



1
2
3
4

Document Number: DSP-IS0202

Date: 2010-04-15

Version: 1.0.0

5 **CIM-RS Binding to JSON**

- 6 **Document Type: Specification**
- 7 **Document Status: DMTF Informational Specification**
- 8 **Document Language: en-US**

9 IMPORTANT: This specification is not a standard. It is an exploratory, informational document developed
10 in order to obtain industry feedback. It does not reflect the views of the DMTF or all of its members. It is
11 possible that future standards may or may not consider this work product to be an input in whole or in
12 part.

13

14 Copyright Notice

15 Copyright © 2010,2011 Distributed Management Task Force, Inc. (DMTF). All rights reserved.

16 DMTF is a not-for-profit association of industry members dedicated to promoting enterprise and systems
17 management and interoperability. Members and non-members may reproduce DMTF specifications and
18 documents, provided that correct attribution is given. As DMTF specifications may be revised from time to
19 time, the particular version and release date should always be noted.

20 Implementation of certain elements of this standard or proposed standard may be subject to third party
21 patent rights, including provisional patent rights (herein "patent rights"). DMTF makes no representations
22 to users of the standard as to the existence of such rights, and is not responsible to recognize, disclose,
23 or identify any or all such third party patent right, owners or claimants, nor for any incomplete or
24 inaccurate identification or disclosure of such rights, owners or claimants. DMTF shall have no liability to
25 any party, in any manner or circumstance, under any legal theory whatsoever, for failure to recognize,
26 disclose, or identify any such third party patent rights, or for such party's reliance on the standard or
27 incorporation thereof in its product, protocols or testing procedures. DMTF shall have no liability to any
28 party implementing such standard, whether such implementation is foreseeable or not, nor to any patent
29 owner or claimant, and shall have no liability or responsibility for costs or losses incurred if a standard is
30 withdrawn or modified after publication, and shall be indemnified and held harmless by any party
31 implementing the standard from any and all claims of infringement by a patent owner for such
32 implementations.

33 For information about patents held by third-parties which have notified the DMTF that, in their opinion,
34 such patent may relate to or impact implementations of DMTF standards, visit
35 <http://www.dmtf.org/about/policies/disclosures.php>.

36

CONTENTS

37	Foreword	4
38	Introduction	5
39	Document conventions.....	5
40	Typographical conventions	5
41	ABNF usage conventions	5
42	1 Scope	6
43	2 Normative references	6
44	3 Terms and definitions	7
45	4 Symbols and abbreviated terms.....	9
46	5 Concepts	10
47	6 Conformance.....	10
48	7 CIM-RS binding to JSON	10
49	7.1 General	10
50	7.2 Character repertoire, representation, and escaping	11
51	7.3 MIME media type	11
52	7.4 Representation of links	12
53	7.5 Representation of protocol payload elements	13
54	7.5.1 NamespaceCollection payload element	13
55	7.5.2 Namespace payload element	14
56	7.5.3 ClassCollection payload element.....	15
57	7.5.4 Class payload element	16
58	7.5.5 InstanceCollection payload element	19
59	7.5.6 Instance payload element.....	21
60	7.5.7 QualifierTypeCollection payload element	22
61	7.5.8 QualifierType payload element	23
62	7.5.9 MethodInvocationRequest payload element	25
63	7.5.10 MethodInvocationResponse payload element.....	26
64	7.5.11 InstanceModificationRequest payload element.....	27
65	7.5.12 InstanceQueryRequest payload element	27
66	7.5.13 ErrorResponse payload element	28
67	7.6 Representation of values used within payload elements.....	29
68	7.6.1 Method declaration	29
69	7.6.2 Return value declaration.....	30
70	7.6.3 Parameter declaration	30
71	7.6.4 Property declaration.....	31
72	7.6.5 CIM value.....	33
73	7.6.6 Embedded instances	34
74	7.6.7 Embedded objects	35
75	ANNEX A (informative) Change log	37
76	Bibliography	38
77		

78 **Tables**

79	Table 1 – Representation of CIM values by data type.....	33
80	Table 2 – Links of embedded instance	34
81		

82

Foreword

83 The *CIM-RS Binding to JSON* (DSP-IS0202) informational specification was prepared by the DMTF CIM-
84 RS Incubator.

85 DMTF is a not-for-profit association of industry members dedicated to promoting enterprise and systems
86 management and interoperability. For information about the DMTF, see <http://www.dmtf.org>.

87 **Acknowledgments**

88 The DMTF acknowledges the following individuals for their contributions to this document:

- 89 • Andreas Maier, IBM (editor)
- 90 • Cornelia Davis, EMC
- 91 • George Ericson, EMC
- 92 • Johannes Holzer, IBM
- 93 • Robert Kieninger, IBM
- 94 • Larry Lamers, VMware
- 95 • Marvin Waschke, Computer Associates

96

Introduction

97 The information in this document should be sufficient to unambiguously identify the representation of
98 payload elements of the CIM-RS protocol defined in [DSP-IS0201](#), in JavaScript Object Notation (JSON).

99 The target audience for this specification is typically implementers who are writing WBEM servers, WBEM
100 clients, or WBEM listeners supporting the CIM-RS protocol with a payload representation in JSON.
101

102 Document conventions

103 Typographical conventions

104 The following typographical conventions are used in this document:

- 105 • Document titles are marked in *italics*.
- 106 • Important terms that are used for the first time are marked in *italics*.
- 107 • Terms include a link to the term definition in the "Terms and definitions" clause, enabling easy
108 navigation to the term definition.
- 109 • ABNF rules are in `monospaced font`.

110 ABNF usage conventions

111 Format definitions in this document are specified using ABNF (see [RFC5234](#)), with the following
112 deviations:

- 113 • Literal strings are to be interpreted as case-sensitive Unicode characters, as opposed to the
114 definition in [RFC5234](#) that interprets literal strings as case-insensitive US-ASCII characters.

115

116

CIM-RS Binding to JSON

117 1 Scope

118 This informational specification describes the representation of payload elements of the CIM-RS protocol
119 defined in [DSP-IS0201](#) in JavaScript Object Notation (JSON).

120 2 Normative references

121 The following referenced documents are indispensable for the application of this document. For dated or
122 versioned references, only the edition cited (including any corrigenda or DMTF update versions) applies.
123 For references without a date or version, the latest published edition of the referenced document
124 (including any corrigenda or DMTF update versions) applies.
125

126 DMTF DSP0004, *CIM Infrastructure Specification 2.6*,
127 http://www.dmtf.org/standards/published_documents/DSP0004_2.6.pdf

128 DMTF DSP0223, *Generic Operations 1.0*,
129 http://www.dmtf.org/standards/published_documents/DSP0223_1.0.pdf

130 DMTF DSP-IS0201, *CIM Operations Over RESTful Services 1.0*,
131 http://www.dmtf.org/standards/published_documents/DSP-IS0201_1.0.pdf

132 ECMA-262, *ECMAScript Language Specification, 5th Edition, December 2009*,
133 <http://www.ecma-international.org/publications/standards/Ecma-262.htm>
134 For its errata dated 2010-06-24 or later, see
135 http://wiki.ecmascript.org/doku.php?id=es3.1:es3.1_proposal_working_draft

136 IETF RFC4627 (Informational), *The application/json Media Type for JavaScript Object Notation (JSON)*,
137 <http://tools.ietf.org/html/rfc4627>

138 IETF RFC5234, *Augmented BNF for Syntax Specifications: ABNF*,
139 <http://tools.ietf.org/html/rfc5234>

140 IETF RFC-json-schema (Internet-Draft), *A JSON Media Type for Describing the Structure and Meaning of*
141 *JSON Documents*, March 2010,
142 <http://tools.ietf.org/html/draft-zyp-json-schema-02>
143 This version of this document uses a draft of the upcoming draft version 03, available at
144 http://groups.google.com/group/json-schema/browse_thread/thread/5dc7d56a18abd81b

145 ISO/IEC 10646:2003, *Information technology -- Universal Multiple-Octet Coded Character Set (UCS)*,
146 [http://standards.iso.org/ittf/PubliclyAvailableStandards/c039921_ISO_IEC_10646_2003\(E\).zip](http://standards.iso.org/ittf/PubliclyAvailableStandards/c039921_ISO_IEC_10646_2003(E).zip)

147 ISO/IEC Directives, Part 2, *Rules for the structure and drafting of International Standards (2004, 5th*
148 *edition)*,
149 <http://isotc.iso.org/livelink/livelink.exe?func=ll&objId=4230456&objAction=browse>

150 The Unicode Consortium, *The Unicode Standard, Version 5.2.0, Annex #15: Unicode Normalization*
151 *Forms*,
152 <http://www.unicode.org/reports/tr15/>

153 **3 Terms and definitions**

154 In this document, some terms have a specific meaning beyond the normal English meaning. Those terms
155 are defined in this clause.

156 The terms "shall" ("required"), "shall not," "should" ("recommended"), "should not" ("not recommended"),
157 "may," "need not" ("not required"), "can" and "cannot" in this document are to be interpreted as described
158 in [ISO/IEC Directives, Part 2](#), Annex H. The terms in parenthesis are alternatives for the preceding term,
159 for use in exceptional cases when the preceding term cannot be used for linguistic reasons. Note that
160 [ISO/IEC Directives, Part 2](#), Annex H specifies additional alternatives. Occurrences of such additional
161 alternatives shall be interpreted in their normal English meaning.

162 The terms "clause," "subclause," "paragraph," and "annex" in this document are to be interpreted as
163 described in [ISO/IEC Directives, Part 2](#), Clause 5.

164 The terms "normative" and "informative" in this document are to be interpreted as described in [ISO/IEC](#)
165 [Directives, Part 2](#), Clause 3. In this document, clauses, subclauses, or annexes labeled "(informative)" do
166 not contain normative content. Notes and examples are always informative elements.

167 The terms defined in [DSP0004](#) and [DSP0223](#) apply to this document. The following additional terms are
168 used in this document.

169 **3.1**

170 **JSON schema**

171 a description of the structure and meaning of a JSON object, using the syntax defined in [RFC-json-](#)
172 [schema](#)

173 **3.2**

174 **link**

175 a named relation targeting a resource or a resource collection. For details, see 7.4

176 **3.3**

177 **MIME media type**

178 a string identification for media types in Internet protocols, as defined in [RFC2045](#) and [RFC2046](#)

179 **3.4**

180 **Normalization Form C**

181 a normalization form for UCS characters that avoids the use of combining marks where possible and that
182 allows comparing UCS character strings on a per-code-point basis. It is defined in [The Unicode Standard,](#)
183 [Annex #15](#).

184 **3.5**

185 **REST resource**

186 any entity that is accessed through and targeted by a RESTful service

187 **3.6**

188 **RESTful service**

189 a service accessible using REST concepts

- 190 **3.7**
191 **UCS character**
192 a character from the Universal Character Set defined in [ISO/IEC 10646:2003](#). See also [DSP0004](#) for an
193 overview.
- 194 **3.8**
195 **UCS code position**
196 a numeric identification for a UCS character in the range of 0x0 to 0x10FFFF, as defined in [ISO/IEC](#)
197 [10646:2003](#). See also [DSP0004](#) for an overview.
- 198 **3.9**
199 **Unicode character**
200 an alternate term for UCS character
- 201 **3.10**
202 **WBEM client**
203 a CIM client (see [DSP0004](#)) that supports a WBEM protocol. A WBEM client originates WBEM operations
204 for processing by a WBEM server. For details, see [DSP0223](#).
- 205 **3.11**
206 **WBEM indication**
207 an interaction within a WBEM protocol that is originated on a WBEM server and processed by a WBEM
208 listener. For details, see [DSP0223](#). This release of this document does not cover WBEM indications.
- 209 **3.12**
210 **WBEM listener**
211 a CIM listener (see [DSP0004](#)) that supports a WBEM protocol. A WBEM listener processes WBEM
212 indications originated by a WBEM server. For details, see [DSP0223](#).
- 213 **3.13**
214 **WBEM operation**
215 an interaction within a WBEM protocol that is originated by a WBEM client and processed by a WBEM
216 server. For details, see [DSP0223](#).
- 217 **3.14**
218 **WBEM protocol**
219 a communications protocol between WBEM client, WBEM server and WBEM listener. A WBEM protocol
220 defines how the WBEM operations and WBEM indications work, on top of an underlying protocol layer
221 (for example, HTTP, SOAP, or TCP). For details, see [DSP0223](#).
- 222 **3.15**
223 **WBEM resource identifier**
224 a URI that addresses a REST resource in a WBEM server. Some but not all of those REST resources
225 correspond to CIM objects in a WBEM server. Also called "resource identifier" in this document.
- 226 **3.16**
227 **WBEM server**
228 a CIM server (see [DSP0004](#)) that supports a WBEM protocol. A WBEM server processes WBEM
229 operations originated by a WBEM client, and originates WBEM indications for processing by a WBEM
230 listener. For details, see [DSP0223](#).

231 **4 Symbols and abbreviated terms**

232 The abbreviations defined in [DSP0004](#) and [DSP0223](#) apply to this document. The following additional
233 abbreviations are used in this document.

234 **4.1**

235 **ABNF**

236 Augmented Backus-Naur Form, as defined in [RFC5234](#)

237 **4.2**

238 **CIM**

239 Common Information Model, as defined by DMTF

240 **4.3**

241 **CIM-RS**

242 CIM Operations Over RESTful Services, as defined in [DSP-IS0201](#)

243 **4.4**

244 **CQL**

245 CIM Query Language, as defined in [DSP0202](#)

246 **4.5**

247 **ECMAScript**

248 a scripting language that is the standard version of what was called JavaScript. It is defined in [ECMA-
249 262](#).

250 **4.6**

251 **HTTP**

252 Hyper Text Transfer Protocol. HTTP version 1.1 is defined in [RFC2616](#).

253 **4.7**

254 **IANA**

255 Internet Assigned Numbers Authority; see <http://www.iana.org>

256 **4.8**

257 **JSON**

258 JavaScript Object Notation, as defined in [ECMA-262](#)

259 **4.9**

260 **MIME**

261 Multipurpose Internet Mail Extensions, as defined in [IANA MIME Media Types](#)

262 **4.10**

263 **REST**

264 Representational State Transfer, as originally and informally described in [Architectural Styles and the
265 Design of Network-based Software Architectures](#)

266 **4.11**

267 **UCS**

268 Universal Character Set, as defined in [ISO/IEC 10646:2003](#)

269 **4.12**

270 **URI**

271 Uniform Resource Identifier, as defined in [RFC3986](#)

272 **4.13**
273 **UTF-8**
274 UCS Transformation Format 8, as defined in [ISO/IEC 10646:2003](#)

275 **4.14**
276 **WBEM**
277 Web Based Enterprise Management, as defined by DMTF

278 **4.15**
279 **XML**
280 eXtensible Markup Language, as defined by W3C

281 **5 Concepts**

282 This clause defines concepts that are the basis for the definition of the representation of CIM-RS payload
283 elements in JSON.

284 In most cases, the CIM-RS payload elements are represented as JSON objects with attributes that match
285 the generic properties of the payload element as defined in [DSP-IS0201](#).

286 In most cases of using JSON arrays, the array members are named JSON objects based on the value of
287 a generic property defining the name, in order to provide a name-based direct access to the array
288 member.

289 Embedded CIM instances are represented as nested JSON objects.

290 **6 Conformance**

291 A representation of CIM-RS payload elements in JSON conforms to this document only if it conforms to
292 all normative rules stated in this document.

293 The term "CIM-RS representation in JSON" shall be used only for representations of CIM-RS payload
294 elements in JSON that conform to this document.

295 **7 CIM-RS binding to JSON**

296 This clause defines the binding of CIM-RS to JSON.

297 **7.1 General**

298 A CIM-RS payload element represented in JSON shall conform to the grammar defined by the symbol
299 *JSONText* defined in clause 15.12.1 ("The JSON Grammar") of [ECMA-262](#).

300 Care should be taken to distinguish text in [ECMA-262](#) that applies to the JSON language from text that
301 applies to the ECMAScript language. Because the use of JSON for the representation of CIM represents
302 a different usage context than the use of JSON within ECMAScript, rules defined outside of but
303 referenced from within clause 15.12.1 of [ECMA-262](#) shall apply to this document unless otherwise noted.

304 NOTE: Although [RFC4627](#) defines the grammar of the JSON language consistently with clause 15.12.1 of [ECMA-](#)
305 [262](#), [RFC4627](#) is an informational RFC whose purpose is to describe the MIME media type for JSON but not to be
306 the normative definition of the JSON grammar. For this reason, this document references [ECMA-262](#) as the
307 normative definition of the JSON grammar.

308 7.2 Character repertoire, representation, and escaping

309 The JSON grammar defined in clause 15.12.1 of [ECMA-262](#) references the *SourceCharacter* symbol
 310 defined in its clause 6 as the basis for its characters. The definition of the *SourceCharacter* symbol
 311 applies to the ECMAScript use of JSON and uses the character repertoire of Unicode V3, requires a
 312 representation of UCS characters in Normalization Form C, and effectively implies a requirement for an
 313 encoding in UTF-16 (or one of its little endian and big endian derivatives).

314 The following rules apply to a use of the *SourceCharacter* symbol for the representation of CIM-RS
 315 payload elements in JSON:

316 1) The character repertoire of *SourceCharacter* shall be that defined for values of the CIM string
 317 type (defined in [DSP0004](#)).

318 NOTE: That character repertoire is larger than that defined by [ECMA-262](#).

319 2) *SourceCharacter* shall be represented in Normalization Form C.

320 3) *SourceCharacter* shall be encoded in UTF-8.

321 [ECMA-262](#) defines backslash-based escaping for certain characters including the representation of UCS
 322 characters using their UCS code positions. However, in the definition of the *UnicodeEscapeSequence*
 323 symbol in its clause 7.8.4 ("String Literals"), [ECMA-262](#) limits the representation of UCS code positions to
 324 a value range of four hex digits. This is not sufficient for representing the character repertoire defined for
 325 values of the CIM string type (and also not for representing the character repertoire used by [ECMA-262](#)
 326 itself).

327 Therefore, the representation of CIM-RS payload elements in JSON shall support the following extended
 328 definition of the *UnicodeEscapeSequence* symbol:

```
329 UnicodeEscapeSequence ::
330     u HexDigit HexDigit HexDigit HexDigit
331     u HexDigit HexDigit HexDigit HexDigit HexDigit
332     u HexDigit HexDigit HexDigit HexDigit HexDigit HexDigit
```

333 NOTE: This extended definition is consistent with the four-to-six-digit form of the short identifier for UCS characters
 334 defined in clause 6.5 of [ISO/IEC 10646:2003](#) (for example, U+000A, U+12345, and U+10FFFF).

335 7.3 MIME media type

336 [DSP-IS0201](#) requires that CIM-RS payload representation descriptions define a unique MIME media type
 337 that identifies the representation.

338 [RFC4627](#) defines "application/json" as the MIME media type for JSON. This media type is registered with
 339 IANA (see [IANA MIME Media Types](#)). That media type does not allow distinguishing between different
 340 uses of JSON. Because JSON does not have a strong notion of a schema for defining a particular use of
 341 JSON, this document defines a MIME media type that expresses the particular use of JSON for
 342 representing payload elements of the CIM-RS protocol.

343 The MIME media type identifying the representation of CIM-RS payload elements in JSON shall be:

```
344 application/vnd.dmtf.cimrs+json
```

345 The version of this document shall be used to identify the version of the JSON representation. That
 346 version shall be included in the MIME media type as a media type parameter named "version".

347 For example, the current version of the JSON representation results in the following MIME media type:

```
348 application/vnd.dmtf.cimrs+json;version=1.0.0
```

349

350 7.4 Representation of links

351 [DSP-IS0201](#) requires that CIM-RS payload representation descriptions define how links are represented.

352 A set of links shall be represented in JSON as a JSON property named "links" whose value is a JSON
353 object that conforms to the following JSON schema:

```
354 {  
355   "id": "http://schemas.dmtf.org/cimrs/is1/json/is1/links",  
356   "description": "Object type of 'links' property.",  
357   "type": "object",  
358   "properties": {  
359     "links": {  
360       "description": "A set of links.",  
361       "type": "object",  
362       "additionalProperties": {  
363         "description": "Each additional JSON property shall represent a link. The  
364 JSON property name shall be the link name.",  
365         "type": "http://schemas.dmtf.org/cimrs/is1/json/is1/link"  
366       }  
367     }  
368   }  
369 }
```

370 A single link shall be represented in JSON as a JSON property that is named with the link name and
371 whose value is a JSON object that conforms to the following JSON schema:

```
372 {  
373   "id": "http://schemas.dmtf.org/cimrs/is1/json/is1/link",  
374   "description": "Object type of a link property.",  
375   "type": "object",  
376   "properties": {  
377     "href": {  
378       "description": "The target of the link.",  
379       "type": "string",  
380       "format": "uri",  
381       "required": true  
382     }  
383   }  
384 }
```

385 NOTE: This definition for a set of links is similar to that of the *link description object* defined in [RFC-json-structure](#).
386 The difference is the usage of associative arrays in this definition whereby the link set is represented as a JSON
387 object with properties named with the link names, while in [RFC-json-structure](#) the link set is represented as a JSON
388 array with items that are JSON objects whose "rel" property has a value that is the link name. Using the associative
389 array approach has the advantage of providing direct access to a link by link name.

390 EXAMPLE:

```
391 "links": {  
392   "self": {  
393     "href": "http://my.server/my/path"
```

```

394     },
395     "parent": {
396         "href": "http://my.server/my"
397     }
398 }

```

399 This example defines a JSON property representing a set of links consisting of two links: A link named
400 "self" targeting "http://my.server/my/path" and a link named "parent" targeting "http://my.server/my".

401 7.5 Representation of protocol payload elements

402 [DSP-IS0201](#) requires that CIM-RS payload representation descriptions define how the payload elements
403 defined in [DSP-IS0201](#) are represented. This is defined in the subclauses of this clause.

404 7.5.1 NamespaceCollection payload element

405 A NamespaceCollection payload element shall be represented as a JSON object that conforms to the
406 following JSON schema:

```

407 {
408     "id": "http://schemas.dmtf.org/cimrs/is1/json/is1/namespacecollection",
409     "description": "A NamespaceCollection payload element.",
410     "type": "object",
411     "properties": {
412         "links": {
413             "description": "The set of links of this payload element. For the meaning of
414 each link and for the requirement to specify this property and each link, see the
415 CIM-RS operations defined in DSP-IS0201 that use this payload element.",
416             "type": "object",
417             "properties": {
418                 "self": {
419                     "type": "http://schemas.dmtf.org/cimrs/is1/json/is1/link",
420                     "required": true
421                 }
422             }
423         },
424         "namespaces": {
425             "description": "The set of namespaces in the collection. When no child
426 properties are specified, this property may be specified or omitted.",
427             "type": "object",
428             "additionalProperties": {
429                 "description": "Each additional JSON property shall represent a namespace.
430 The JSON property name shall be the namespace name.",
431                 "type": "http://schemas.dmtf.org/cimrs/is1/json/is1/namespace"
432             }
433         }
434     }
435 }

```

436 EXAMPLE:

```

437 {
438     "links": {

```

```

439     "self": {
440         "href": "http://acme.com:5988/cimrs/namespaces"}
441     },
442     "namespaces": {
443         "root/cimv2": { # A namespace, as defined in 7.5.2.
444             "links": {
445                 "self": {
446                     "href": "http://acme.com:5988/cimrs/namespaces/root%2Fcimv2"},
447                 "classes": {
448                     "href": "classes"},
449                 "qualifiers": {
450                     "href": "qualifiers"},
451                 "instancequery": {
452                     "href": "instancequery"}
453             }
454         },
455         . . . # more namespaces
456     }
457 }

```

458 7.5.2 Namespace payload element

459 A Namespace payload element shall be represented as a JSON property that is named with the
460 namespace name and whose value is a JSON object that conforms to the following JSON schema:

```

461 {
462     "id": "http://schemas.dmtf.org/cimrs/is1/json/is1/namespace",
463     "description": "Object type for the namespace property of a Namespace payload
464 element.",
465     "type": "object",
466     "properties": {
467         "links": {
468             "description": "The set of links of this payload element. For the meaning of
469 each link and for the requirement to specify this property and each link, see the
470 CIM-RS operations defined in DSP-IS0201 that use this payload element.",
471             "type": "object",
472             "properties": {
473                 "self": {
474                     "type": "http://schemas.dmtf.org/cimrs/is1/json/is1/link",
475                     "required": true
476                 },
477                 "classes": {
478                     "type": "http://schemas.dmtf.org/cimrs/is1/json/is1/link",
479                     "required": true
480                 },
481                 "qualifiers": {
482                     "type": "http://schemas.dmtf.org/cimrs/is1/json/is1/link",
483                     "required": true
484                 },
485                 "instancequery": {

```

```

486     "type": "http://schemas.dmtf.org/cimrs/is1/json/is1/link",
487     "required": true
488   }
489 }
490 }
491 }
492 }

```

493 EXAMPLE:

```

494 "root/cimv2": {
495   "links": {
496     "self": {
497       "href": "http://acme.com:5988/cimrs/namespaces/root%2Fcimv2"},
498     "classes": {
499       "href": "classes"},
500     "qualifiers": {
501       "href": "qualifiers"},
502     "instancequery": {
503       "href": "instancequery"}
504   }
505 }

```

506 7.5.3 ClassCollection payload element

507 A ClassCollection payload element shall be represented as a JSON object that conforms to the following
508 JSON schema:

```

509 {
510   "id": "http://schemas.dmtf.org/cimrs/is1/json/is1/classcollection",
511   "description": "A ClassCollection payload element.",
512   "type": "object",
513   "properties": {
514     "links": {
515       "description": "The set of links of this payload element. For the meaning of
516 each link and for the requirement to specify this property and each link, see the
517 CIM-RS operations defined in DSP-IS0201 that use this payload element.",
518       "type": "object",
519       "properties": {
520         "self": {
521           "type": "http://schemas.dmtf.org/cimrs/is1/json/is1/link",
522           "required": true
523         },
524         "namespace": {
525           "type": "http://schemas.dmtf.org/cimrs/is1/json/is1/link",
526           "required": true
527         }
528       }
529     },
530     "classes": {
531       "description": "The set of classes in the collection. When no child
532 properties are specified, this property may be specified or omitted.",

```

```

533     "type": "object",
534     "additionalProperties": {
535         "description": "Each additional JSON property shall represent a class. The
536 JSON property name shall be the class name.",
537         "type": "http://schemas.dmtf.org/cimrs/is1/json/is1/class"
538     }
539 }
540 }
541 }

```

542 EXAMPLE:

```

543 {
544     "links": {
545         "self": {
546             "href": "http://acme.com:5988/cimrs/namespaces/root%2Fcimv2/classes",
547             "namespace": {
548                 "href": ".."}
549         },
550     "classes": {
551         "CIM_SubClass1OfExample": { # A class, as defined in 7.5.4.
552             "links": {
553                 "self": {
554                     "href": "http://acme.com:5988/cimrs/namespaces/root%2Fcimv2/classes/CIM_S
555 ubClass1OfExample"},
556                 "namespace": {
557                     "href": "../.."},
558                 "instances": {
559                     "href": "instances"},
560                 "methodinvocation": {
561                     "href": "methodinvocation"},
562                 "associators": {
563                     "href": "associators"},
564                 "references": {
565                     "href": "references"}
566             },
567             "superclass": "CIM_Example",
568             "qualifiers": { . . . }, # for details, see 7.5.4.
569             "methods": { . . . }, # for details, see 7.5.4.
570             "properties": { . . . } # for details, see 7.5.4.
571         },
572         . . . # more classes
573     }
574 }

```

575 7.5.4 Class payload element

576 A Class payload element shall be represented as a JSON property that is named with the class name and
577 whose value is a JSON object that conforms to the following JSON schema:

```
578 {
```



```
579 "id": "http://schemas.dmtf.org/cimrs/is1/json/is1/class",
580 "description": "Object type for the class property of a Class payload element.",
581 "type": "object",
582 "properties": {
583   "links": {
584     "description": "The set of links of this payload element. For the meaning of
585     each link and for the requirement to specify this property and each link, see the
586     CIM-RS operations defined in DSP-IS0201 that use this payload element.",
587     "type": "object",
588     "properties": {
589       "self": {
590         "type": "http://schemas.dmtf.org/cimrs/is1/json/is1/link",
591         "required": true
592       },
593       "namespace": {
594         "type": "http://schemas.dmtf.org/cimrs/is1/json/is1/link",
595         "required": true
596       },
597       "instances": {
598         "type": "http://schemas.dmtf.org/cimrs/is1/json/is1/link",
599         "required": true
600       },
601       "methodinvocation": {
602         "type": "http://schemas.dmtf.org/cimrs/is1/json/is1/link",
603         "required": true
604       },
605       "associators": {
606         "type": "http://schemas.dmtf.org/cimrs/is1/json/is1/link",
607         "required": true
608       },
609       "references": {
610         "type": "http://schemas.dmtf.org/cimrs/is1/json/is1/link",
611         "required": true
612       }
613     },
614   },
615   "superclass": {
616     "description": "The name of the superclass of this class, if any.",
617     "type": "string",
618     "format": "cim-classname"
619   },
620   "qualifiers": {
621     "description": "The set of qualifier values of this class. When no child
622     properties are specified, this property may be specified or omitted.",
623     "type": "object",
624     "additionalProperties": {
625       "description": "Each additional JSON property shall represent a qualifier
626       value. The JSON property name shall be the qualifier name. For details on the JSON
627       type used to represent the value, see 7.6.5.",
628       "type": "any"
```

```

629     }
630   },
631   "methods": {
632     "description": "The set of method declarations of this class. When no child
633 properties are specified, this property may be specified or omitted.",
634     "type": "object",
635     "additionalProperties": {
636       "description": "Each additional JSON property shall represent a method
637 declaration. The JSON property name shall be the method name.",
638       "type": "http://schemas.dmtf.org/cimrs/is1/json/is1/methoddeclaration"
639     }
640   },
641   "properties": {
642     "description": "The set of property declarations of this class. When no child
643 properties are specified, this property may be specified or omitted.",
644     "type": "object",
645     "additionalProperties": {
646       "description": "Each additional JSON property shall represent a CIM
647 property declaration. The JSON property name shall be the CIM property name.",
648       "type": "http://schemas.dmtf.org/cimrs/is1/json/is1/propertydeclaration"
649     }
650   }
651 }
652 }

```

653 EXAMPLE:

```

654 "CIM_RegisteredProfile": {
655   "links": {
656     "self": {
657       "href": "http://acme.com:5988/cimrs/namespaces/root%2Fcimv2/classes/CIM_Regis
658 teredProfile"},
659     "namespace": {
660       "href": "../.."},
661     "instances": {
662       "href": "instances"},
663     "methodinvocation": {
664       "href": "methodinvocation"},
665     "associators": {
666       "href": "associators"},
667     "references": {
668       "href": "references"}
669   },
670   "superclass": "CIM_RegisteredSpecification",
671   "qualifiers": {
672     "Version": "2.26.0",
673     . . . # more qualifier value specifications on the class
674   },
675   "methods": {
676     "CloseConformantInstances": {
677       "qualifiers": {

```

```

678     "Experimental": true,
679     . . . # more qualifier value specifications on the method
680   },
681   "returnvalue": {
682     "type": "uint32"
683   },
684   "parameters": {
685     "EnumerationContext": {
686       "qualifiers": {
687         "In": true,
688         . . . # more qualifier value specifications on the parameter
689       },
690       "type": "string"
691     },
692     . . . # more parameter declarations
693   }
694 },
695 . . . # more method declarations
696 },
697 "properties": {
698   "SpecificationType": {
699     "qualifiers": {
700       "Override": "SpecificationType",
701       "ValueMap": ["2"],
702       "Values": ["Profile"],
703       . . . # more qualifier value specifications on the property
704     },
705     "type": "uint16"
706   },
707   . . . # more property declarations
708 }
709 }

```

710 7.5.5 InstanceCollection payload element

711 An InstanceCollection payload element shall be represented as a JSON object that conforms to the
712 following JSON schema:

```

713 {
714   "id": "http://schemas.dmtf.org/cimrs/is1/json/is1/instancecollection",
715   "description": "An InstanceCollection payload element.",
716   "type": "object",
717   "properties": {
718     "links": {
719       "description": "The set of links of this payload element. For the meaning of
720 each link and for the requirement to specify this property and each link, see the
721 CIM-RS operations defined in DSP-IS0201 that use this payload element.",
722       "type": "object",
723       "properties": {
724         "self": {
725           "type": "http://schemas.dmtf.org/cimrs/is1/json/is1/link",

```

```

726     "required": true
727   },
728   "class": {
729     "type": "http://schemas.dmtf.org/cimrs/is1/json/is1/link",
730     "required": true
731   }
732 }
733 },
734 "instances": {
735   "description": "The set of instances in the collection. When no child
736 properties are specified, this property may be specified or omitted.",
737   "type": "array",
738   "items": "http://schemas.dmtf.org/cimrs/is1/json/is1/instance"
739 }
740 }
741 }

```

742 EXAMPLE:

```

743 {
744   "links": {
745     "self": {
746       "href": "http://acme.com:5988/cimrs/namespaces/root%2Fcimv2/classes/CIM_Sourc
747 eExample/instances"},
748     "class": {
749       "href": ".."}
750   },
751   "instances": [
752     { # An instance, as defined in 7.5.6.
753       "links": {
754         "self": {
755           "href": "http://acme.com:5988/cimrs/namespaces/root%2Fcimv2/classes/CIM_R
756 egisteredProfile/instances/DMTF%3AFan%3A1.1.0"},
757         "class": {
758           "href": "../.."},
759         "methodinvocation": {
760           "href": "methodinvocation"},
761         "associators": {
762           "href": "associators"},
763         "references": {
764           "href": "references"}
765       },
766       "class": "CIM_RegisteredProfile",
767       "properties": {
768         "InstanceID": "DMTF:Fan:1.1.0",
769         "RegisteredName": "Fan",
770         "RegisteredOrganization": 2,
771         "RegisteredVersion": "1.1.0",
772         . . . # more properties
773     }

```

```

774     },
775     . . . # more instances
776 ]
777 }

```

778 7.5.6 Instance payload element

779 An Instance payload element shall be represented as a JSON object that conforms to the following JSON
780 schema:

```

781 {
782   "id": "http://schemas.dmtf.org/cimrs/is1/json/is1/instance",
783   "description": "An Instance payload element.",
784   "type": "object",
785   "properties": {
786     "links": {
787       "description": "The set of links of this payload element. For the meaning of
788       each link and for the requirement to specify this property and each link, see the
789       CIM-RS operations defined in DSP-IS0201 that use this payload element.",
790       "type": "object",
791       "properties": {
792         "self": {
793           "type": "http://schemas.dmtf.org/cimrs/is1/json/is1/link"
794         },
795         "class": {
796           "type": "http://schemas.dmtf.org/cimrs/is1/json/is1/link"
797         },
798         "methodinvocation": {
799           "type": "http://schemas.dmtf.org/cimrs/is1/json/is1/link"
800         },
801         "associators": {
802           "type": "http://schemas.dmtf.org/cimrs/is1/json/is1/link"
803         },
804         "references": {
805           "type": "http://schemas.dmtf.org/cimrs/is1/json/is1/link"
806         }
807       }
808     },
809     "class": {
810       "description": "The name of the creation class of this instance.",
811       "type": "string",
812       "format": "cim-classname",
813       "required": true
814     },
815     "properties": {
816       "description": "The set of property values of this instance. When no child
817       properties are specified, this property may be specified or omitted.",
818       "type": "object",
819       "additionalProperties": {
820         "description": "Each additional JSON property shall represent a property
821         value. The JSON property name shall be the CIM property name. For details on the

```

```

822     JSON type used to represent the value, see 7.6.5.",
823     "type": "any"
824   }
825 }
826 }
827 }

```

828 EXAMPLE:

```

829 {
830   "links": {
831     "self": {
832       "href": "http://acme.com:5988/cimrs/namespaces/root%2Fcimv2/classes/CIM_Regis
833   teredProfile/instances/DMTF%3AFan%3A1.1.0"},
834     "class": {
835       "href": "../.."},
836     "methodinvocation": {
837       "href": "methodinvocation"},
838     "associators": {
839       "href": "associators"},
840     "references": {
841       "href": "references"}
842   },
843   "class": "CIM_RegisteredProfile",
844   "properties": {
845     "InstanceID": "DMTF:Fan:1.1.0",
846     "RegisteredName": "Fan",
847     "RegisteredOrganization": 2,
848     "RegisteredVersion": "1.1.0",
849     . . . # more properties
850   }
851 }

```

852 7.5.7 QualifierTypeCollection payload element

853 A QualifierTypeCollection payload element shall be represented as a JSON object that conforms to the
854 following JSON schema:

```

855 {
856   "id": "http://schemas.dmtf.org/cimrs/is1/json/is1/qualifiertypecollection",
857   "description": "A QualifierTypeCollection payload element.",
858   "type": "object",
859   "properties": {
860     "links": {
861       "description": "The set of links of this payload element. For the meaning of
862   each link and for the requirement to specify this property and each link, see the
863   CIM-RS operations defined in DSP-IS0201 that use this payload element.",
864       "type": "object",
865       "properties": {
866         "self": {
867           "type": "http://schemas.dmtf.org/cimrs/is1/json/is1/link",
868           "required": true

```

```

869     },
870     "namespace": {
871         "type": "http://schemas.dmtf.org/cimrs/is1/json/is1/link",
872         "required": true
873     }
874 }
875 },
876 "qualifiertypes": {
877     "description": "The set of qualifier types in the collection. When no child
878 properties are specified, this property may be specified or omitted.",
879     "type": "object",
880     "additionalProperties": {
881         "description": "Each additional JSON property shall represent a qualifier
882 type. The JSON property name shall be the qualifier name.",
883         "type": "http://schemas.dmtf.org/cimrs/is1/json/is1/qualifiertype"
884     }
885 }
886 }
887 }

```

888 EXAMPLE:

```

889 {
890     "links": {
891         "self": {
892             "href": "http://acme.com:5988/cimrs/namespaces/root%2Fcimv2/qualifiertypes"},
893         "namespace": {
894             "href": ".."}
895     },
896     "qualifiertypes": {
897         "Abstract": { # A qualifier type, as defined in 7.5.8.
898             "links": {
899                 "self": {
900                     "href": "http://acme.com:5988/cimrs/namespaces/root%2Fcimv2/qualifiers/Ab
901 stract"},
902                 "namespace": {
903                     "href": "../.."}
904             },
905             "type": "boolean",
906             "default": false,
907             "scope": [ "class", "association", "indication" ],
908             "flavor": [ "Restricted" ]
909         },
910         . . . # more qualifier types
911     }
912 }

```

913 7.5.8 QualifierType payload element

914 A QualifierType payload element shall be represented as a JSON property that is named with the qualifier
915 name and whose value is a JSON object that conforms to the following JSON schema:

```
916 {
917   "id": "http://schemas.dmtf.org/cimrs/is1/json/is1/qualifiertype",
918   "description": "Object type for the qualifier property of a QualifierType payload
919 element.",
920   "type": "object",
921   "properties": {
922     "links": {
923       "description": "The set of links of this payload element. For the meaning of
924 each link and for the requirement to specify this property and each link, see the
925 CIM-RS operations defined in DSP-IS0201 that use this payload element.",
926       "type": "object",
927       "properties": {
928         "self": {
929           "type": "http://schemas.dmtf.org/cimrs/is1/json/is1/link",
930           "required": true
931         },
932         "namespace": {
933           "type": "http://schemas.dmtf.org/cimrs/is1/json/is1/link",
934           "required": true
935         }
936       }
937     },
938     "type": {
939       "description": "The CIM datatype of this qualifier type. For details, see
940 7.6.5.",
941       "type": "string",
942       "enum": [ "boolean", "string", "char16", "uint8", "sint8", "uint16",
943               "sint16", "uint32", "sint32", "uint64", "sint64", "datetime",
944               "real32", "real64" ],
945       "required": true
946     },
947     "isarray": {
948       "description": "Indicates whether the CIM datatype of this qualifier type is
949 an array. For details, see 7.6.5.",
950       "type": "boolean",
951       "default": false
952     },
953     "default": {
954       "description": "The default value of this qualifier type, if any. For details
955 on the JSON type used to represent the value, see 7.6.5.",
956       "type": "any"
957     },
958     "scope": {
959       "description": "The scope of this qualifier type.",
960       "type": "array",
961       "required": true,
962       "minItems": 1,
963       "items": {
964         "description": "A CIM element type contributing to the scope of this
965 qualifier type.",
```



```

966     "type": "string",
967     "enum": [ "class", "association", "indication", "property", "reference",
968 "method", "parameter", "any" ]
969   }
970 },
971   "flavor": {
972     "description": "The flavor of this qualifier type. When no array items are
973 specified, this property may be specified or omitted.",
974     "type": "array",
975     "default": [ "tosubclass", "enableoverride" ],
976     "items": {
977       "description": "A single flavor contributing to the overall flavor of this
978 qualifier type. See DSP0004 for valid combinations.",
979       "type": "string",
980       "enum": [ "restricted", "tosubclass", "enableoverride", "disableoverride",
981 "translatable" ]
982     }
983   }
984 }
985 }

```

986 EXAMPLE:

```

987 "Abstract": {
988   "links": {
989     "self": {
990       "href": "http://acme.com:5988/cimrs/namespaces/root%2Fcimv2/qualifiers/Abstra
991 ct"},
992     "namespace": {
993       "href": "../.."}
994   },
995   "type": "boolean",
996   "default": false,
997   "scope": [ "class", "association", "indication" ],
998   "flavor": [ "Restricted" ]
999 }

```

1000 7.5.9 MethodInvocationRequest payload element

1001 A MethodInvocationRequest payload element shall be represented as a JSON property that is named
1002 with the method name and whose value is a JSON object that conforms to the following JSON schema:

```

1003 {
1004   "id": "http://schemas.dmtf.org/cimrs/is1/json/is1/methodinvocationrequest",
1005   "description": "Object type for the method property of a MethodInvocationRequest
1006 payload element.",
1007   "type": "object",
1008   "properties": {
1009     "parameters": {
1010       "description": "The set of input parameter values for the method to be
1011 invoked. When no child properties are specified, this property may be specified or
1012 omitted.",
1013       "type": "object",
1014       "additionalProperties": {

```

```

1015     "description": "Each additional JSON property shall represent a parameter
1016 value. The JSON property name shall be the parameter name. For details on the JSON
1017 type used to represent the value, see 7.6.5.",
1018     "type": "any"
1019   }
1020 }
1021 }
1022 }

```

1023 EXAMPLE:

```

1024 "SomeMethod": {
1025   "parameters": {
1026     "Parm1": "Some input data",
1027     . . . # more parameters
1028   }
1029 }

```

1030 7.5.10 MethodInvocationResponse payload element

1031 A MethodInvocationResponse payload element shall be represented as a JSON property that is named
 1032 with the method name and whose value is a JSON object that conforms to the following JSON schema:

```

1033 {
1034   "id": "http://schemas.dmtf.org/cimrs/is1/json/is1/methodinvocationresponse",
1035   "description": "Object type for the method property of a MethodInvocationResponse
1036 payload element.",
1037   "type": "object",
1038   "properties": {
1039     "returnvalue": {
1040       "description": "The return value of the method that was invoked. For details
1041 on the JSON type used to represent the value, see 7.6.5.",
1042       "type": "any",
1043       "required": true
1044     },
1045     "parameters": {
1046       "description": "The set of output parameter values for the method that was
1047 invoked. When no child properties are specified, this property may be specified or
1048 omitted.",
1049       "type": "object",
1050       "additionalProperties": {
1051         "description": "Each additional JSON property shall represent a parameter
1052 value. The JSON property name shall be the parameter name. For details on the JSON
1053 type used to represent the value, see 7.6.5.",
1054         "type": "any"
1055       }
1056     }
1057   }
1058 }

```

1059 EXAMPLE:

```

1060 "SomeMethod": {
1061   "returnvalue": 42,
1062   "parameters": {

```

```

1063     "Parm2": "Some output data",
1064     . . . # more parameters
1065   }
1066 }

```

1067 7.5.11 InstanceModificationRequest payload element

1068 An InstanceModificationRequest payload element shall be represented as a JSON object that conforms
 1069 to the following JSON schema:

```

1070 {
1071   "id": "http://schemas.dmtf.org/cimrs/is1/json/is1/instancemodificationrequest",
1072   "description": "An InstanceModificationRequest payload element.",
1073   "type": "object",
1074   "properties": {
1075     "replace": {
1076       "description": "The set of properties whose values are to be replaced. When
1077 no child properties are specified, this property may be specified or omitted.",
1078       "type": "object",
1079       "additionalProperties": {
1080         "description": "Each additional JSON property shall represent a CIM
1081 property value. The JSON property name shall be the CIM property name. For details
1082 on the JSON type used to represent the value, see 7.6.5.",
1083         "type": "any"
1084       }
1085     }
1086   }
1087 }

```

1088 This version of this document defines value replacement as the only type of property value modification.
 1089 Future support for other types of property value modifications can be provided by adding sibling (JSON)
 1090 properties to the "replace" property.

1091 EXAMPLE:

```

1092 {
1093   "replace": {
1094     "Caption": "A changed caption",
1095     . . . # more properties to be replaced
1096   }
1097 }

```

1098 7.5.12 InstanceQueryRequest payload element

1099 An InstanceQueryRequest payload element shall be represented as a JSON object that conforms to the
 1100 following JSON schema:

```

1101 {
1102   "id": "http://schemas.dmtf.org/cimrs/is1/json/is1/instancequeryrequest",
1103   "description": "An InstanceQueryRequest payload element.",
1104   "type": "object",
1105   "properties": {
1106     "querystring": {
1107       "description": "The instance query to be executed.",

```

```

1108     "type": "string",
1109     "required": true
1110   }
1111   "querylanguage": {
1112     "description": "The query language used to express the instance query.",
1113     "type": "string",
1114     "required": true
1115   }
1116 }
1117 }

```

1118 EXAMPLE:

```

1119 {
1120   "querystring": "SELECT OBJECTPATH(CIM_StorageExtent) AS Path, ElementName
1121                 FROM CIM_StorageExtent",
1122   "querylanguage": "DMTF:CQL"
1123 }

```

1124 7.5.13 ErrorResponse payload element

1125 An ErrorResponse payload element shall be represented as a JSON object that conforms to the following
1126 JSON schema:

```

1127 {
1128   "id": "http://schemas.dmtf.org/cimrs/is1/json/is1/errorresponse",
1129   "description": "An ErrorResponse payload element.",
1130   "type": "object",
1131   "properties": {
1132     "statusCode": {
1133       "description": "The CIM status code of the error response.",
1134       "type": "integer"
1135     },
1136     "statusDescription": {
1137       "description": "The CIM status description of the error response.",
1138       "type": "string"
1139     },
1140     "errors": {
1141       "description": "The set of CIM_Error instances describing the errors. When no
1142 child properties are specified, this property may be specified or omitted.",
1143       "type": "array",
1144       "items": "http://schemas.dmtf.org/cimrs/is1/json/is1/instance"
1145     }
1146   }
1147 }

```

1148 EXAMPLE:

```

1149 {
1150   "statusCode": 2,
1151   "statusDescription": "The CIM instance does not exist."
1152 }

```

1153 7.6 Representation of values used within payload elements

1154 This clause describes the JSON representation of values used within the payload elements defined in
 1155 7.5, that are not defined in 7.5. The representation of these values is described by means of JSON object
 1156 types that are identified by their "id" property.

1157 7.6.1 Method declaration

1158 A method declaration shall be represented as a JSON property that is named with the method name and
 1159 whose value is a JSON object that conforms to the following JSON schema:

```

1160 {
1161   "id": "http://schemas.dmtf.org/cimrs/is1/json/is1/methoddeclaration",
1162   "description": "A method declaration.",
1163   "type": "object",
1164   "properties": {
1165     "qualifiers": {
1166       "description": "The set of qualifier values of this method. When no child
1167 properties are specified, this property may be specified or omitted.",
1168       "type": "object",
1169       "additionalProperties": {
1170         "description": "Each additional JSON property shall represent a qualifier
1171 value. The JSON property name shall be the qualifier name. For details on the JSON
1172 type used to represent the value, see 7.6.5.",
1173         "type": "any"
1174       }
1175     },
1176     "returnvalue": {
1177       "description": "The declaration of the return value.",
1178       "type": "http://schemas.dmtf.org/cimrs/is1/json/is1/returnvaluedeclaration",
1179       "required": true
1180     },
1181     "parameters": {
1182       "description": "The set of parameters of this method declaration. When no
1183 child properties are specified, this property may be specified or omitted.",
1184       "type": "object",
1185       "additionalProperties": {
1186         "description": "Each additional JSON property shall represent a parameter
1187 declaration. The JSON property name shall be the parameter name.",
1188         "type": "http://schemas.dmtf.org/cimrs/is1/json/is1/parameterdeclaration"
1189       }
1190     }
1191   }
1192 }

```

1193 EXAMPLE:

```

1194 "SomeMethod": {
1195   "qualifiers": {
1196     "Static": true,
1197     . . . # more qualifier value specifications on the method
1198   },
1199   "returnvalue": {

```

```

1200     "type": "uint32"
1201   },
1202   "parameters": {
1203     "Parm1": {
1204       "qualifiers": {
1205         "In": true,
1206         . . . # more qualifier value specifications on the parameter
1207       },
1208       "type": "uint32"
1209     },
1210     . . . # more parameter declarations
1211   }
1212 }

```

1213 7.6.2 Return value declaration

1214 A return value declaration shall be represented as a JSON property named "returnvalue", whose value is
 1215 a JSON object that conforms to the following JSON schema:

```

1216 {
1217   "id": "http://schemas.dmtf.org/cimrs/is1/json/is1/returnvaluedeclaration",
1218   "description": "A method return value declaration.",
1219   "type": "object",
1220   "properties": {
1221     "type": {
1222       "description": "The CIM datatype of the return value. For details, see
1223 7.6.5.",
1224       "type": "string",
1225       "enum": [ "boolean", "string", "char16", "uint8", "sint8", "uint16",
1226               "sint16", "uint32", "sint32", "uint64", "sint64", "datetime",
1227               "real32", "real64", "reference", "object", "instance" ],
1228       "required": true
1229     },
1230     "referenceClass": {
1231       "description": "The name of the CIM class of any referenced instances. This
1232 property is required if the 'type' property has a value of 'reference' and is not
1233 allowed otherwise. For details, see 7.6.5.",
1234       "type": "string",
1235       "format": "cim-classname"
1236     }
1237   }
1238 }

```

1239 EXAMPLE: See 7.6.1.

1240 7.6.3 Parameter declaration

1241 A parameter declaration shall be represented as a JSON property that is named with the parameter name
 1242 and whose value is a JSON object that conforms to the following JSON schema:

```

1243 {
1244   "id": "http://schemas.dmtf.org/cimrs/is1/json/is1/parameterdeclaration",

```

```

1245     "description": "A method parameter declaration.",
1246     "type": "object",
1247     "properties": {
1248         "qualifiers": {
1249             "description": "The set of qualifier values of this parameter. When no child
1250 properties are specified, this property may be specified or omitted.",
1251             "type": "object",
1252             "additionalProperties": {
1253                 "description": "Each additional JSON property shall represent a qualifier
1254 value. The JSON property name shall be the qualifier name. For details on the JSON
1255 type used to represent the value, see 7.6.5.",
1256                 "type": "any"
1257             }
1258         },
1259         "type": {
1260             "description": "The CIM datatype of the parameter. For details, see 7.6.5.",
1261             "type": "string",
1262             "enum": [ "boolean", "string", "char16", "uint8", "sint8", "uint16",
1263                 "sint16", "uint32", "sint32", "uint64", "sint64", "datetime",
1264                 "real32", "real64", "reference", "object", "instance" ],
1265             "required": true
1266         },
1267         "isarray": {
1268             "description": "Indicates whether the CIM datatype of the parameter is an
1269 array. For details, see 7.6.5.",
1270             "type": "boolean",
1271             "default": false
1272         },
1273         "referenceClass": {
1274             "description": "The name of the CIM class of any referenced instances. This
1275 property is required if the 'type' property has a value of 'reference' and is not
1276 allowed otherwise. For details, see 7.6.5.",
1277             "type": "string",
1278             "format": "cim-classname"
1279         }
1280     }
1281 }

```

1282 EXAMPLE: See 7.6.1.

1283 7.6.4 Property declaration

1284 A property declaration shall be represented as a JSON property that is named with the property name
1285 and whose value is a JSON object that conforms to the following JSON schema:

```

1286 {
1287     "id": "http://schemas.dmtf.org/cimrs/is1/json/is1/propertydeclaration",
1288     "description": "A property declaration.",
1289     "type": "object",
1290     "properties": {
1291         "qualifiers": {
1292             "description": "The set of qualifier values of this property. When no child

```

```

1293 properties are specified, this property may be specified or omitted.",
1294     "type": "object",
1295     "additionalProperties": {
1296         "description": "Each additional JSON property shall represent a qualifier
1297 value. The JSON property name shall be the qualifier name. For details on the JSON
1298 type used to represent the value, see 7.6.5.",
1299         "type": "any"
1300     }
1301 },
1302 "type": {
1303     "description": "The CIM datatype of the property. For details, see 7.6.5.",
1304     "type": "string",
1305     "enum": [ "boolean", "string", "char16", "uint8", "sint8", "uint16",
1306             "sint16", "uint32", "sint32", "uint64", "sint64", "datetime",
1307             "real32", "real64", "reference", "object", "instance" ],
1308     "required": true
1309 },
1310 "isarray": {
1311     "description": "Indicates whether the CIM datatype of the property is an
1312 array. For details, see 7.6.5.",
1313     "type": "boolean",
1314     "default": false
1315 },
1316 "referenceClass": {
1317     "description": "The name of the CIM class of any referenced instances. This
1318 property is required if the 'type' property has a value of 'reference' and is not
1319 allowed otherwise. For details, see 7.6.5.",
1320     "type": "string",
1321     "format": "cim-classname"
1322 },
1323 "default": {
1324     "description": "The default value of this property, if any. For details on
1325 the JSON type used to represent the value, see 7.6.5.",
1326     "type": "any"
1327 }
1328 }
1329 }

```

1330 EXAMPLE:

```

1331 "SomeProperty": {
1332     "qualifiers": {
1333         "ValueMap": ["0", "1", "5", ".."],
1334         "Values": ["Unknown", "Other", "Enabled", "DMTF Reserved"],
1335         . . . # more qualifier value specifications on the property
1336     },
1337     "type": "uint32",
1338     "default": 5
1339 }

```


1340 **7.6.5 CIM value**

1341 A CIM value shall be represented as a JSON value (that is, a *JSONValue* entity as defined in [ECMA-262](#))
 1342 that conforms to the JSON data type corresponding to the CIM data type of the CIM value, as defined in
 1343 Table 1.

1344 Table 1 also defines the values of the JSON properties "type", "isArray" and "referenceClass" for each
 1345 CIM data type. These JSON properties are used to specify the data type of a CIM value, as defined in 7.5
 1346 and 0.

1347 **Table 1 – Representation of CIM values by data type**

CIM data type	JSON data type	"type" property	"isArray" property	"referenceClass" property
boolean	boolean	"boolean"	false (default)	not specified
string	string	"string"	false (default)	not specified
char16	string	"char16"	false (default)	not specified
uint8	integer	"uint8"	false (default)	not specified
sint8	integer	"sint8"	false (default)	not specified
uint16	integer	"uint16"	false (default)	not specified
sint16	integer	"sint16"	false (default)	not specified
uint32	integer	"uint32"	false (default)	not specified
sint32	integer	"sint32"	false (default)	not specified
uint64	integer	"uint64"	false (default)	not specified
sint64	integer	"sint64"	false (default)	not specified
datetime	string	"datetime"	false (default)	not specified
real32	number	"real32"	false (default)	not specified
real64	number	"real64"	false (default)	not specified
<classname> ref	string	"reference"	false (default)	name of class of referenced instances
string, with EmbeddedInstance qualifier	object, see 0	"instance"	false (default)	not specified
string, with EmbeddedObject qualifier	object, see 0	"object"	false (default)	not specified
boolean []	boolean	"boolean"	true	not specified
string []	string	"string"	true	not specified
char16 []	string	"char16"	true	not specified
uint8 []	integer	"uint8"	true	not specified
sint8 []	integer	"sint8"	true	not specified
uint16 []	integer	"uint16"	true	not specified
sint16 []	integer	"sint16"	true	not specified
uint32 []	integer	"uint32"	true	not specified
sint32 []	integer	"sint32"	true	not specified

CIM data type	JSON data type	"type" property	"isarray" property	"referenceClass" property
uint64 []	integer	"uint64"	true	not specified
sint64 []	integer	"sint64"	true	not specified
datetime []	string	"datetime"	true	not specified
real32 []	number	"real32"	true	not specified
real64 []	number	"real64"	true	not specified
<classname> ref []	string	"reference"	true	name of class of referenced instances
string [], with EmbeddedInstance qualifier	object, see 0	"instance"	true	not specified
string [], with EmbeddedObject qualifier	object, see 0	"object"	true	not specified

1348

1349 **7.6.6 Embedded instances**

1350 CIM values that are embedded instances declared with the EmbeddedInstance qualifier shall be
 1351 represented as a JSON object that conforms to the JSON schema for Instance payload (see 7.5.6), and
 1352 that contains the links as defined in Table 2:

1353

Table 2 – Links of embedded instance

Link name	Absolute / Relative	Requirement	Targeted resource
self	N/A	Prohibited	N/A
class	Absolute or relative to self link of outermost embedding instance	Mandatory	creation class of the represented embedded instance
method- invocation	N/A	Prohibited	N/A
associators	N/A	Prohibited	N/A
references	N/A	Prohibited	N/A

1354

1355 EXAMPLE:

1356 This example shows an instance of class CIM_Outer that contains a property Emb1 representing an embedded
 1357 instance of class CIM_Embedded that contains a property Prop1.

1358

1359

1360

1361

1362

1363

1364

```
{
  "links": {
    "self": {
      "href": "http://acme.com:5988/cimrs/namespaces/root%2Fcimv2/classes/CIM_Outer/instances/acme1"},
    "class": {
      "href": "../.."}
  }
}
```

```

1365     "methodinvocation": {
1366         "href": "methodinvocation"},
1367     "associators": {
1368         "href": "associators"},
1369     "references": {
1370         "href": "references"}
1371 },
1372 "class": "CIM_Outer",
1373 "properties": {
1374     "InstanceID": "acme1",
1375     "Emb1": { # The property value is an embedded instance
1376         "links": {
1377             "class": {
1378                 "href": "../../../CIM_Outer"},
1379             },
1380             "class": "CIM_Embedded",
1381             "properties": {
1382                 "Prop1": "value1"
1383             }
1384         }
1385     }
1386 }

```

1387 7.6.7 Embedded objects

1388 CIM values that are embedded objects declared with the EmbeddedObject qualifier shall be represented
 1389 dependent on whether they represent instances or classes, as follows:

- 1390 • If the CIM value represents an instance, it shall be represented as a JSON object that conforms
 1391 to the JSON schema for an Instance payload element (see 7.5.6) that does not contain the
 1392 properties "links" or "class".
- 1393 • If the CIM value represents a class, it shall be represented as a JSON object that conforms to
 1394 the JSON schema for a Class payload element (see 7.5.4) that does not contain the property
 1395 "links".

1396 EXAMPLE:

1397 This example shows an instance of class CIM_Outer that contains a property Emb1 representing an embedded
 1398 object that is an instance of class CIM_Embedded that contains a property Prop1.

```

1399 {
1400     "links": {
1401         "self": {
1402             "href": "http://acme.com:5988/cimrs/namespaces/root%2Fcimv2/classes/CIM_Outer
1403 /instances/acme1"},
1404         "class": {
1405             "href": "../../"},
1406         "methodinvocation": {
1407             "href": "methodinvocation"},
1408         "associators": {
1409             "href": "associators"},
1410         "references": {
1411             "href": "references"}

```

```
1412 },
1413 "class": "CIM_Outer",
1414 "properties": {
1415     "InstanceID": "acme1",
1416     "Emb1": { # The property value is an embedded object that is an instance
1417         "properties": {
1418             "Prop1": "value1"
1419         }
1420     }
1421 }
1422 }
```

1423
1424
1425
1426
1427

ANNEX A (informative)

Change log

Version	Date	Description
1.0.0a	2010-10-11	Released as Work in Progress (of an Informational Specification)
1.0.0	2011-04-15	Released as DMTF Draft Informational Specification

Bibliography

1428

1429 This bibliography contains a list of non-normative references for this document.

1430 DMTF DSP0202, *CIM Query Language Specification 1.0*,

1431 http://www.dmtf.org/standards/published_documents/DSP0202_1.0.pdf

1432 IANA MIME Media Types,

1433 <http://www.iana.org/assignments/media-types/>

1434 IETF RFC2045, *Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies*,

1435 <http://tools.ietf.org/html/rfc2045>

1437 IETF RFC2046, *Multipurpose Internet Mail Extensions (MIME) Part Two: Media Types*,

1438 <http://tools.ietf.org/html/rfc2046>

1439 *JSON Schema Proposal* (a predecessor of [RFC-json-structure](#)),

1440 <http://json-schema.org/>

1441 J. Holzer, *RESTful Web Services and JSON for WBEM Operations*, Master thesis, University of Applied Sciences, Konstanz, Germany, June 2009,

1442 <http://mond.htwg-konstanz.de/Abschlussarbeiten/Details.aspx?id=1120>