "maxLength": 64,
8451                 "type": "string"
8452             }
8453         }
8454     }
8455 }

```

```

8454         }
8455     },
8456     "type": "object"
8457 },
8458 "minItems": 1,
8459 "type": "array"
8460 },
8461
8462 "locn" :
8463 {
8464     "description": "Human Friendly Name for location",
8465     "maxLength": 64,
8466     "type": "string"
8467 },
8468
8469 "dl" :
8470 {
8471     "allof": [
8472     {
8473         "description": "Format pattern according to IETF RFC 5646 (language tag).",
8474         "pattern": "^[A-Za-z]{1,8}(-[A-Za-z0-9]{1,8})*$",
8475         "type": "string"
8476     },
8477     {
8478         "description": "Default Language as an RFC 5646 language tag."
8479     }
8480 ]
8481 },
8482
8483 "n" :
8484 {
8485     "description": "Friendly name of the resource",
8486     "maxLength": 64,
8487     "type": "string"
8488 },
8489
8490 "r" :
8491 {
8492     "description": "Region",
8493     "maxLength": 64,
8494     "type": "string"
8495 },
8496
8497 "id" :
8498 {
8499     "anyOf": [
8500     {
8501         "maxLength": 64,
8502         "type": "string"
8503     },
8504     {
8505         "description": "Format pattern according to IETF RFC 4122.",
8506         "pattern": "^[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{12}$",
8507         "type": "string"
8508     }
8509 ],
8510     "description": "Instance ID of this specific resource",
8511     "readOnly": true
8512 },
8513
8514 "if" :
8515 {
8516     "description": "The interface set supported by this resource",
8517     "items": {
8518         "enum": [
8519             "oic.if.baseline",
8520             "oic.if.ll",
8521             "oic.if.b",
8522             "oic.if.lb",
8523             "oic.if.rw",

```

```

8525         "oic.if.r",
8526         "oic.if.a",
8527         "oic.if.s"
8528     ],
8529     "type": "string"
8530 },
8531 "minItems": 1,
8532 "readOnly": true,
8533 "type": "array"
8534 }
8535
8536 }
8537 }
8538 }
8539 }

```

#### E.5.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
locn	string			Human Friendly Name for location
dl	string			Default Language
c	string			Currency
rt	array: see schema		Read Only	Resource Type
id	string		Read Only	Instance ID of this specific resource
ln	array: see schema			Localized names
n	string	yes	Read Only	Friendly name of the resource
loc	array: see schema			Location information
r	string			Region
if	array: see schema		Read Only	The interface set supported by this resource
locn	string			Human Friendly Name for location
dl	string			Default Language
c	string			Currency
rt	array: see schema		Read Only	Resource Type
id	string		Read Only	Instance ID of this specific resource
ln	array: see schema			Localized names
n	string	yes		Friendly name of the resource
loc	array: see schema			Location information
r	string			Region

if	array: see schema		Read Only	The interface set supported by this resource
----	----------------------	--	-----------	--

## E.5.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/example/DeviceConfigurationResURI		get	post		

## E.6 Device

### E.6.1 Introduction

Known resource that is hosted by every Server. Allows for logical device specific information to be discovered. Retrieve the information about the Device

### E.6.2 Wellknown URI

/oic/d

### E.6.3 Resource Type

The resource type (rt) is defined as: ['oic.wk.d'].

### E.6.4 Swagger2.0 Definition

```
{
  "swagger": "2.0",
  "info": {
    "title": "Device",
    "version": "v1-20160622",
    "license": {
      "name": "copyright 2016-2017 Open Connectivity Foundation, Inc. All rights reserved.",
      "x-description": "Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:\n      1.
Redistributions of source code must retain the above copyright notice, this list of conditions and
the following disclaimer.\n      2. Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer in the documentation and/or
other materials provided with the distribution.\n\n      THIS SOFTWARE IS PROVIDED BY THE Open
Connectivity Foundation, INC. \\"AS IS\\" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE OR
WARRANTIES OF NON-INFRINGEMENT, ARE DISCLAIMED.\n      IN NO EVENT SHALL THE Open Connectivity
Foundation, INC. OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)\n      HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR
OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY
OF SUCH DAMAGE.\n"
    }
  },
  "schemes": ["http"],
  "consumes": ["application/json"],
  "produces": ["application/json"],
  "paths": {
    "/oic/d" : {
      "get": {
        "description": "Known resource that is hosted by every Server.\nAllows for logical device
specific information to be discovered.\nRetrieve the information about the Device\n",
        "parameters": [
        ],
        "responses": {
          "200": {
            "description": "",
            "x-example": {
              "n": "Device 1",
              "rt": ["oic.wk.d"],
              "di": "54919CA5-4101-4AE4-595B-353C51AA983C",
```



```

8595         "icv": "ocf.1.0.0",
8596         "dmv": "ocf.res.1.0.0, ocf.sh.1.0.0",
8597         "piid": "6F0AAC04-2BB0-468D-B57C-16570A26AE48"
8598     }
8599     ,
8600     "schema": { "$ref": "#/definitions/Device" }
8601 }
8602 }
8603 }
8604 }
8605 },
8606 "parameters": {
8607     "interface" : {
8608         "in" : "query",
8609         "name" : "if",
8610         "type" : "string",
8611         "enum" : ["oic.if.r", "oic.if.baseline"]
8612     }
8613 },
8614 "definitions": {
8615     "Device" : {
8616         "properties": {
8617             "rt" :
8618                 {
8619                     "description": "Resource Type of the Resource",
8620                     "items": {
8621                         "maxLength": 64,
8622                         "type": "string"
8623                     },
8624                     "minItems": 1,
8625                     "readOnly": true,
8626                     "type": "array"
8627                 },
8628             "ld" :
8629                 {
8630                     "description": "Localized Descriptions.",
8631                     "items": {
8632                         "properties": {
8633                             "language": {
8634                                 "allof": [
8635                                     {
8636                                         "description": "Format pattern according to IETF RFC 5646 (language tag).",
8637                                         "pattern": "^[A-Za-z]{1,8}(-[A-Za-z0-9]{1,8})*$",
8638                                         "type": "string"
8639                                     },
8640                                     {
8641                                         "description": "An RFC 5646 language tag.",
8642                                         "readOnly": true
8643                                     }
8644                                 ]
8645                             }
8646                         },
8647                         "value": {
8648                             "description": "Device description in the indicated language.",
8649                             "maxLength": 64,
8650                             "readOnly": true,
8651                             "type": "string"
8652                         }
8653                     },
8654                     "type": "object"
8655                 },
8656             "minItems": 1,
8657             "readOnly": true,
8658             "type": "array"
8659         },
8660         "piid" :
8661             {
8662                 "allof": [
8663                     {
8664                         "description": "Format pattern according to IETF RFC 4122.",

```

```

8666         "pattern": "^[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-
8667 9]{12}$",
8668         "type": "string"
8669     },
8670     {
8671         "description": "Protocol independent unique identifier for device that is
8672 immutable.",
8673         "readOnly": true
8674     }
8675 ],
8676 },
8677
8678 "di" :
8679 {
8680     "allof": [
8681     {
8682         "description": "Format pattern according to IETF RFC 4122.",
8683         "pattern": "^[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-
8684 9]{12}$",
8685         "type": "string"
8686     },
8687     {
8688         "description": "Unique identifier for device",
8689         "readOnly": true
8690     }
8691     ]
8692 },
8693
8694 "dmno" :
8695 {
8696     "description": "Model number as designated by manufacturer.",
8697     "maxLength": 64,
8698     "readOnly": true,
8699     "type": "string"
8700 },
8701
8702 "sv" :
8703 {
8704     "description": "Software version.",
8705     "maxLength": 64,
8706     "readOnly": true,
8707     "type": "string"
8708 },
8709
8710 "dmn" :
8711 {
8712     "description": "Manufacturer Name.",
8713     "items": {
8714         "properties": {
8715             "language": {
8716                 "allof": [
8717                 {
8718                     "description": "Format pattern according to IETF RFC 5646 (language tag).",
8719                     "pattern": "^[A-Za-z]{1,8}(-[A-Za-z0-9]{1,8})*$",
8720                     "type": "string"
8721                 },
8722                 {
8723                     "description": "An RFC 5646 language tag.",
8724                     "readOnly": true
8725                 }
8726                 ]
8727             },
8728             "value": {
8729                 "description": "Manufacturer name in the indicated language.",
8730                 "maxLength": 64,
8731                 "readOnly": true,
8732                 "type": "string"
8733             }
8734         },
8735         "type": "object"
8736     },

```

```

8737         "minItems": 1,
8738         "readOnly": true,
8739         "type": "array"
8740     },
8741
8742     "icv" :
8743     {
8744         "description": "The version of the OIC Server",
8745         "maxLength": 64,
8746         "readOnly": true,
8747         "type": "string"
8748     },
8749
8750     "dmv" :
8751     {
8752         "description": "Spec versions of the Resource and Device Specifications to which this
8753 device data model is implemented",
8754         "maxLength": 256,
8755         "readOnly": true,
8756         "type": "string"
8757     },
8758
8759     "n" :
8760     {
8761         "description": "Friendly name of the resource",
8762         "maxLength": 64,
8763         "readOnly": true,
8764         "type": "string"
8765     },
8766
8767     "id" :
8768     {
8769         "description": "Instance ID of this specific resource",
8770         "maxLength": 64,
8771         "readOnly": true,
8772         "type": "string"
8773     },
8774
8775     "if" :
8776     {
8777         "description": "The interface set supported by this resource",
8778         "items": {
8779             "enum": [
8780                 "oic.if.baseline",
8781                 "oic.if.ll",
8782                 "oic.if.b",
8783                 "oic.if.lb",
8784                 "oic.if.rw",
8785                 "oic.if.r",
8786                 "oic.if.a",
8787                 "oic.if.s"
8788             ],
8789             "type": "string"
8790         },
8791         "minItems": 1,
8792         "readOnly": true,
8793         "type": "array"
8794     }
8795 },
8796
8797 "required": ["n", "di", "icv", "dmv", "piid"]
8798 }
8799 }
8800 }

```

### E.6.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
dmn	array: see schema		Read Only	Manufacturer Name.

id	string		Read Only	Instance ID of this specific resource
if	array: see schema		Read Only	The interface set supported by this resource
piid	string	yes	Read Only	Protocol independent unique identifier for device (UUID) that is immutable.
ld	array: see schema		Read Only	Localized Description.
dmno	string		Read Only	Model number as designated by manufacturer.
sv	string		Read Only	Software version.
rt	array: see schema		Read Only	Resource Type
dmv	string	yes	Read Only	Spec versions of the Resource and Device Specifications to which this device data model is implemented
icv	string	yes	Read Only	The version of the OIC Server
di	string	yes	Read Only	Unique identifier for device (UUID)
n	string	yes	Read Only	Friendly name of the resource

## E.6.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/oic/d		get			

## E.7 Maintenance

### E.7.1 Introduction

The resource through which a Device is maintained and can be used for diagnostic purposes.

fr (Factory Reset) is a boolean.

The value 0 means No action (Default), the value 1 means Start Factory Reset

After factory reset, this value shall be changed back to the default value

rb (Reboot) is a boolean.

The value 0 means No action (Default), the value 1 means Start Reboot

After Reboot, this value shall be changed back to the default value

Retrieve the maintenance action status

### E.7.2 Wellknown URI

/oic/mnt

## E.7.3 Resource Type

The resource type (rt) is defined as: ['oic.wk.mnt'].

## E.7.4 Swagger2.0 Definition

```
{
  "swagger": "2.0",
  "info": {
    "title": "Maintenance",
    "version": "v1-20160622",
    "license": {
      "name": "copyright 2016-2017 Open Connectivity Foundation, Inc. All rights reserved.",
      "x-description": "Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:\n      1.
Redistributions of source code must retain the above copyright notice, this list of conditions and
the following disclaimer.\n      2. Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer in the documentation and/or
other materials provided with the distribution.\n\n      THIS SOFTWARE IS PROVIDED BY THE Open
Connectivity Foundation, INC. \"AS IS\" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE OR
WARRANTIES OF NON-INFRINGEMENT, ARE DISCLAIMED.\n      IN NO EVENT SHALL THE Open Connectivity
Foundation, INC. OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)\n      HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR
OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY
OF SUCH DAMAGE.\n"
    },
  },
  "schemes": ["http"],
  "consumes": ["application/json"],
  "produces": ["application/json"],
  "paths": {
    "/oic/mnt" : {
      "get": {
        "description": "The resource through which a Device is maintained and can be used for
diagnostic purposes.\nfr (Factory Reset) is a boolean.\n The value 0 means No action (Default),
the value 1 means Start Factory Reset\nAfter factory reset, this value shall be changed back to the
default value\nrb (Reboot) is a boolean.\n The value 0 means No action (Default), the value 1
means Start Reboot\nAfter Reboot, this value shall be changed back to the default value\nRetrieve
the maintenance action status",
        "parameters": [
          { "$ref": "#/parameters/interface-all" }
        ],
        "responses": {
          "200": {
            "description": "",
            "x-example": {
              "rt": ["oic.wk.mnt"],
              "fr": false,
              "rb": false
            }
          },
          "schema": { "$ref": "#/definitions/MNT" }
        }
      },
      "post": {
        "description": "Set the maintenance action(s)\n",
        "parameters": [
          { "$ref": "#/parameters/interface-rw" },
          {
            "name": "body",
            "in": "body",
            "required": true,
            "schema": { "$ref": "#/definitions/MNT" },
            "x-example": {
              "fr": false,
              "rb": false
            }
          }
        ]
      }
    }
  }
}
```

```

8884     }
8885   }
8886 ],
8887 "responses": {
8888   "200": {
8889     "description" : "",
8890     "x-example":
8891     {
8892       "fr": false,
8893       "rb": false
8894     }
8895   },
8896   "schema": { "$ref": "#/definitions/MNT" }
8897 }
8898 }
8899 }
8900 }
8901 },
8902 "parameters": {
8903   "interface-rw" : {
8904     "in" : "query",
8905     "name" : "if",
8906     "type" : "string",
8907     "enum" : ["oic.if.rw", "oic.if.baseline"]
8908   },
8909   "interface-all" : {
8910     "in" : "query",
8911     "name" : "if",
8912     "type" : "string",
8913     "enum" : ["oic.if.rw", "oic.if.r", "oic.if.baseline"]
8914   }
8915 },
8916 "definitions": {
8917   "MNT" : {
8918     "properties": {
8919       "rt" :
8920       {
8921         "description": "Resource Type of the Resource",
8922         "items": {
8923           "maxLength": 64,
8924           "type": "string"
8925         },
8926         "minItems": 1,
8927         "readOnly": true,
8928         "type": "array"
8929       },
8930       "fr" :
8931       {
8932         "description": "Factory Reset",
8933         "type": "boolean"
8934       },
8935       "n" :
8936       {
8937         "description": "Friendly name of the resource",
8938         "maxLength": 64,
8939         "readOnly": true,
8940         "type": "string"
8941       },
8942       "rb" :
8943       {
8944         "description": "Reboot Action",
8945         "type": "boolean"
8946       },
8947       "id" :
8948       {
8949         "description": "Instance ID of this specific resource",
8950         "maxLength": 64,

```

```
8955         "readOnly": true,
8956         "type": "string"
8957     },
8958
8959     "if" :
8960     {
8961         "description": "The interface set supported by this resource",
8962         "items": {
8963             "enum": [
8964                 "oic.if.baseline",
8965                 "oic.if.ll",
8966                 "oic.if.b",
8967                 "oic.if.lb",
8968                 "oic.if.rw",
8969                 "oic.if.r",
8970                 "oic.if.a",
8971                 "oic.if.s"
8972             ],
8973             "type": "string"
8974         },
8975         "minItems": 1,
8976         "readOnly": true,
8977         "type": "array"
8978     }
8979 }
8980 }
8981 }
8982 }
8983 }
```

8984 **E.7.5 Property Definition**

Property name	Value type	Mandatory	Access mode	Description
id	string		Read Only	Instance ID of this specific resource
n	string		Read Only	Friendly name of the resource
rb	boolean	yes		Reboot Action
rt	array: see schema		Read Only	Resource Type
if	array: see schema		Read Only	The interface set supported by this resource
fr	boolean			Factory Reset

8985 **E.7.6 CRUDN behavior**

Resource	Create	Read	Update	Delete	Notify
/oic/mnt		get	post		

8986 **E.8 Platform**

8987 **E.8.1 Introduction**

8988 Known resource that is defines the platform on which an Server is hosted.  
8989 Allows for platform specific information to be discovered.  
8990 Retrieve the information about the Platform  
8991

8992 **E.8.2 Wellknown URI**

8993 /oic/p

8994 **E.8.3 Resource Type**

8995 The resource type (rt) is defined as: ['oic.wk.p'].

## E.8.4 Swagger2.0 Definition

```
{
  "swagger": "2.0",
  "info": {
    "title": "Platform",
    "version": "v1-20160622",
    "license": {
      "name": "copyright 2016-2017 Open Connectivity Foundation, Inc. All rights reserved.",
      "x-description": "Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:\n      1.
Redistributions of source code must retain the above copyright notice, this list of conditions and
the following disclaimer.\n      2. Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer in the documentation and/or
other materials provided with the distribution.\n\n      THIS SOFTWARE IS PROVIDED BY THE Open
Connectivity Foundation, INC. \"AS IS\" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE OR
WARRANTIES OF NON-INFRINGEMENT, ARE DISCLAIMED.\n\n      IN NO EVENT SHALL THE Open Connectivity
Foundation, INC. OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)\n\n      HOWEVER CAUSED AND
ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR
OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY
OF SUCH DAMAGE.\n"
    }
  },
  "schemes": ["http"],
  "consumes": ["application/json"],
  "produces": ["application/json"],
  "paths": {
    "/oic/p" : {
      "get": {
        "description": "Known resource that is defines the platform on which an Server is
hosted.\nAllows for platform specific information to be discovered.\nRetrieve the information about
the Platform\n",
        "parameters": [
        ],
        "responses": {
          "200": {
            "description": "",
            "x-example": {
              "pi": "54919CA5-4101-4AE4-595B-353C51AA983C",
              "rt": ["oic.wk.p"],
              "mmmn": "Acme, Inc"
            }
          },
          "schema": { "$ref": "#/definitions/Platform" }
        }
      }
    }
  },
  "parameters": {
    "interface": {
      "in": "query",
      "name": "if",
      "type": "string",
      "enum": ["oic.if.r", "oic.if.baseline"]
    }
  },
  "definitions": {
    "Platform": {
      "properties": {
        "rt": {
          "description": "Resource Type of the Resource",
          "items": {
            "maxLength": 64,
            "type": "string"
          }
        }
      }
    }
  }
}
```



```

9066         "minItems": 1,
9067         "readOnly": true,
9068         "type": "array"
9069     },
9070
9071     "pi" :
9072     {
9073         "allof": [
9074             {
9075                 "description": "Format pattern according to IETF RFC 4122.",
9076                 "pattern": "^[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-
9077 9]{12}$",
9078                 "type": "string"
9079             },
9080             {
9081                 "description": "Platform Identifier",
9082                 "readOnly": true
9083             }
9084         ]
9085     },
9086
9087     "mnfv" :
9088     {
9089         "description": "Manufacturer's firmware version",
9090         "maxLength": 64,
9091         "readOnly": true,
9092         "type": "string"
9093     },
9094
9095     "vid" :
9096     {
9097         "description": "Manufacturer's defined information for the platform. The content is
9098 freeform, with population rules up to the manufacturer",
9099         "maxLength": 64,
9100         "readOnly": true,
9101         "type": "string"
9102     },
9103
9104     "mnmn" :
9105     {
9106         "description": "Manufacturer Name",
9107         "maxLength": 64,
9108         "readOnly": true,
9109         "type": "string"
9110     },
9111
9112     "mnmo" :
9113     {
9114         "description": "Model number as designated by the manufacturer",
9115         "maxLength": 64,
9116         "readOnly": true,
9117         "type": "string"
9118     },
9119
9120     "mnhw" :
9121     {
9122         "description": "Platform Hardware Version",
9123         "maxLength": 64,
9124         "readOnly": true,
9125         "type": "string"
9126     },
9127
9128     "mnos" :
9129     {
9130         "description": "Platform Resident OS Version",
9131         "maxLength": 64,
9132         "readOnly": true,
9133         "type": "string"
9134     },
9135
9136     "mndt" :

```

```

9137         {
9138         "allof": [
9139         {
9140             "description": "Format pattern as defined in ISO 8601. The format is [yyyy]-[mm]-
9141 [dd].",
9142             "pattern": "^[0-9]{4}-(1[0-2]|0[1-9])-(3[0-1]|2[0-9]|1[0-9]|0[1-9])$",
9143             "type": "string"
9144         },
9145         {
9146             "description": "Manufacturing Date in ISO8601 format.",
9147             "readOnly": true
9148         }
9149     ]
9150 },
9151
9152 "id" :
9153     {
9154         "description": "Instance ID of this specific resource",
9155         "maxLength": 64,
9156         "readOnly": true,
9157         "type": "string"
9158     },
9159
9160 "mnsi" :
9161     {
9162         "description": "Manufacturer's Support Information URL",
9163         "format": "uri",
9164         "maxLength": 256,
9165         "readOnly": true,
9166         "type": "string"
9167     },
9168
9169 "mnpv" :
9170     {
9171         "description": "Platform Version",
9172         "maxLength": 64,
9173         "readOnly": true,
9174         "type": "string"
9175     },
9176
9177 "st" :
9178     {
9179         "description": "Reference time for the device in ISO8601 format.",
9180         "format": "date-time",
9181         "readOnly": true,
9182         "type": "string"
9183     },
9184
9185 "n" :
9186     {
9187         "description": "Friendly name of the resource",
9188         "maxLength": 64,
9189         "readOnly": true,
9190         "type": "string"
9191     },
9192
9193 "mnml" :
9194     {
9195         "description": "Manufacturer's URL",
9196         "format": "uri",
9197         "maxLength": 256,
9198         "readOnly": true,
9199         "type": "string"
9200     },
9201
9202 "if" :
9203     {
9204         "description": "The interface set supported by this resource",
9205         "items": {
9206             "enum": [
9207                 "oic.if.baseline",

```

```

9208         "oic.if.ll",
9209         "oic.if.b",
9210         "oic.if.lb",
9211         "oic.if.rw",
9212         "oic.if.r",
9213         "oic.if.a",
9214         "oic.if.s"
9215     ],
9216     "type": "string"
9217 },
9218     "minItems": 1,
9219     "readOnly": true,
9220     "type": "array"
9221 }
9222 },
9223     "required": ["pi", "mnmn"]
9224 }
9225 }
9226 }
9227 }

```

## E.8.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
n	string		Read Only	Friendly name of the resource
if	array: see schema		Read Only	The interface set supported by this resource
mnmo	string		Read Only	Model number as designated by manufacturer
mpv	string		Read Only	Platform Version
mnos	string		Read Only	Platform Resident OS Version
pi	string	yes	Read Only	Platform Identifier as a UUID
mndt	string		Read Only	Manufacturing Date.
mnfv	string		Read Only	Manufacturer's firmware version
mnml	string		Read Only	Manufacturer's URL
st	string		Read Only	Reference time for the device as defined in ISO 8601, where concatenation of 'date' and 'time' with the 'T' as a delimiter between 'date' and 'time'.
rt	array: see schema		Read Only	Resource Type
mnsi	string		Read Only	Manufacturer's Support Information URL

mnmn	string	yes	Read Only	Manufacturer Name
mnhw	string		Read Only	Platform Hardware Version
vid	string		Read Only	Manufacturer's defined string for the platform. The string is freeform and up to the manufacturer on what text to populate it
id	string		Read Only	Instance ID of this specific resource

## 9229 E.8.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/oic/p		get			

## 9230 E.9 Ping

### 9231 E.9.1 Introduction

9232 The resource using which an Client keeps its Connection with an Server active.  
 9233 Retrieve the ping information

### 9234 E.9.2 Wellknown URI

9235 /oic/ping

### 9236 E.9.3 Resource Type

9237 The resource type (rt) is defined as: ['oic.wk.ping'].

### 9238 E.9.4 Swagger2.0 Definition

```

9239 {
9240   "swagger": "2.0",
9241   "info": {
9242     "title": "Ping",
9243     "version": "v1-20160622",
9244     "license": {
9245       "name": "copyright 2016-2017 Open Connectivity Foundation, Inc. All rights reserved.",
9246       "x-description": "Redistribution and use in source and binary forms, with or without
9247 modification, are permitted provided that the following conditions are met:\n      1.
9248 Redistributions of source code must retain the above copyright notice, this list of conditions and
9249 the following disclaimer.\n      2. Redistributions in binary form must reproduce the above
9250 copyright notice, this list of conditions and the following disclaimer in the documentation and/or
9251 other materials provided with the distribution.\n\n      THIS SOFTWARE IS PROVIDED BY THE Open
9252 Connectivity Foundation, INC. \AS IS\ AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
9253 LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE OR
9254 WARRANTIES OF NON-INFRINGEMENT, ARE DISCLAIMED.\n      IN NO EVENT SHALL THE Open Connectivity
9255 Foundation, INC. OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
9256 EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
9257 OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)\n      HOWEVER CAUSED AND
9258 ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR
9259 OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY
9260 OF SUCH DAMAGE.\n"
9261     }
9262   },
9263   "schemes": ["http"],
9264   "consumes": ["application/json"],
9265   "produces": ["application/json"],
9266   "paths": {

```

```

9267     "/oic/ping" : {
9268         "get": {
9269             "description": "The resource using which an Client keeps its Connection with an Server
9270 active.\nRetrieve the ping information",
9271             "parameters": [
9272                 { "$ref": "#/parameters/interface" }
9273             ],
9274             "responses": {
9275                 "200": {
9276                     "description" : "",
9277                     "x-example":
9278                     {
9279                         "rt": ["oic.wk.ping"],
9280                         "n": "Ping Information",
9281                         "in": 16
9282                     }
9283                 },
9284                 "schema": { "$ref": "#/definitions/PING" }
9285             }
9286         },
9287     },
9288     "post": {
9289         "description": "Update or reset the alive interval",
9290         "parameters": [
9291             { "$ref": "#/parameters/interface",
9292             {
9293                 "name": "body",
9294                 "in": "body",
9295                 "required": true,
9296                 "schema": { "$ref": "#/definitions/PING" },
9297                 "x-example":
9298                 {
9299                     "in": 16
9300                 }
9301             }
9302         ],
9303         "responses": {
9304             "203": {
9305                 "description" : "Successfully updated & restarted alive interval timer.",
9306                 "x-example":
9307                 {
9308                     "in": 16
9309                 }
9310             },
9311             "schema": { "$ref": "#/definitions/PING" }
9312         }
9313     }
9314 },
9315 },
9316 },
9317 "parameters": {
9318     "interface" : {
9319         "in" : "query",
9320         "name" : "if",
9321         "type" : "string",
9322         "enum" : ["oic.if.rw", "oic.if.baseline"]
9323     }
9324 },
9325 "definitions": {
9326     "PING" :
9327     {
9328         "properties": {
9329             "id": {
9330                 "description": "Instance ID of this specific resource",
9331                 "maxLength": 64,
9332                 "readOnly": true,
9333                 "type": "string"
9334             },
9335             "if": {
9336                 "description": "The interface set supported by this resource",
9337                 "items": {

```

```

9338         "enum": [
9339             "oic.if.baseline",
9340             "oic.if.ll",
9341             "oic.if.b",
9342             "oic.if.lb",
9343             "oic.if.rw",
9344             "oic.if.r",
9345             "oic.if.a",
9346             "oic.if.s"
9347         ],
9348         "type": "string"
9349     },
9350     "minItems": 1,
9351     "readOnly": true,
9352     "type": "array"
9353 },
9354 "in": {
9355     "description": "Indicates the interval for which connection shall be kept alive",
9356     "readOnly": false,
9357     "type": "integer"
9358 },
9359 "n": {
9360     "description": "Friendly name of the resource",
9361     "maxLength": 64,
9362     "readOnly": true,
9363     "type": "string"
9364 },
9365 "rt": {
9366     "description": "Resource Type",
9367     "items": {
9368         "maxLength": 64,
9369         "type": "string"
9370     },
9371     "minItems": 1,
9372     "readOnly": true,
9373     "type": "array"
9374 },
9375 },
9376 "required": [
9377     "in"
9378 ],
9379 "type": "object"
9380 }
9381 }
9382 }
9383 }
9384

```

### E.9.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
if	array: see schema		Read Only	The interface set supported by this resource
rt	array: see schema		Read Only	Resource Type
id	string		Read Only	Instance ID of this specific resource
in	integer	yes	Read Write	Indicates the interval for which connection shall be kept alive
n	string		Read Only	Friendly name of the resource

## 9386 E.9.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/oic/ping		get	post		

## 9387 E.10 Resource directory resource

### 9388 E.10.1 Introduction

9389 Resource to be exposed by any Device that can act as a Resource Directory.  
 9390 1) Provides selector criteria (e.g., integer) with GET request  
 9391 2) Publish or Update a Link in /oic/res with POST request  
 9392 3) Delete a Link in /oic/res with DELETE request  
 9393 Get the attributes of the Resource Directory for selection purposes.  
 9394

### 9395 E.10.2 Wellknown URI

9396 /oic/rd

### 9397 E.10.3 Resource Type

9398 The resource type (rt) is defined as: ['oic.wk.rd'].

### 9399 E.10.4 Swagger2.0 Definition

```

9400 {
9401   "swagger": "2.0",
9402   "info": {
9403     "title": "Resource directory resource",
9404     "version": "v1-20160622",
9405     "license": {
9406       "name": "copyright 2016-2017 Open Connectivity Foundation, Inc. All rights reserved.",
9407       "x-description": "Redistribution and use in source and binary forms, with or without
9408 modification, are permitted provided that the following conditions are met:\n      1.
9409 Redistributions of source code must retain the above copyright notice, this list of conditions and
9410 the following disclaimer.\n      2. Redistributions in binary form must reproduce the above
9411 copyright notice, this list of conditions and the following disclaimer in the documentation and/or
9412 other materials provided with the distribution.\n\n      THIS SOFTWARE IS PROVIDED BY THE Open
9413 Connectivity Foundation, INC. \\"AS IS\\" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
9414 LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE OR
9415 WARRANTIES OF NON-INFRINGEMENT, ARE DISCLAIMED.\n\n      IN NO EVENT SHALL THE Open Connectivity
9416 Foundation, INC. OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
9417 EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
9418 OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)\n\n      HOWEVER CAUSED AND
9419 ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR
9420 OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY
9421 OF SUCH DAMAGE.\n"
9422   },
9423   },
9424   "schemes": ["http"],
9425   "consumes": ["application/json"],
9426   "produces": ["application/json"],
9427   "paths": {
9428     "/oic/rd" : {
9429       "get": {
9430         "description": "Resource to be exposed by any Device that can act as a Resource
9431 Directory.\n1) Provides selector criteria (e.g., integer) with GET request\n2) Publish a Link in
9432 /oic/res with POST request\nGet the attributes of the Resource Directory for selection
9433 purposes.\n",
9434         "parameters": [
9435           {"$ref": "#/parameters/rdgetinterface"}
9436         ],
9437         "responses": {
9438           "200": {
9439             "description": "Respond with the selector criteria - either the set of attributes or
9440 the bias factor\n",
9441             "x-example":
9442               {
9443                 "rt": ["oic.wk.rd"],

```

```

9444         "if": ["oic.if.baseline"],
9445         "sel": 50
9446     }
9447 },
9448     "schema": { "$ref": "#/definitions/rdSelection" }
9449 }
9450 },
9451 },
9452     "post": {
9453         "description": "Publish the resource information for the first time in /oic/res. Updates to
9454 existing entries are not allowed.\nAppropriates parts of the information, i.e., Links of the
9455 published Resources will be discovered through /oic/res.\n1) When a Device first publishes a Link,
9456 the request payload to RD may include the Links without an \"ins\" Parameter.\n2) Upon granting the
9457 request, the RD assigns a unique instance value identifying the Link among all the Links it
9458 advertises\n and sends back the instance value in the \"ins\" Parameter in the Link to the
9459 publishing Device.\n",
9460         "parameters": [
9461             { "$ref": "#/parameters/rdpostinterface",
9462             {
9463                 "name": "body",
9464                 "in": "body",
9465                 "required": true,
9466                 "schema": { "$ref": "#/definitions/rdPublish" },
9467                 "x-example":
9468                 {
9469                     "di": "e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",
9470                     "links": [
9471                         {
9472                             "anchor": "ocf://e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",
9473                             "href": "/myLightSwitch",
9474                             "rt": ["oic.r.switch.binary"],
9475                             "if": ["oic.if.a", "oic.if.baseline"],
9476                             "p": {"bm": 3},
9477                             "eps": [
9478                                 {"ep": "coaps://[2001:db8:a::b1d6]:1111", "pri": 2},
9479                                 {"ep": "coaps://[2001:db8:a::b1d6]:1122"},
9480                                 {"ep": "coaps+tcp://[2001:db8:a::123]:2222", "pri": 3}
9481                             ]
9482                         },
9483                         {
9484                             "anchor": "ocf://e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",
9485                             "href": "/myLightBrightness",
9486                             "rt": ["oic.r.brightness"],
9487                             "if": ["oic.if.a", "oic.if.baseline"],
9488                             "p": {"bm": 3},
9489                             "eps": [
9490                                 {"ep": "coaps://[2001:db8:a::123]:2222"}
9491                             ]
9492                         }
9493                     ],
9494                     "ttl": 600
9495                 }
9496             }
9497 ],
9498     "responses": {
9499         "200": {
9500             "description": "Respond with the same schema as publish with the additional \"ins\"
9501 Parameter in the Link.\n",
9502             "x-example":
9503             {
9504                 "di": "e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",
9505                 "links": [
9506                     {
9507                         "anchor": "ocf://e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",
9508                         "href": "/myLightSwitch",
9509                         "rt": ["oic.r.switch.binary"],
9510                         "if": ["oic.if.a", "oic.if.baseline"],
9511                         "p": {"bm": 3},
9512                         "eps": [
9513                             {"ep": "coaps://[2001:db8:a::b1d6]:1111", "pri": 2},
9514                             {"ep": "coaps://[2001:db8:a::b1d6]:1122"},

```



```

9515         {"ep": "coaps+tcp://[2001:db8:a::123]:2222", "pri": 3}
9516     ],
9517     "ins": 11235
9518 },
9519 {
9520     "anchor": "ocf://e61c3e6b-9c54-4b81-8ce5-f9039c1d04d9",
9521     "href": "/myLightBrightness",
9522     "rt": ["oic.r.brightness"],
9523     "if": ["oic.if.a", "oic.if.baseline"],
9524     "p": {"bm": 3},
9525     "eps": [
9526         {"ep": "coaps://[2001:db8:a::123]:2222"}
9527     ],
9528     "ins": 112358
9529 }
9530 ],
9531 "ttl": 600
9532 }
9533 ,
9534 "schema": { "$ref": "#/definitions/rdPublish" }
9535 }
9536 }
9537 }
9538 }
9539 },
9540 "parameters": {
9541     "rdgetinterface": {
9542         "in": "query",
9543         "name": "if",
9544         "type": "string",
9545         "enum": ["oic.if.baseline"],
9546         "description": "enumdescription"
9547     },
9548     "rdpostinterface": {
9549         "in": "query",
9550         "name": "if",
9551         "type": "string",
9552         "enum": ["oic.if.baseline"],
9553         "description": "enumdescription"
9554     }
9555 },
9556 "definitions": {
9557     "rdSelection": {
9558         "properties": {
9559             "rt": {
9560                 {
9561                     "description": "Resource Type of the Resource",
9562                     "items": {
9563                         "maxLength": 64,
9564                         "type": "string"
9565                     },
9566                     "minItems": 1,
9567                     "readOnly": true,
9568                     "type": "array"
9569                 },
9570             },
9571             "n": {
9572                 {
9573                     "description": "Friendly name of the resource",
9574                     "maxLength": 64,
9575                     "readOnly": true,
9576                     "type": "string"
9577                 },
9578             },
9579             "sel": {
9580                 {
9581                     "description": "A bias factor calculated by the Resource directory",
9582                     "maximum": 100,
9583                     "minimum": 0,
9584                     "readOnly": true,
9585                     "type": "integer"

```

```

9586     },
9587
9588     "id" :
9589     {
9590         "description": "Instance ID of this specific resource",
9591         "maxLength": 64,
9592         "readOnly": true,
9593         "type": "string"
9594     },
9595
9596     "if" :
9597     {
9598         "description": "The interface set supported by this resource",
9599         "items": {
9600             "enum": [
9601                 "oic.if.baseline",
9602                 "oic.if.ll",
9603                 "oic.if.b",
9604                 "oic.if.lb",
9605                 "oic.if.rw",
9606                 "oic.if.r",
9607                 "oic.if.a",
9608                 "oic.if.s"
9609             ],
9610             "type": "string"
9611         },
9612         "minItems": 1,
9613         "readOnly": true,
9614         "type": "array"
9615     }
9616
9617 }
9618
9619 ,
9620 "rdPublish" : {
9621     "properties": {
9622         "rt" :
9623         {
9624             "description": "Resource Type of the Resource",
9625             "items": {
9626                 "maxLength": 64,
9627                 "type": "string"
9628             },
9629             "minItems": 1,
9630             "readOnly": true,
9631             "type": "array"
9632         },
9633
9634         "links" :
9635         {
9636             "description": "A set of simple or individual OIC Links.",
9637             "items": {
9638                 "properties": {
9639                     "anchor": {
9640 the containing collection.",
9641                     "format": "uri",
9642                     "maxLength": 256,
9643                     "type": "string"
9644                 },
9645                 "di": {
9646                     "description": "Format pattern according to IETF RFC 4122.",
9647                     "pattern": "^[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{12}$",
9648                     "type": "string"
9649                 },
9650                 "eps": {
9651                     "description": "the Endpoint information of the target Resource",
9652                     "items": {
9653                         "properties": {
9654                             "ep": {

```

```

9657         "description": "Transport Protocol Suite + Endpoint Locator",
9658         "format": "uri",
9659         "type": "string"
9660     },
9661     "pri": {
9662         "description": "The priority among multiple Endpoints",
9663         "minimum": 1,
9664         "type": "integer"
9665     }
9666 },
9667 "type": "object"
9668 },
9669 "type": "array"
9670 },
9671 "href": {
9672     "description": "This is the target URI, it can be specified as a Relative Reference
9673 or fully-qualified URI.",
9674     "format": "uri",
9675     "maxLength": 256,
9676     "type": "string"
9677 },
9678 "if": {
9679     "description": "The interface set supported by this resource",
9680     "items": {
9681         "enum": [
9682             "oic.if.baseline",
9683             "oic.if.ll",
9684             "oic.if.b",
9685             "oic.if.rw",
9686             "oic.if.r",
9687             "oic.if.a",
9688             "oic.if.s"
9689         ],
9690         "type": "string"
9691     },
9692     "minItems": 1,
9693     "type": "array"
9694 },
9695 "ins": {
9696     "description": "The instance identifier for this web link in an array of web links
9697 - used in collections",
9698     "type": "integer"
9699 },
9700 "p": {
9701     "description": "Specifies the framework policies on the Resource referenced by the
9702 target URI",
9703     "properties": {
9704         "bm": {
9705             "description": "Specifies the framework policies on the Resource referenced by
9706 the target URI for e.g. observable and discoverable",
9707             "type": "integer"
9708         }
9709     },
9710     "required": [
9711         "bm"
9712     ],
9713     "type": "object"
9714 },
9715 "rel": {
9716     "description": "The relation of the target URI referenced by the link to the
9717 context URI",
9718     "oneOf": [
9719         {
9720             "default": [
9721                 "hosts"
9722             ],
9723             "items": {
9724                 "maxLength": 64,
9725                 "type": "string"
9726             },
9727             "minItems": 1,

```

```

9728         "type": "array"
9729     },
9730     {
9731         "default": "hosts",
9732         "maxLength": 64,
9733         "type": "string"
9734     }
9735 ]
9736 },
9737 "rt": {
9738     "description": "Resource Type of the Resource",
9739     "items": {
9740         "maxLength": 64,
9741         "type": "string"
9742     },
9743     "minItems": 1,
9744     "type": "array"
9745 },
9746 "title": {
9747     "description": "A title for the link relation. Can be used by the UI to provide a
context.",
9748     "maxLength": 64,
9749     "type": "string"
9750 },
9751 "type": {
9752     "default": "application/cbor",
9753     "description": "A hint at the representation of the resource referenced by the
target URI. This represents the media types that are used for both accepting and emitting.",
9754     "items": {
9755         "maxLength": 64,
9756         "type": "string"
9757     },
9758     "minItems": 1,
9759     "type": "array"
9760 },
9761 },
9762 },
9763 },
9764 "required": [
9765     "href",
9766     "rt",
9767     "if"
9768 ],
9769 "type": "object"
9770 },
9771 "type": "array"
9772 },
9773
9774 "di" :
9775 {
9776     "$ref": "#/definitions/uuid",
9777     "description": "A UUID that is the identifier for the publishing Device"
9778 },
9779
9780 "n" :
9781 {
9782     "description": "Friendly name of the resource",
9783     "maxLength": 64,
9784     "readOnly": true,
9785     "type": "string"
9786 },
9787
9788 "ttl" :
9789 {
9790     "description": "Time to indicate a RD, i.e. how long to keep this published item.",
9791     "type": "integer"
9792 },
9793
9794 "id" :
9795 {
9796     "description": "Instance ID of this specific resource",
9797     "maxLength": 64,
9798     "readOnly": true,

```

```

9799     "type": "string"
9800   },
9801
9802   "if" :
9803     {
9804       "description": "The interface set supported by this resource",
9805       "items": {
9806         "enum": [
9807           "oic.if.baseline",
9808           "oic.if.ll",
9809           "oic.if.b",
9810           "oic.if.lb",
9811           "oic.if.rw",
9812           "oic.if.r",
9813           "oic.if.a",
9814           "oic.if.s"
9815         ],
9816         "type": "string"
9817       },
9818       "minItems": 1,
9819       "readOnly": true,
9820       "type": "array"
9821     }
9822   },
9823   "required": ["di", "links", "ttl"]
9824 }
9825
9826 , "oic.rd.publish" :
9827   {
9828     "properties": {
9829       "di": {
9830         "$ref": "#/definitions/uuid",
9831         "description": "A UUID that is the identifier for the publishing Device"
9832       },
9833       "ttl": {
9834         "description": "Time to indicate a RD, i.e. how long to keep this published item.",
9835         "type": "integer"
9836       }
9837     }
9838   }
9839
9840 , "oic.collection.linksexpanded" :
9841   {
9842     "properties": {
9843       "links": {
9844         "description": "A set of simple or individual OIC Links.",
9845         "items": {
9846           "properties": {
9847             "anchor": {
9848               "description": "This is used to override the context URI e.g. override the URI of
9849 the containing collection.",
9850               "format": "uri",
9851               "maxLength": 256,
9852               "type": "string"
9853             },
9854             "di": {
9855               "description": "Format pattern according to IETF RFC 4122.",
9856               "pattern": "^[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-
9857 9]{12}$",
9858               "type": "string"
9859             },
9860             "eps": {
9861               "description": "the Endpoint information of the target Resource",
9862               "items": {
9863                 "properties": {
9864                   "ep": {
9865                     "description": "Transport Protocol Suite + Endpoint Locator",
9866                     "format": "uri",
9867                     "type": "string"
9868                   },
9869                   "pri": {

```

```

9870         "description": "The priority among multiple Endpoints",
9871         "minimum": 1,
9872         "type": "integer"
9873     },
9874 },
9875     "type": "object"
9876 },
9877     "type": "array"
9878 },
9879     "href": {
9880         "description": "This is the target URI, it can be specified as a Relative Reference
9881 or fully-qualified URI.",
9882         "format": "uri",
9883         "maxLength": 256,
9884         "type": "string"
9885     },
9886     "if": {
9887         "description": "The interface set supported by this resource",
9888         "items": {
9889             "enum": [
9890                 "oic.if.baseline",
9891                 "oic.if.ll",
9892                 "oic.if.b",
9893                 "oic.if.rw",
9894                 "oic.if.r",
9895                 "oic.if.a",
9896                 "oic.if.s"
9897             ],
9898             "type": "string"
9899         },
9900         "minItems": 1,
9901         "type": "array"
9902     },
9903     "ins": {
9904         "description": "The instance identifier for this web link in an array of web links
9905 - used in collections",
9906         "type": "integer"
9907     },
9908     "p": {
9909         "description": "Specifies the framework policies on the Resource referenced by the
9910 target URI",
9911         "properties": {
9912             "bm": {
9913                 "description": "Specifies the framework policies on the Resource referenced by
9914 the target URI for e.g. observable and discoverable",
9915                 "type": "integer"
9916             }
9917         },
9918         "required": [
9919             "bm"
9920         ],
9921         "type": "object"
9922     },
9923     "rel": {
9924         "description": "The relation of the target URI referenced by the link to the
9925 context URI",
9926         "oneOf": [
9927             {
9928                 "default": [
9929                     "hosts"
9930                 ],
9931                 "items": {
9932                     "maxLength": 64,
9933                     "type": "string"
9934                 },
9935                 "minItems": 1,
9936                 "type": "array"
9937             },
9938             {
9939                 "default": "hosts",
9940                 "maxLength": 64,

```

```

9941         "type": "string"
9942     }
9943 ]
9944 },
9945 "rt": {
9946     "description": "Resource Type of the Resource",
9947     "items": {
9948         "maxLength": 64,
9949         "type": "string"
9950     },
9951     "minItems": 1,
9952     "type": "array"
9953 },
9954 "title": {
9955     "description": "A title for the link relation. Can be used by the UI to provide a
9956 context.",
9957     "maxLength": 64,
9958     "type": "string"
9959 },
9960 "type": {
9961     "default": "application/cbor",
9962     "description": "A hint at the representation of the resource referenced by the
9963 target URI. This represents the media types that are used for both accepting and emitting.",
9964     "items": {
9965         "maxLength": 64,
9966         "type": "string"
9967     },
9968     "minItems": 1,
9969     "type": "array"
9970 },
9971 },
9972 "required": [
9973     "href",
9974     "rt",
9975     "if"
9976 ],
9977 "type": "object"
9978 },
9979 "type": "array"
9980 }
9981 },
9982 "type": "object"
9983 }
9984
9985 , "uuid" :
9986 {
9987     "description": "Format pattern according to IETF RFC 4122.",
9988     "pattern": "^[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{12}$",
9989     "type": "string"
9990 }
9991
9992 , "oic.core" :
9993 {
9994     "properties": {
9995         "id": {
9996             "description": "Instance ID of this specific resource",
9997             "maxLength": 64,
9998             "readOnly": true,
9999             "type": "string"
10000         },
10001         "if": {
10002             "description": "The interface set supported by this resource",
10003             "items": {
10004                 "enum": [
10005                     "oic.if.baseline",
10006                     "oic.if.ll",
10007                     "oic.if.b",
10008                     "oic.if.lb",
10009                     "oic.if.rw",
10010                     "oic.if.r",
10011                     "oic.if.a",

```

```

10012         "oic.if.s"
10013     ],
10014     "type": "string"
10015 },
10016 "minItems": 1,
10017 "readOnly": true,
10018 "type": "array"
10019 },
10020 "n": {
10021     "description": "Friendly name of the resource",
10022     "maxLength": 64,
10023     "readOnly": true,
10024     "type": "string"
10025 },
10026 "rt": {
10027     "description": "Resource Type of the Resource",
10028     "items": {
10029         "maxLength": 64,
10030         "type": "string"
10031     },
10032     "minItems": 1,
10033     "readOnly": true,
10034     "type": "array"
10035 },
10036 },
10037 "type": "object"
10038 }
10039 }
10040 }
10041 }

```

## E.10.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
id	string		Read Only	Instance ID of this specific resource
n	string		Read Only	Friendly name of the resource
ttl	integer			Time to indicate a RD, how long to keep this published item. After this time (in seconds) elapses, the RD invalidates the links. To keep link alive the publishing device updates the ttl using the update schema
links	array: see schema			A set (array) of simple or individual OIC Links. In addition to properties required for an OIC Link, the identifier for that link in this set is also required



if	array: see schema	yes	Read Only	The interface set supported by this resource
rt	array: see schema	yes	Read Only	Resource Type
di	string			A unique identifier for the publishing Device, i.e., its device ID
id	string		Read Only	Instance ID of this specific resource
sel	object: see schema	yes		Selection criteria that a device wanting to publish to any RD can use to choose this Resource Directory over others that are discovered
if	array: see schema		Read Only	The interface set supported by this resource
rt	array: see schema		Read Only	Resource Type
n	string		Read Only	Friendly name of the resource

## 10043 E.10.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/oic/rd		get	post	delete	

## 10044 E.11 Discoverable Resources

### 10045 E.11.1 Introduction

10046 Retrieve the discoverable resource set.  
10047

### 10048 E.11.2 Wellknown URI

10049 /oic/res

### 10050 E.11.3 Resource Type

### 10051 E.11.4 Swagger2.0 Definition

```

10052 {
10053   "swagger": "2.0",
10054   "info": {
10055     "title": "Discoverable Resources Link List interface",
10056     "version": "v1-20160622",
10057     "license": {
10058       "name": "copyright 2016-2017 Open Connectivity Foundation, Inc. All rights reserved.",
10059       "x-description": "Redistribution and use in source and binary forms, with or without
10060 modification, are permitted provided that the following conditions are met:\n      1.
10061 Redistributions of source code must retain the above copyright notice, this list of conditions and
10062 the following disclaimer.\n      2. Redistributions in binary form must reproduce the above
10063 copyright notice, this list of conditions and the following disclaimer in the documentation and/or
```

```

10064 other materials provided with the distribution.\n\n      THIS SOFTWARE IS PROVIDED BY THE Open
10065 Connectivity Foundation, INC. \n"AS IS\n" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
10066 LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE OR
10067 WARRANTIES OF NON-INFRINGEMENT, ARE DISCLAIMED.\n      IN NO EVENT SHALL THE Open Connectivity
10068 Foundation, INC. OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
10069 EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
10070 OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)\n      HOWEVER CAUSED AND
10071 ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR
10072 OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY
10073 OF SUCH DAMAGE.\n"
10074 }
10075 },
10076 "schemes": ["http"],
10077 "consumes": ["application/json"],
10078 "produces": ["application/json"],
10079 "paths": {
10080   "/oic/res?if=oic.if.ll" : {
10081     "get": {
10082       "description": "Link list representation of /oic/res; list of discoverable
10083 resources\nRetrieve the discoverable resource set, link list interface\n",
10084       "parameters": [
10085       ],
10086       "responses": {
10087         "200": {
10088           "description" : "",
10089           "x-example":
10090             [
10091               {
10092                 "href": "/humidity",
10093                 "rt": ["oic.r.humidity"],
10094                 "if": ["oic.if.s"],
10095                 "p": {"bm": 3},
10096                 "eps": [
10097                   {"ep": "coaps://[fe80::b1d6]:1111", "pri": 2},
10098                   {"ep": "coaps://[fe80::b1d6]:1122"},
10099                   {"ep": "coaps+tcp://[2001:db8:a::123]:2222", "pri": 3}
10100                 ]
10101               },
10102               {
10103                 "href": "/temperature",
10104                 "rt": ["oic.r.temperature"],
10105                 "if": ["oic.if.s"],
10106                 "p": {"bm": 3},
10107                 "eps": [
10108                   {"ep": "coaps://[[2001:db8:a::123]:2222"}
10109                 ]
10110               }
10111             ]
10112         },
10113         "schema": { "$ref": "#/definitions/slinklist" }
10114       }
10115     }
10116   },
10117 },
10118 "/oic/res?if=oic.if.baseline" : {
10119   "get": {
10120     "description": "Baseline representation of /oic/res; list of discoverable
10121 resources\nRetrieve the discoverable resource set, baseline interface\n",
10122     "parameters": [
10123     ],
10124     "responses": {
10125       "200": {
10126         "description" : "",
10127         "x-example":
10128           [
10129             {
10130               "rt": ["oic.wk.res"],
10131               "if": ["oic.if.baseline", "oic.if.ll" ],
10132               "links":
10133                 [
10134                   {

```

```

10135         "href": "/humidity",
10136         "rt": ["oic.r.humidity"],
10137         "if": ["oic.if.s"],
10138         "p": {"bm": 3},
10139         "eps": [
10140             {"ep": "coaps://[fe80:bld6]:1111", "pri": 2},
10141             {"ep": "coaps://[fe80:bld6]:1122"},
10142             {"ep": "coap+tcp://[2001:db8:a::123]:2222", "pri": 3}
10143         ]
10144     },
10145     {
10146         "href": "/temperature",
10147         "rt": ["oic.r.temperature"],
10148         "if": ["oic.if.s"],
10149         "p": {"bm": 3},
10150         "eps": [
10151             {"ep": "coaps://[2001:db8:a::123]:2222"}
10152         ]
10153     }
10154 ]
10155 }
10156 ]
10157 ,
10158 "schema": { "$ref": "#/definitions/sbaseline" }
10159 }
10160 }
10161 }
10162 },
10163 },
10164 "parameters": {
10165     "interface-ll" : {
10166         "in" : "query",
10167         "name" : "if",
10168         "type" : "string",
10169         "enum" : ["oic.if.ll"]
10170     },
10171     "interface-baseline" : {
10172         "in" : "query",
10173         "name" : "if",
10174         "type" : "string",
10175         "enum" : ["oic.if.baseline"]
10176     },
10177     "interface-all" : {
10178         "in" : "query",
10179         "name" : "if",
10180         "type" : "string",
10181         "enum" : ["oic.if.ll", "oic.if.baseline"]
10182     }
10183 },
10184 "definitions": {
10185     "slinklist" : {
10186         "items" :
10187         {
10188             "properties": {
10189                 "anchor": {
10190                     "description": "This is used to override the context URI e.g. override the URI of the
10191 containing collection.",
10192                     "format": "uri",
10193                     "maxLength": 256,
10194                     "type": "string"
10195                 },
10196                 "di": {
10197                     "allOf": [
10198                         {
10199                             "description": "Format pattern according to IETF RFC 4122.",
10200                             "pattern": "^[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-
10201 9]{12}$",
10202                             "type": "string"
10203                         },
10204                         {
10205                             "description": "The device ID"

```

```

10206     }
10207   ]
10208 },
10209 "eps": {
10210   "description": "the Endpoint information of the target Resource",
10211   "items": {
10212     "properties": {
10213       "ep": {
10214         "description": "Transport Protocol Suite + Endpoint Locator",
10215         "format": "uri",
10216         "type": "string"
10217       },
10218       "pri": {
10219         "description": "The priority among multiple Endpoints",
10220         "minimum": 1,
10221         "type": "integer"
10222       }
10223     },
10224     "type": "object"
10225   },
10226   "type": "array"
10227 },
10228 "href": {
10229   "description": "This is the target URI, it can be specified as a Relative Reference or
10230 fully-qualified URI.",
10231   "format": "uri",
10232   "maxLength": 256,
10233   "type": "string"
10234 },
10235 "if": {
10236   "description": "The interface set supported by this resource",
10237   "items": {
10238     "enum": [
10239       "oic.if.baseline",
10240       "oic.if.ll",
10241       "oic.if.b",
10242       "oic.if.rw",
10243       "oic.if.r",
10244       "oic.if.a",
10245       "oic.if.s"
10246     ],
10247     "type": "string"
10248   },
10249   "minItems": 1,
10250   "type": "array"
10251 },
10252 "ins": {
10253   "description": "The instance identifier for this web link in an array of web links -
10254 used in collections",
10255   "type": "integer"
10256 },
10257 "p": {
10258   "description": "Specifies the framework policies on the Resource referenced by the
10259 target URI",
10260   "properties": {
10261     "bm": {
10262       "description": "Specifies the framework policies on the Resource referenced by the
10263 target URI for e.g. observable and discoverable",
10264       "type": "integer"
10265     }
10266   },
10267   "required": [
10268     "bm"
10269   ],
10270   "type": "object"
10271 },
10272 "rel": {
10273   "description": "The relation of the target URI referenced by the link to the context
10274 URI",
10275   "oneOf": [
10276     {

```

```

10277         "default": [
10278             "hosts"
10279         ],
10280         "items": {
10281             "maxLength": 64,
10282             "type": "string"
10283         },
10284         "minItems": 1,
10285         "type": "array"
10286     },
10287     {
10288         "default": "hosts",
10289         "maxLength": 64,
10290         "type": "string"
10291     }
10292 ]
10293 },
10294 "rt": {
10295     "description": "Resource Type of the Resource",
10296     "items": {
10297         "maxLength": 64,
10298         "type": "string"
10299     },
10300     "minItems": 1,
10301     "type": "array"
10302 },
10303 "title": {
10304     "description": "A title for the link relation. Can be used by the UI to provide a
10305 context.",
10306     "maxLength": 64,
10307     "type": "string"
10308 },
10309 "type": {
10310     "default": "application/cbor",
10311     "description": "A hint at the representation of the resource referenced by the target
10312 URI. This represents the media types that are used for both accepting and emitting.",
10313     "items": {
10314         "maxLength": 64,
10315         "type": "string"
10316     },
10317     "minItems": 1,
10318     "type": "array"
10319 }
10320 },
10321 "required": [
10322     "href",
10323     "rt",
10324     "if"
10325 ],
10326 "type": "object"
10327 }
10328
10329 , "type" :
10330     "array"
10331 }
10332 ,
10333 "sbaseline" : {
10334     "items" :
10335         {
10336             "properties": {
10337                 "if": {
10338                     "description": "The interface set supported by this resource",
10339                     "items": {
10340                         "enum": [
10341                             "oic.if.baseline",
10342                             "oic.if.ll"
10343                         ],
10344                         "type": "string"
10345                     },
10346                 },
10347                 "minItems": 1,

```

```

10348         "readOnly": true,
10349         "type": "array"
10350     },
10351     "links": {
10352         "items": {
10353             "properties": {
10354                 "anchor": {
10355                     "description": "This is used to override the context URI e.g. override the URI of
10356 the containing collection.",
10357                     "format": "uri",
10358                     "maxLength": 256,
10359                     "type": "string"
10360                 },
10361                 "di": {
10362                     "allOf": [
10363                         {
10364                             "description": "Format pattern according to IETF RFC 4122.",
10365                             "pattern": "^[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-
10366 fA-F0-9]{12}$",
10367                             "type": "string"
10368                         },
10369                         {
10370                             "description": "The device ID"
10371                         }
10372                     ]
10373                 },
10374                 "eps": {
10375                     "description": "the Endpoint information of the target Resource",
10376                     "items": {
10377                         "properties": {
10378                             "ep": {
10379                                 "description": "Transport Protocol Suite + Endpoint Locator",
10380                                 "format": "uri",
10381                                 "type": "string"
10382                             },
10383                             "pri": {
10384                                 "description": "The priority among multiple Endpoints",
10385                                 "minimum": 1,
10386                                 "type": "integer"
10387                             }
10388                         },
10389                         "type": "object"
10390                     },
10391                     "type": "array"
10392                 },
10393                 "href": {
10394                     "description": "This is the target URI, it can be specified as a Relative
10395 Reference or fully-qualified URI.",
10396                     "format": "uri",
10397                     "maxLength": 256,
10398                     "type": "string"
10399                 },
10400                 "if": {
10401                     "description": "The interface set supported by this resource",
10402                     "items": {
10403                         "enum": [
10404                             "oic.if.baseline",
10405                             "oic.if.ll",
10406                             "oic.if.b",
10407                             "oic.if.rw",
10408                             "oic.if.r",
10409                             "oic.if.a",
10410                             "oic.if.s"
10411                         ],
10412                         "type": "string"
10413                     },
10414                     "minItems": 1,
10415                     "type": "array"
10416                 },
10417                 "ins": {

```

```

10418         "description": "The instance identifier for this web link in an array of web
10419 links - used in collections",
10420         "type": "integer"
10421     },
10422     "p": {
10423         "description": "Specifies the framework policies on the Resource referenced by
10424 the target URI",
10425         "properties": {
10426             "bm": {
10427                 "description": "Specifies the framework policies on the Resource referenced
10428 by the target URI for e.g. observable and discoverable",
10429                 "type": "integer"
10430             }
10431         },
10432         "required": [
10433             "bm"
10434         ],
10435         "type": "object"
10436     },
10437     "rel": {
10438         "description": "The relation of the target URI referenced by the link to the
10439 context URI",
10440         "oneOf": [
10441             {
10442                 "default": [
10443                     "hosts"
10444                 ],
10445                 "items": {
10446                     "maxLength": 64,
10447                     "type": "string"
10448                 },
10449                 "minItems": 1,
10450                 "type": "array"
10451             },
10452             {
10453                 "default": "hosts",
10454                 "maxLength": 64,
10455                 "type": "string"
10456             }
10457         ],
10458     },
10459     "rt": {
10460         "description": "Resource Type of the Resource",
10461         "items": {
10462             "maxLength": 64,
10463             "type": "string"
10464         },
10465         "minItems": 1,
10466         "type": "array"
10467     },
10468     "title": {
10469         "description": "A title for the link relation. Can be used by the UI to provide a
10470 context.",
10471         "maxLength": 64,
10472         "type": "string"
10473     },
10474     "type": {
10475         "default": "application/cbor",
10476         "description": "A hint at the representation of the resource referenced by the
10477 target URI. This represents the media types that are used for both accepting and emitting.",
10478         "items": {
10479             "maxLength": 64,
10480             "type": "string"
10481         },
10482         "minItems": 1,
10483         "type": "array"
10484     }
10485 },
10486 "required": [
10487     "href",
10488     "rt",

```

```

10489         "if"
10490     ],
10491     "type": "object"
10492 },
10493     "type": "array"
10494 },
10495     "mpro": {
10496         "description": "Supported messaging protocols",
10497         "maxLength": 64,
10498         "readOnly": true,
10499         "type": "string"
10500     },
10501     "n": {
10502         "description": "Human friendly name",
10503         "maxLength": 64,
10504         "readOnly": true,
10505         "type": "string"
10506     },
10507     "rt": {
10508         "description": "Resource Type of the Resource",
10509         "items": {
10510             "maxLength": 64,
10511             "type": "string"
10512         },
10513         "minItems": 1,
10514         "readOnly": true,
10515         "type": "array"
10516     }
10517 },
10518     "required": [
10519         "rt",
10520         "if",
10521         "links"
10522     ],
10523     "type": "object"
10524 }
10525
10526 , "type" :
10527     "array"
10528
10529 }
10530 }
10531 }

```

### E.11.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
links	array: see schema	yes		
mpro	string		Read Only	Supported messaging protocols
if	array: see schema	yes	Read Only	The interface set supported by this resource
rt	array: see schema	yes	Read Only	Resource Type
n	string		Read Only	Human friendly name
p	object: see schema			Specifies the framework policies on the Resource referenced by the target URI



type	array: see schema			A hint at the representation of the resource referenced by the target URI. This represents the media types that are used for both accepting and emitting
if	array: see schema	yes		The interface set supported by this resource
rt	array: see schema	yes		Resource Type
anchor	string			This is used to override the context URI e.g. override the URI of the containing collection
ins	multiple types: see schema			The instance identifier for this web link in an array of web links - used in collections
eps	array: see schema			the Endpoint information of the target Resource
di	string			Unique identifier for device (UUID)
title	string			A title for the link relation. Can be used by the UI to provide a context
href	string	yes		This is the target URI, it can be specified as a Relative Reference or fully-qualified URI. Relative Reference should be used along with the di parameter to make it unique.
rel	multiple types: see schema			The relation of the target URI referenced by the link to the context URI

## 10533 E.11.6 CRUDN behavior

Resource	Create	Read	Update	Delete	Notify
/oic-res- InterfaceURI		get			

## 10534 E.12 Scenes

### 10535 E.12.1 Introduction

10536 Toplevel Scene resource.  
 10537 This resource is a generic collection resource.  
 10538 The rts value shall contain oic.wk.scenecollection resource types.  
 10539 Provides the current list of web links pointing to scenes  
 10540

### 10541 E.12.2 Example URI

10542 /SceneListResURI

### 10543 E.12.3 Resource Type

10544 The resource types (rt) are defined as:  
 10545 ['oic.wk.scenelist','oic.wk.scenemember','oic.wk.scenecollection'].

### 10546 E.12.4 Swagger2.0 Definition

```

10547 {
10548   "swagger": "2.0",
10549   "info": {
10550     "title": "Scenes (Top level)",
10551     "version": "v1-20160622",
10552     "license": {
10553       "name": "copyright 2016-2017 Open Connectivity Foundation, Inc. All rights reserved.",
10554       "x-description": "Redistribution and use in source and binary forms, with or without
10555 modification, are permitted provided that the following conditions are met:\n      1.
10556 Redistributions of source code must retain the above copyright notice, this list of conditions and
10557 the following disclaimer.\n      2. Redistributions in binary form must reproduce the above
10558 copyright notice, this list of conditions and the following disclaimer in the documentation and/or
10559 other materials provided with the distribution.\n\n      THIS SOFTWARE IS PROVIDED BY THE Open
10560 Connectivity Foundation, INC. \AS IS\ AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
10561 LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE OR
10562 WARRANTIES OF NON-INFRINGEMENT, ARE DISCLAIMED.\n      IN NO EVENT SHALL THE Open Connectivity
10563 Foundation, INC. OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
10564 EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
10565 OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)\n      HOWEVER CAUSED AND
10566 ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR
10567 OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY
10568 OF SUCH DAMAGE.\n"
10569     },
10570   },
10571   "schemes": ["http"],
10572   "consumes": ["application/json"],
10573   "produces": ["application/json"],
10574   "paths": {
10575     "/SceneListResURI" : {
10576       "get": {
10577         "description": "Toplevel Scene resource.\nThis resource is a generic collection
10578 resource.\nThe rts value shall contain oic.wk.scenecollection resource types.\nProvides the current
10579 list of web links pointing to scenes\n",
10580         "parameters": [
10581           ],
10582         "responses": {
10583           "200": {
10584             "description" : "",
10585             "x-example":
10586               {
10587                 "rt": ["oic.wk.scenelist"],
10588                 "n": "list of scene Collections",
10589                 "rts": ["oic.wk.scenecollection"],

```

```

10590         "links": [
10591             ]
10592     },
10593     ,
10594     "schema": { "$ref": "#/definitions/Collection" }
10595 }
10596 }
10597 }
10598 },
10599 "/SceneMemberResURI" : {
10600     "get": {
10601         "description": "Collection that models a scene member.\nProvides the scene member\n",
10602         "parameters": [
10603             ],
10604         "responses": {
10605             "200": {
10606                 "description" : "",
10607                 "x-example":
10608                 {
10609                     "rt": ["oic.wk.scenemember"],
10610                     "id": "0685B960-FFFF-46F7-BEC0-9E6234671ADC1",
10611                     "n": "my binary switch (for light bulb) mappings",
10612                     "link": {
10613                         "href": "binarySwitch",
10614                         "rt": ["oic.r.switch.binary"],
10615                         "if": ["oic.if.a", "oic.if.baseline"],
10616                         "eps": [
10617                             { "ep": "coap://[fe80::b1d6]:1111", "pri": 2},
10618                             { "ep": "coaps://[fe80::b1d6]:1122"},
10619                             { "ep": "coap+tcp://[2001:db8:a::123]:2222", "pri": 3}
10620                         ]
10621                     },
10622                     "sceneMappings": [
10623                         {
10624                             "scene": "off",
10625                             "memberProperty": "value",
10626                             "memberValue": true
10627                         },
10628                         {
10629                             "scene": "Reading",
10630                             "memberProperty": "value",
10631                             "memberValue": false
10632                         },
10633                         {
10634                             "scene": "TVWatching",
10635                             "memberProperty": "value",
10636                             "memberValue": true
10637                         }
10638                     ]
10639                 }
10640             },
10641             "schema": { "$ref": "#/definitions/SceneMember" }
10642         }
10643     }
10644 },
10645 },
10646 "/SceneCollectionResURI" : {
10647     "get": {
10648         "description": "Collection that models a set of Scenes.\nThis resource is a generic
10649 collection resource with additional parameters.\nThe rts value shall contain oic.scenemember
10650 resource types.\nThe additional parameters are\n lastScene, this is the scene value last set by
10651 any OCF Client\n sceneValues, this is the list of available scenes\n lastScene shall be listed in
10652 sceneValues.\nProvides the current list of web links pointing to scenes\n",
10653         "parameters": [
10654             ],
10655         "responses": {
10656             "200": {
10657                 "description" : "",
10658                 "x-example":
10659                 {
10660                     "lastScene": "off",

```

```

10661         "sceneValues": ["off","Reading","TVWatching"],
10662         "rt": ["oic.wk.scenecollection"],
10663         "n": "My Scenes for my living room",
10664         "id": "0685B960-736F-46F7-BEC0-9E6CBD671ADC1",
10665         "rts": ["oic.wk.scenemember"],
10666         "links": [
10667         ]
10668     }
10669     ,
10670     "schema": { "$ref": "#/definitions/SceneCollection" }
10671 }
10672 },
10673 },
10674 "post": {
10675     "description": "Provides the action to change the last set scene selection.\nCalling this
10676 method shall update all scene members to the prescribed membervalue.\nWhen this method is called
10677 with the same value as the current lastScene value\nthen all scene members shall be updated.\n",
10678     "parameters": [
10679     {
10680         "name": "body",
10681         "in": "body",
10682         "required": true,
10683         "schema": { "$ref": "#/definitions/SceneCollectionUpdate" },
10684         "x-example":
10685         {
10686             "lastScene": "Reading"
10687         }
10688     }
10689 ],
10690     "responses": {
10691         "200": {
10692             "description": "Indicates that the value is changed.\nThe changed properties are
10693 provided in the response.\n",
10694             "x-example":
10695             {
10696                 "lastScene": "Reading"
10697             }
10698             ,
10699             "schema": { "$ref": "#/definitions/SceneCollectionUpdate" }
10700         }
10701     }
10702 },
10703 },
10704 },
10705 "parameters": {
10706     "interface" : {
10707         "in" : "query",
10708         "name" : "if",
10709         "type" : "string",
10710         "enum" : ["oic.if.a", "oic.if.ll", "oic.if.baseline"]
10711     }
10712 },
10713 "definitions": {
10714     "Collection" : {
10715         "properties": {
10716             "links" :
10717             {
10718                 "description": "A set of simple or individual OIC Links.",
10719                 "items": {
10720                     "$ref": "#/definitions/oic.oic-link"
10721                 },
10722                 "type": "array"
10723             }
10724         }
10725     }
10726 },
10727 ,
10728 "SceneMember" : {
10729     "properties": {
10730         "rt" :
10731         {

```

```

10732         "description": "Resource Type of the Resource",
10733         "items": {
10734             "maxLength": 64,
10735             "type": "string"
10736         },
10737         "minItems": 1,
10738         "readOnly": true,
10739         "type": "array"
10740     },
10741
10742     "SceneMappings" :
10743     {
10744         "description": "array of mappings per scene, can be one(1)",
10745         "items": {
10746             "properties": {
10747                 "memberProperty": {
10748                     "description": "property name that will be mapped",
10749                     "readOnly": true,
10750                     "type": "string"
10751                 },
10752                 "memberValue": {
10753                     "description": "value of the Member Property",
10754                     "readOnly": true,
10755                     "type": "string"
10756                 },
10757                 "scene": {
10758                     "description": "Specifies a scene value that will be acted upon",
10759                     "type": "string"
10760                 }
10761             },
10762             "required": [
10763                 "scene",
10764                 "memberProperty",
10765                 "memberValue"
10766             ],
10767             "type": "object"
10768         },
10769         "type": "array"
10770     },
10771
10772     "n" :
10773     {
10774         "description": "Friendly name of the resource",
10775         "maxLength": 64,
10776         "readOnly": true,
10777         "type": "string"
10778     },
10779
10780     "link" :
10781     {
10782         "allof": [
10783             {
10784                 "properties": {
10785                     "anchor": {
10786 the containing collection.",
10787                     "description": "This is used to override the context URI e.g. override the URI of
10788                     "format": "uri",
10789                     "maxLength": 256,
10790                     "type": "string"
10791                 },
10792                 "di": {
10793                     "allof": [
10794                         {
10795                             "description": "Format pattern according to IETF RFC 4122.",
10796                             "pattern": "^[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-
10797 fa-F0-9]{12}$",
10798                             "type": "string"
10799                         },
10800                         {
10801                             "description": "The device ID"
10802                         }
10803                     ]
10804                 }
10805             }
10806         ]
10807     }

```

```

10803     ]
10804   },
10805   "eps": {
10806     "description": "the Endpoint information of the target Resource",
10807     "items": {
10808       "properties": {
10809         "ep": {
10810           "description": "Transport Protocol Suite + Endpoint Locator",
10811           "format": "uri",
10812           "type": "string"
10813         },
10814         "pri": {
10815           "description": "The priority among multiple Endpoints",
10816           "minimum": 1,
10817           "type": "integer"
10818         }
10819       },
10820       "type": "object"
10821     },
10822     "type": "array"
10823   },
10824   "href": {
10825     "description": "This is the target URI, it can be specified as a Relative
10826 Reference or fully-qualified URI.",
10827     "format": "uri",
10828     "maxLength": 256,
10829     "type": "string"
10830   },
10831   "if": {
10832     "description": "The interface set supported by this resource",
10833     "items": {
10834       "enum": [
10835         "oic.if.baseline",
10836         "oic.if.ll",
10837         "oic.if.b",
10838         "oic.if.rw",
10839         "oic.if.r",
10840         "oic.if.a",
10841         "oic.if.s"
10842       ],
10843       "type": "string"
10844     },
10845     "minItems": 1,
10846     "type": "array"
10847   },
10848   "ins": {
10849     "description": "The instance identifier for this web link in an array of web
10850 links - used in collections",
10851     "type": "integer"
10852   },
10853   "p": {
10854     "description": "Specifies the framework policies on the Resource referenced by
10855 the target URI",
10856     "properties": {
10857       "bm": {
10858         "description": "Specifies the framework policies on the Resource referenced
10859 by the target URI for e.g. observable and discoverable",
10860         "type": "integer"
10861       }
10862     },
10863     "required": [
10864       "bm"
10865     ],
10866     "type": "object"
10867   },
10868   "rel": {
10869     "description": "The relation of the target URI referenced by the link to the
10870 context URI",
10871     "oneOf": [
10872       {
10873         "default": [

```

```

10874         "hosts"
10875     ],
10876     "items": {
10877         "maxLength": 64,
10878         "type": "string"
10879     },
10880     "minItems": 1,
10881     "type": "array"
10882 },
10883 {
10884     "default": "hosts",
10885     "maxLength": 64,
10886     "type": "string"
10887 }
10888 ]
10889 },
10890 "rt": {
10891     "description": "Resource Type of the Resource",
10892     "items": {
10893         "maxLength": 64,
10894         "type": "string"
10895     },
10896     "minItems": 1,
10897     "type": "array"
10898 },
10899 "title": {
10900     "description": "A title for the link relation. Can be used by the UI to provide a
10901 context.",
10902     "maxLength": 64,
10903     "type": "string"
10904 },
10905 "type": {
10906     "default": "application/cbor",
10907     "description": "A hint at the representation of the resource referenced by the
10908 target URI. This represents the media types that are used for both accepting and emitting.",
10909     "items": {
10910         "maxLength": 64,
10911         "type": "string"
10912     },
10913     "minItems": 1,
10914     "type": "array"
10915 }
10916 },
10917 "required": [
10918     "href",
10919     "rt",
10920     "if"
10921 ],
10922 "type": "object"
10923 },
10924 {
10925     "description": "OCF link that points to a resource"
10926 }
10927 ]
10928 },
10929
10930 "id" :
10931 {
10932     "description": "Instance ID of this specific resource",
10933     "maxLength": 64,
10934     "readOnly": true,
10935     "type": "string"
10936 },
10937
10938 "if" :
10939 {
10940     "description": "The interface set supported by this resource",
10941     "items": {
10942         "enum": [
10943             "oic.if.baseline",
10944             "oic.if.ll",

```

```

10945         "oic.if.b",
10946         "oic.if.lb",
10947         "oic.if.rw",
10948         "oic.if.r",
10949         "oic.if.a",
10950         "oic.if.s"
10951     ],
10952     "type": "string"
10953 },
10954 "minItems": 1,
10955 "readOnly": true,
10956 "type": "array"
10957 }
10958
10959 }
10960 }
10961 ,
10962 "SceneCollection" : {
10963     "properties": {
10964         "rt" :
10965         {
10966             "description": "Resource Type of the Resource",
10967             "items": {
10968                 "maxLength": 64,
10969                 "type": "string"
10970             },
10971             "minItems": 1,
10972             "readOnly": true,
10973             "type": "array"
10974         },
10975
10976         "lastScene" :
10977         {
10978             "description": "Last selected Scene from the set of sceneValues",
10979             "type": "string"
10980         },
10981
10982         "links" :
10983         {
10984             "description": "A set of simple or individual OIC Links.",
10985             "items": {
10986                 "properties": {
10987                     "anchor": {
10988                         "description": "This is used to override the context URI e.g. override the URI of
10989 the containing collection.",
10990                         "format": "uri",
10991                         "maxLength": 256,
10992                         "type": "string"
10993                     },
10994                     "di": {
10995                         "description": "Format pattern according to IETF RFC 4122.",
10996                         "pattern": "^[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-
10997 9]{12}$",
10998                         "type": "string"
10999                     },
11000                     "eps": {
11001                         "description": "the Endpoint information of the target Resource",
11002                         "items": {
11003                             "properties": {
11004                                 "ep": {
11005                                     "description": "Transport Protocol Suite + Endpoint Locator",
11006                                     "format": "uri",
11007                                     "type": "string"
11008                                 },
11009                                 "pri": {
11010                                     "description": "The priority among multiple Endpoints",
11011                                     "minimum": 1,
11012                                     "type": "integer"
11013                                 }
11014                             },
11015                             "type": "object"

```



```

11016         },
11017         "type": "array"
11018     },
11019     "href": {
11020         "description": "This is the target URI, it can be specified as a Relative Reference
11021 or fully-qualified URI.",
11022         "format": "uri",
11023         "maxLength": 256,
11024         "type": "string"
11025     },
11026     "if": {
11027         "description": "The interface set supported by this resource",
11028         "items": {
11029             "enum": [
11030                 "oic.if.baseline",
11031                 "oic.if.ll",
11032                 "oic.if.b",
11033                 "oic.if.rw",
11034                 "oic.if.r",
11035                 "oic.if.a",
11036                 "oic.if.s"
11037             ],
11038             "type": "string"
11039         },
11040         "minItems": 1,
11041         "type": "array"
11042     },
11043     "ins": {
11044         "description": "The instance identifier for this web link in an array of web links
11045 - used in collections",
11046         "type": "integer"
11047     },
11048     "p": {
11049         "description": "Specifies the framework policies on the Resource referenced by the
11050 target URI",
11051         "properties": {
11052             "bm": {
11053                 "description": "Specifies the framework policies on the Resource referenced by
11054 the target URI for e.g. observable and discoverable",
11055                 "type": "integer"
11056             }
11057         },
11058         "required": [
11059             "bm"
11060         ],
11061         "type": "object"
11062     },
11063     "rel": {
11064         "description": "The relation of the target URI referenced by the link to the
11065 context URI",
11066         "oneOf": [
11067             {
11068                 "default": [
11069                     "hosts"
11070                 ],
11071                 "items": {
11072                     "maxLength": 64,
11073                     "type": "string"
11074                 },
11075                 "minItems": 1,
11076                 "type": "array"
11077             },
11078             {
11079                 "default": "hosts",
11080                 "maxLength": 64,
11081                 "type": "string"
11082             }
11083         ]
11084     },
11085     "rt": {
11086         "description": "Resource Type of the Resource",

```

```

11087         "items": {
11088             "maxLength": 64,
11089             "type": "string"
11090         },
11091         "minItems": 1,
11092         "type": "array"
11093     },
11094     "title": {
11095         "description": "A title for the link relation. Can be used by the UI to provide a
context.",
11096         "maxLength": 64,
11097         "type": "string"
11098     },
11099     "type": {
11100         "default": "application/cbor",
11101         "description": "A hint at the representation of the resource referenced by the
target URI. This represents the media types that are used for both accepting and emitting.",
11102         "items": {
11103             "maxLength": 64,
11104             "type": "string"
11105         },
11106         "minItems": 1,
11107         "type": "array"
11108     },
11109     "required": [
11110         "href",
11111         "rt",
11112         "if"
11113     ],
11114     "type": "object"
11115 },
11116 "type": "array"
11117 },
11118 "sceneValues" :
11119 {
11120     "description": "All available scene values",
11121     "items": {
11122         "type": "string"
11123     },
11124     "readOnly": true,
11125     "type": "array"
11126 },
11127 "n" :
11128 {
11129     "description": "Friendly name of the resource",
11130     "maxLength": 64,
11131     "readOnly": true,
11132     "type": "string"
11133 },
11134 "rts" :
11135 {
11136     "description": "Resource Type of the Resource",
11137     "items": {
11138         "maxLength": 64,
11139         "type": "string"
11140     },
11141     "minItems": 1,
11142     "readOnly": true,
11143     "type": "array"
11144 },
11145 "id" :
11146 {
11147     "description": "Instance ID of this specific resource",
11148     "maxLength": 64,
11149     "readOnly": true,
11150     "type": "string"
11151 }

```

```

11158     },
11159
11160     "if" :
11161     {
11162         "description": "The interface set supported by this resource",
11163         "items": {
11164             "enum": [
11165                 "oic.if.baseline",
11166                 "oic.if.ll",
11167                 "oic.if.b",
11168                 "oic.if.lb",
11169                 "oic.if.rw",
11170                 "oic.if.r",
11171                 "oic.if.a",
11172                 "oic.if.s"
11173             ],
11174             "type": "string"
11175         },
11176         "minItems": 1,
11177         "readOnly": true,
11178         "type": "array"
11179     }
11180
11181 }
11182
11183 ,
11184 "SceneCollectionUpdate" : {
11185     "properties": {
11186         "rt" :
11187         {
11188             "description": "Resource Type of the Resource",
11189             "items": {
11190                 "maxLength": 64,
11191                 "type": "string"
11192             },
11193             "minItems": 1,
11194             "readOnly": true,
11195             "type": "array"
11196         },
11197
11198         "lastScene" :
11199         {
11200             "description": "Last selected Scene from the set of sceneValues",
11201             "type": "string"
11202         },
11203
11204         "n" :
11205         {
11206             "description": "Friendly name of the resource",
11207             "maxLength": 64,
11208             "readOnly": true,
11209             "type": "string"
11210         },
11211
11212         "id" :
11213         {
11214             "description": "Instance ID of this specific resource",
11215             "maxLength": 64,
11216             "readOnly": true,
11217             "type": "string"
11218         },
11219
11220         "if" :
11221         {
11222             "description": "The interface set supported by this resource",
11223             "items": {
11224                 "enum": [
11225                     "oic.if.baseline",
11226                     "oic.if.ll",
11227                     "oic.if.b",
11228                     "oic.if.lb",

```

```

11229         "oic.if.rw",
11230         "oic.if.r",
11231         "oic.if.a",
11232         "oic.if.s"
11233     ],
11234     "type": "string"
11235 },
11236 "minItems": 1,
11237 "readOnly": true,
11238 "type": "array"
11239 }
11240
11241 }
11242 }
11243 , "uuid" :
11244 {
11245     "description": "Format pattern according to IETF RFC 4122.",
11246     "pattern": "^[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{12}$",
11247     "type": "string"
11248 }
11249
11250 , "oic.collection.properties" :
11251 {
11252     "description": "A collection is a set of links along with additional properties to describe
11253 the collection itself",
11254     "properties": {
11255         "rts": {
11256             "$ref": "#/definitions/oic.core/properties/rt",
11257             "description": "The list of allowable resource types (for Target and anchors) in links
11258 included in the collection"
11259         }
11260     },
11261     "type": "object"
11262 }
11263
11264 , "oic.core" :
11265 {
11266     "properties": {
11267         "rt": {
11268             "description": "Resource Type of the Resource",
11269             "items": {
11270                 "maxLength": 64,
11271                 "type": "string"
11272             },
11273             "minItems": 1,
11274             "readOnly": true,
11275             "type": "array"
11276         }
11277     },
11278     "type": "object"
11279 }
11280
11281 , "oic.collection.linksexpanded" :
11282 {
11283     "properties": {
11284         "links": {
11285             "description": "A set of simple or individual OIC Links.",
11286             "items": {
11287                 "properties": {
11288                     "anchor": {
11289                         "description": "This is used to override the context URI e.g. override the URI of
11290 the containing collection.",
11291                         "format": "uri",
11292                         "maxLength": 256,
11293                         "type": "string"
11294                     },
11295                     "di": {
11296                         "description": "Format pattern according to IETF RFC 4122.",
11297                         "pattern": "^[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-
11298 9]{12}$",
11299                         "type": "string"

```

```

11300     },
11301     "eps": {
11302         "description": "the Endpoint information of the target Resource",
11303         "items": {
11304             "properties": {
11305                 "ep": {
11306                     "description": "Transport Protocol Suite + Endpoint Locator",
11307                     "format": "uri",
11308                     "type": "string"
11309                 },
11310                 "pri": {
11311                     "description": "The priority among multiple Endpoints",
11312                     "minimum": 1,
11313                     "type": "integer"
11314                 }
11315             },
11316             "type": "object"
11317         },
11318         "type": "array"
11319     },
11320     "href": {
11321         "description": "This is the target URI, it can be specified as a Relative Reference
11322 or fully-qualified URI.",
11323         "format": "uri",
11324         "maxLength": 256,
11325         "type": "string"
11326     },
11327     "if": {
11328         "description": "The interface set supported by this resource",
11329         "items": {
11330             "enum": [
11331                 "oic.if.baseline",
11332                 "oic.if.ll",
11333                 "oic.if.b",
11334                 "oic.if.rw",
11335                 "oic.if.r",
11336                 "oic.if.a",
11337                 "oic.if.s"
11338             ],
11339             "type": "string"
11340         },
11341         "minItems": 1,
11342         "type": "array"
11343     },
11344     "ins": {
11345         "description": "The instance identifier for this web link in an array of web links
11346 - used in collections",
11347         "type": "integer"
11348     },
11349     "p": {
11350         "description": "Specifies the framework policies on the Resource referenced by the
11351 target URI",
11352         "properties": {
11353             "bm": {
11354                 "description": "Specifies the framework policies on the Resource referenced by
11355 the target URI for e.g. observable and discoverable",
11356                 "type": "integer"
11357             }
11358         },
11359         "required": [
11360             "bm"
11361         ],
11362         "type": "object"
11363     },
11364     "rel": {
11365         "description": "The relation of the target URI referenced by the link to the
11366 context URI",
11367         "oneOf": [
11368             {
11369                 "default": [
11370                     "hosts"

```

```

11371         ],
11372         "items": {
11373             "maxLength": 64,
11374             "type": "string"
11375         },
11376         "minItems": 1,
11377         "type": "array"
11378     },
11379     {
11380         "default": "hosts",
11381         "maxLength": 64,
11382         "type": "string"
11383     }
11384 ]
11385 },
11386 "rt": {
11387     "description": "Resource Type of the Resource",
11388     "items": {
11389         "maxLength": 64,
11390         "type": "string"
11391     },
11392     "minItems": 1,
11393     "type": "array"
11394 },
11395 "title": {
11396     "description": "A title for the link relation. Can be used by the UI to provide a
11397 context.",
11398     "maxLength": 64,
11399     "type": "string"
11400 },
11401 "type": {
11402     "default": "application/cbor",
11403     "description": "A hint at the representation of the resource referenced by the
11404 target URI. This represents the media types that are used for both accepting and emitting.",
11405     "items": {
11406         "maxLength": 64,
11407         "type": "string"
11408     },
11409     "minItems": 1,
11410     "type": "array"
11411 }
11412 },
11413 "required": [
11414     "href",
11415     "rt",
11416     "if"
11417 ],
11418 "type": "object"
11419 },
11420 "type": "array"
11421 }
11422 },
11423 "type": "object"
11424 }
11425 , "oic.collection.links" :
11426 {
11427     "properties": {
11428         "links": {
11429             "description": "A set of simple or individual OIC Links.",
11430             "items": {
11431                 "$ref": "#/definitions/oic.oic-link"
11432             },
11433             "type": "array"
11434         }
11435     },
11436     "type": "object"
11437 }
11438 }
11439 , "oic.oic-link" :
11440 {
11441

```

```

11442     "properties": {
11443         "anchor": {
11444             "description": "This is used to override the context URI e.g. override the URI of the
11445 containing collection.",
11446             "format": "uri",
11447             "maxLength": 256,
11448             "type": "string"
11449         },
11450         "di": {
11451             "$ref": "#/definitions/uuid",
11452             "description": "The device ID"
11453         },
11454         "eps": {
11455             "description": "the Endpoint information of the target Resource",
11456             "items": {
11457                 "properties": {
11458                     "ep": {
11459                         "description": "Transport Protocol Suite + Endpoint Locator",
11460                         "format": "uri",
11461                         "type": "string"
11462                     },
11463                     "pri": {
11464                         "description": "The priority among multiple Endpoints",
11465                         "minimum": 1,
11466                         "type": "integer"
11467                     }
11468                 },
11469                 "type": "object"
11470             },
11471             "type": "array"
11472         },
11473         "href": {
11474             "description": "This is the target URI, it can be specified as a Relative Reference or
11475 fully-qualified URI.",
11476             "format": "uri",
11477             "maxLength": 256,
11478             "type": "string"
11479         },
11480         "if": {
11481             "description": "The interface set supported by this resource",
11482             "items": {
11483                 "enum": [
11484                     "oic.if.baseline",
11485                     "oic.if.ll",
11486                     "oic.if.b",
11487                     "oic.if.rw",
11488                     "oic.if.r",
11489                     "oic.if.a",
11490                     "oic.if.s"
11491                 ],
11492                 "type": "string"
11493             },
11494             "minItems": 1,
11495             "type": "array"
11496         },
11497         "ins": {
11498             "description": "The instance identifier for this web link in an array of web links - used
11499 in collections",
11500             "type": "integer"
11501         },
11502         "p": {
11503             "description": "Specifies the framework policies on the Resource referenced by the target
11504 URI",
11505             "properties": {
11506                 "bm": {
11507                     "description": "Specifies the framework policies on the Resource referenced by the
11508 target URI for e.g. observable and discoverable",
11509                     "type": "integer"
11510                 }
11511             },
11512             "required": [

```

```

11513         "bm"
11514     ],
11515     "type": "object"
11516 },
11517 "rel": {
11518     "description": "The relation of the target URI referenced by the link to the context
11519 URI",
11520     "oneOf": [
11521         {
11522             "default": [
11523                 "hosts"
11524             ],
11525             "items": {
11526                 "maxLength": 64,
11527                 "type": "string"
11528             },
11529             "minItems": 1,
11530             "type": "array"
11531         },
11532         {
11533             "default": "hosts",
11534             "maxLength": 64,
11535             "type": "string"
11536         }
11537     ],
11538 },
11539 "rt": {
11540     "description": "Resource Type of the Resource",
11541     "items": {
11542         "maxLength": 64,
11543         "type": "string"
11544     },
11545     "minItems": 1,
11546     "type": "array"
11547 },
11548 "title": {
11549     "description": "A title for the link relation. Can be used by the UI to provide a
11550 context.",
11551     "maxLength": 64,
11552     "type": "string"
11553 },
11554 "type": {
11555     "default": "application/cbor",
11556     "description": "A hint at the representation of the resource referenced by the target
11557 URI. This represents the media types that are used for both accepting and emitting.",
11558     "items": {
11559         "maxLength": 64,
11560         "type": "string"
11561     },
11562     "minItems": 1,
11563     "type": "array"
11564 },
11565 },
11566 "required": [
11567     "href",
11568     "rt",
11569     "if"
11570 ],
11571 "type": "object"
11572 }
11573 }
11574 }
11575 }

```

## E.12.5 Property Definition

Property name	Value type	Mandatory	Access mode	Description
lastScene	string	yes		Last selected Scene, shall be



				part of sceneValues
sceneValues	string		Read Only	All available scene values
n	string			Used to name the Scene collection
rts	array: see schema		Read Only	Defines the list of allowable resource types in links included in the collection; new links being created can only be from this list
rt	array: see schema		Read Only	Resource Type
links	array: see schema			Array of OIC web links that are reference from this collection
id	string			A unique string that could be a hash or similarly unique
if	array: see schema		Read Only	The interface set supported by this resource
lastScene	string	yes		Last selected Scene, shall be part of sceneValues
sceneValues	string	yes	Read Only	All available scene values
n	string			Used to name the Scene collection
rts	array: see schema	yes	Read Only	Defines the list of allowable resource types in links included in the collection; new links being created can only be from this list
rt	array: see schema		Read Only	Resource Type
links	array: see schema			Array of OIC web links that are reference from this collection
id	string	yes		A unique string that could be a hash or similarly unique

if	array: see schema		Read Only	The interface set supported by this resource
ins	multiple types: see schema			The instance identifier for this web link in an array of web links - used in collections
type	array: see schema			A hint at the representation of the resource referenced by the target URI. This represents the media types that are used for both accepting and emitting
n	string		Read Only	Friendly name of the resource
rt	array: see schema	yes		Resource Type
id	string		Read Only	Instance ID of this specific resource
drel	string			When specified this is the default relationship to use when an OIC Link does not specify an explicit relationship with *rel* parameter
rel	multiple types: see schema			The relation of the target URI referenced by the link to the context URI
di	string			Unique identifier for device (UUID)
anchor	string			This is used to override the context URI e.g. override the URI of the containing collection
href	string	yes		This is the target URI, it can be specified as a Relative Reference or fully-qualified URI. Relative

				Reference should be used along with the di parameter to make it unique.
eps	array: see schema			the Endpoint information of the target Resource
rts	array: see schema		Read Only	Defines the list of allowable resource types (for Target and anchors) in links included in the collection; new links being created can only be from this list
p	object: see schema			Specifies the framework policies on the Resource referenced by the target URI
links	multiple types: see schema			All forms of links in a collection
title	string			A title for the link relation. Can be used by the UI to provide a context
if	array: see schema	yes		The interface set supported by this resource
SceneMappings	array: see schema			array of mappings per scene, can be 1
n	string			Used to name the Scene collection
link	string	yes		web link that points at a resource
rt	array: see schema		Read Only	Resource Type
id	string			Can be an value that is unique to the use context or a UUIDv4
if	array: see schema		Read Only	The interface set supported by this resource

11577

**E.12.6 CRUDN behavior**

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

/SceneListResURI		get			
------------------	--	-----	--	--	--

