



1

2

3

Cloud Infrastructure Management Interface (CIMI) Model and REST Interface over HTTP

4

An Interface for Managing Cloud Infrastructure

Information for Work-in-Progress version:

IMPORTANT: This specification is not a standard. It does not necessarily reflect the views of the DMTF or all of its members. Because this document is a Work in Progress, this specification may still change, perhaps profoundly. This document is available for public review and comment until the stated expiration date.

It expires on: 2012-05-15

Provide any comments through the DMTF Feedback Portal:

<http://www.dmtf.org/standards/feedback>

5

6

Version: 0.0.46a

7

Status: Work In Progress - not a DMTF Standard

8

Publication Date: 2011-11-15

9

Document Number: DSP0263

10

11 Copyright Notice

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

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

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27 withdrawn or modified after publication, and shall be indemnified and held harmless by any party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

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

33 Abstract

34 This document is a deliverable from the DMTF Cloud Management Working Group. It defines a logical
35 model for the management of resources within the Infrastructure as a Service domain. This model was
36 developed to address the use cases outlined in the “Scoping Framework for Cloud Management Models
37 and Protocol Requirements” document.

38 Acknowledgments

39 TBD

CONTENTS

40		
41	Cloud Infrastructure Management Interface (CIMI) Model and REST Interface over HTTP	1
42	CONTENTS.....	4
43	FIGURES	7
44	1 Scope	8
45	1.1 Document Structure	8
46	1.2 Typographical Conventions.....	8
47	2 References	8
48	3 Terms and Definitions	10
49	3.1 Authentication.....	10
50	3.2 Authorization	10
51	3.3 Cloud Service Consumer	10
52	3.4 Cloud Service Provider	10
53	3.5 Configuration	11
54	3.6 Message Confidentiality	11
55	3.7 Message Integrity	11
56	3.8 Template.....	11
57	4 REST/HTTP Protocol	11
58	4.1 Protocol Definition	11
59	4.1.1 Protocol Security	12
60	4.1.2 XML Namespaces	12
61	4.1.3 URI Space	12
62	4.1.4 Media Types	12
63	4.1.5 Request Headers	12
64	4.1.6 Request Parameters	13
65	4.1.7 Response Headers.....	14
66	4.1.8 HTTP Status Codes.....	14
67	4.1.9 Serialization of References	16
68	4.1.10 Serialization of Arrays.....	16
69	4.2 Protocol Resource Operations	16
70	4.2.1 Operational Principles	17
71	4.2.2 Common CRUD (Create Read Update and Delete) Operations.....	19
72	5 Model.....	23
73	5.1 Identifiers.....	24
74	5.2 Attribute "Properties"	24
75	5.3 Data Types and Values	24
76	5.4 Relationship Semantics.....	25
77	5.5 Alternative Model Formats	25

78	5.6	Entities.....	26
79	5.6.1	Common Attributes.....	26
80	5.7	Entity Metadata.....	26
81	5.7.1	Attribute Types.....	29
82	5.7.2	Examples.....	30
83	5.8	Cloud Entry Point.....	31
84	5.8.1	Operations.....	34
85	5.9	System Entities and Relationships.....	34
86	5.9.1	System Template.....	35
87	5.9.2	System Template Collection.....	36
88	5.9.3	System.....	37
89	5.9.4	System Collection.....	38
90	5.10	Machine Entities and Relationships.....	43
91	5.10.1	Machine Template.....	43
92	5.10.2	Machine Template Collection.....	48
93	5.10.3	Machine Configuration.....	49
94	5.10.4	Machine Configuration Collection.....	51
95	5.10.5	Machine Image.....	52
96	5.10.6	Machine Image Collection.....	53
97	5.10.7	Machine.....	54
98	5.10.8	Machine Collection.....	61
99	5.10.9	Machine Admin.....	64
100	5.10.10	Machine Admin Collection.....	65
101	5.11	Volume Entities and Relationships.....	66
102	5.11.1	Volume Template.....	67
103	5.11.2	Volume Template Collection.....	68
104	5.11.3	Volume Configuration.....	68
105	5.11.4	Volume Configuration Collection.....	70
106	5.11.5	Volume Image.....	71
107	5.11.6	Volume Image Collection.....	72
108	5.11.7	Volume.....	73
109	5.11.8	Volume Collection.....	75
110	5.12	Network Entities and Relationships.....	77
111	5.12.1	Network Template.....	78
112	5.12.2	Network Template Collection.....	80
113	5.12.3	Network Configuration.....	80
114	5.12.4	Network Configuration Collection.....	82
115	5.12.5	Network.....	83
116	5.12.6	Network Collection.....	85

117	5.12.7 VSP (Virtual Switch Port) Template	87
118	5.12.8 VSP (Virtual Switch Port) Template Collection	88
119	5.12.9 VSP (Virtual Switch Port) Configuration	89
120	5.12.10 VSP (Virtual Switch Port) Configuration Collection	90
121	5.12.11 VSP (Virtual Switch Port).....	91
122	5.12.12 VSP (Virtual Switch Port) Collection.....	93
123	5.12.13 Routing Group	95
124	5.12.14 Routing Group Collection	96
125	5.13 Monitoring Entities and Relationships	98
126	5.13.1 Job.....	99
127	5.13.2 Job Collection	101
128	5.13.3 Meter Template	102
129	5.13.4 Meter Template Collection	104
130	5.13.5 Meter	105
131	5.13.6 Meter Collection.....	108
132	5.13.7 Event Log	110
133	5.13.8 Event Log Collection	112
134	5.13.9 Event	113
135	5.13.10 Event Collection.....	114
136	6 Scenarios	115
137	6.1 Initial Scenario.....	115
138	6.1.1 Create and deploy a Machine using a Provider created Machine Template.....	116
139	6.1.2 Create a Machine by passing a Machine Template by value	116
140	6.1.3 Create a Machine using a Consumer created Machine Template	117
141	6.1.4 Create a Machine Template by specifying individual components.....	117
142	6.1.5 Create a Machine Template from a template file.....	117
143	6.1.6 Control Machine State (CMWG065).....	117
144	6.2 Machine Image Scenarios.....	118
145	6.2.1 Create new Machine Image from an image file.....	118
146	6.2.2 Create new Machine Image from Machine instance	118
147	6.3 System Scenario	118
148	6.3.1 List System Templates (CMWG010)	118
149	6.3.2 Create System Template	118
150	6.3.3 Create and Deploy a System to a Site Using a System Template	
151	(CMWG017/CMWG035)	118
152	6.3.4 Get Monitoring Information (CMWG066)	119
153	6.3.5 Control System State (CMWG065)	119
154	6.3.6 Remove System from a Site (CMWG051)	119
155	7 Security.....	119

156 7.1 API Level Security 119

157 7.1.1 Authentication 119

158 7.1.2 Message Integrity 119

159 7.1.3 Message Confidentiality 120

160 7.1.4 Authorization 120

161 7.1.5 Multi-Tenancy 120

162 7.2 Resource Level Credentials 120

163

FIGURES

165 Figure 1 - System Entities 34

166 Figure 2 - Machine Entities 43

167 Figure 3 - Volume Entities 66

168 Figure 4 - Network Entities 78

169 Figure 5 - Monitoring Entities 99

170

171

172 **1 Scope**

173 Quoting the “Architecture for Managing Clouds White Paper”:

174 Any programmatic API has an underlying resource model, whether implicit or explicit. In the IT
175 management domain, the practice has long been to make resource models explicit and clearly
176 separated from the protocols used to manipulate model elements.

177 This document describes an abstract, service offering model by defining a set of logical entities that are
178 shared between consumers and service providers.

179 **1.1 Document Structure**

180 This document defines a model and an HTTP/REST-based protocol.

181 The core REST pattern is defined first and, after each entity is defined, any REST-specific information for
182 that entity will be specified.

183 **1.2 Typographical Conventions**

184 This specification uses the following syntax to define the serialization of resources:

- 185 • Values in *italics* indicate data types instead of literal values.
- 186 • Characters are appended to items to indicate cardinality:
 - 187 ○ "?" (0 or 1)
 - 188 ○ "*" (0 or more)
 - 189 ○ "+" (1 or more)
- 190 • Ellipses (i.e., "...") indicate points of extensibility. Note that the lack of an ellipses does not mean
191 no extensibility point exists, rather it is just not explicitly called out - usually for the sake of
192 brevity.

193 **2 References**

194 The following referenced documents are indispensable for the application of this document. For dated
195 references, only the edition cited applies. For undated references, the latest edition of the referenced
196 document (including any amendments) applies:

197 **IEC 80000-13:2008**, International Organization for Standardization, Geneva, Switzerland, *Quantities and*
198 *units – Part 13: Information science and technology*, April 2008,
199 http://www.iso.org/iso/catalogue_detail?csnumber=31898

200 **IETF RFC 2045**, N. Freed et al, *Multipurpose Internet Mail Extensions (MIME) Part One: Format of*
201 *Internet Message Bodies*, November 1996, <http://www.ietf.org/rfc/rfc2045.txt>

202 **IETF RFC 2616**, R. Fielding et al, *Hypertext Transfer Protocol -- HTTP/1.1*, June 1999,
203 <http://www.ietf.org/rfc/rfc2616.txt>

204 **IETF RFC 2246**, T. Dierks and C. Allen, *The TLS Protocol Version 1.0*, January 1999,
205 <http://www.ietf.org/rfc/rfc2246.txt>

- 206 **IETF RFC 3986**, T. Berners-Lee et al, *Uniform Resource Identifiers (URI): Generic Syntax*, August 1998,
207 <http://www.ietf.org/rfc/rfc3986.txt>
- 208 **IETF RFC 4288**, N. Freed and J. Klensin, *Media Type Specifications and Registration Procedures*,
209 December 2005, <http://www.ietf.org/rfc/rfc4288.txt>
- 210 **IETF RFC 4346**, T. Dierks and E. Rescorla, *The Transport Layer Security (TLS) Protocol Version 1.1*,
211 April 2006, <http://www.ietf.org/rfc/rfc4346.txt>
- 212 **IETF RFC 4627**, D. Crockford, *The application/json Media Type for JavaScript Object Notation (JSON)*,
213 July 2006, <http://www.ietf.org/rfc/rfc4627.txt>
- 214 **IETF RFC 5246**, T. Dierks and E. Rescorla, *The Transport Layer Security (TLS) Protocol Version 1.1*,
215 <http://www.ietf.org/rfc/rfc5246.txt>
- 216 **ISO 8601:2004**, International Organization for Standardization, Geneva, Switzerland, *Data elements and*
217 *interchange formats -- Information interchange - - Representation of dates and times*, March 2008,
218 http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=40874
- 219 **ISO/IEC Directives, Part 2**, *Rules for the structure and drafting of International Standards*,
220 <http://isotc.iso.org/livelink/livelink.exe?func=ll&objId=4230456&objAction=browse&sort=subtype>
- 221 **ITU-T X.509**, Telecommunication Standardization Sector of ITU, *Information technology - Open Systems*
222 *Interconnection - The Directory: Public-key and attribute certificate frameworks*, November 2008,
223 <http://www.itu.int/rec/T-REC-X.509-200811-I>
- 224 **NIST Special Publication 800-57**, Elaine Barker et al, *Recommendation for Key Management – Part 1:*
225 *General (Revised)*, March 2007, [http://csrc.nist.gov/publications/nistpubs/800-57/sp800-57-Part1-](http://csrc.nist.gov/publications/nistpubs/800-57/sp800-57-Part1-revised2_Mar08-2007.pdf)
226 [revised2_Mar08-2007.pdf](http://csrc.nist.gov/publications/nistpubs/800-57/sp800-57-Part1-revised2_Mar08-2007.pdf)
- 227 **NIST Special Publication 800-131A**, Elaine Barker and Allen Roginsky, *Transitions: Recommendation*
228 *for Transitioning the Use of Cryptographic Algorithms and Key Lengths*, January 2011,
229 <http://csrc.nist.gov/publications/nistpubs/800-131A/sp800-131A.pdf>
- 230 **Representational State Transfer**, Roy Fielding, Doctoral dissertation, University of California,
231 *Architectural Styles and the Design of Network-based Software Architectures (Chapter 5)*, 2000,
232 http://www.ics.uci.edu/~fielding/pubs/dissertation/rest_arch_style.htm
- 233 **XMLSchema - Part 1**, World Wide Web Consortium (W3C) Recommendation, H. Thompson, et al.,
234 Editors, *XML Schema Part 1: Structures Second Edition*, 28 October 2004,
235 <http://www.w3.org/TR/xmlschema-1/>
- 236 **XMLSchema - Part 2**, World Wide Web Consortium (W3C) Recommendation, P. Biron, A. Malhotra,
237 Editors, *XML Schema Part 2: Datatypes (Second Edition)*, 28 October 2004,
238 <http://www.w3.org/TR/xmlschema-2/>
- 239 **DMTF DSP-0243**, Distributed Management Task Force, Inc., *Open Virtualization Format Specification*
240 *1.1.0*, http://www.dmtf.org/sites/default/files/standards/documents/DSP0243_1.1.0.pdf
- 241 **DMTF DSP-0259**, Distributed Management Task Force, Inc., *Cloud Infrastructure Management Interface -*
242 *CIM Model (CIMI-CIM) 0.0.1*, <http://members.dmtf.org/apps/org/workgroup/cmwg/download.php/yyyy>
- 243 **DMTF DSP-XXXX**, Distributed Management Task Force, Inc., *Cloud Infrastructure Management Interface*
244 *- RelaxNG Model (CIMI-RNG) 0.0.1*,
245 <http://members.dmtf.org/apps/org/workgroup/cmwg/download.php/zzz>
- 246 *Note: the CIMI-RNG document is not yet available*

247 **3 Terms and Definitions**

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

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

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

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

261 **3.1 Authentication**

262 The process of verifying a claim, made by a subject, that it should be allowed to act on behalf of a given
263 principal (person, service, etc.). Typical authentication mechanisms involved the use of
264 username/password combination or public/private key pairs.

265 **3.2 Authorization**

266 (also known as **Access Control**) The process of verifying that an authenticated principal (person, service,
267 etc.) has permission to perform certain operations (e.g. read, update) on specific resources.

268 **3.3 Cloud Service Consumer**

269 A category of actors that includes the Consumer Business Manager (who approves business and
270 financial expenditures for consumed services, accounts for used service instances, establishes business
271 relationships; sets up accounts, budget, and terms; etc.), the Consumer Service Administrator (who
272 requests service instances and changes to service instances, purchase services within the business
273 relationship; create Service Users (including policies), allocate resources, such as compute and storage,
274 generate reports (usage), etc.), and Service Users (who uses service instances provided by a Cloud
275 Service Provider). The term "**Consumer**" is used when the indicated action or activity could involve one or
276 more of the above actors. In cases where the distinction between the actors in this category is relevant,
277 the more detailed term will be used.

278 **3.4 Cloud Service Provider**

279 A category of actors that includes the Service Operations Manager (who manages the technical
280 infrastructure required for providing cloud services, monitors and measures performance and utilization
281 against SLAs, provides reports from monitoring and measurement, etc.), Service Business Manager (who
282 offers all types of services developed by cloud service developers, accounts for services potentially
283 offered by service providers themselves and services offered on behalf of cloud service developers,
284 establishes a portfolio of business relationships, and sets up accounts and terms for Consumers, etc.),
285 and Service Transition Manager (who enables a customer to use the cloud service, including
286 "onboarding", integration, and process adoption, defines and creates service offerings based on
287 Templates and Configurations that can be used by Consumers and are populated into the catalog, etc.).
288 The term "Provider" is used when the indicated action or activity could involve one or more of the above
289 actors. In cases were the distinction between the actors in the category is relevant, the more detailed
290 term will be used.

291 **3.5 Configuration**

292 A Configuration is a set of metadata, the values of which serve as the parameters of a discrete
293 conformation of a specific type of virtual resource. For example, a Machine Configuration may define a
294 Machine with the equivalent of a 2.66 GHz processor, 4 GB of memory, and 320 GB of local disk storage.

295 **3.6 Message Confidentiality**

296 A quality of a message which prevents anyone but the intended receiver(s) from viewing its contents.

297 **3.7 Message Integrity**

298 A quality of a message which allows a receiver of that message to determine if the contents of the
299 message have been altered since its creation.

300 **3.8 Template**

301 A Template is the entity that represents the set of metadata and instructions used to instantiate resources
302 (e.g. a Machine Template is used to create Machines). Templates may aggregate other metadata entities
303 such as other Templates, Configurations and Images. For example, a Machine Template refers to a
304 Machine Configuration and a Machine Image.

305 How a specific protocol mapping, or implementation, chooses to supply Templates as inputs to the
306 instantiation process may vary. However, some common patterns should be considered:

- 307 1. By reference - allow Consumers to reference a Template (that exists as an entity in the Provider) as
308 part of the instantiation operation.
- 309 2. By value - allow Consumers to dynamically provide the Template information as part of the
310 instantiation operation.
- 311 3. Reference with overrides - allow Consumers to reference a Template (that exists as an entity in the
312 Provider) and provide additional values that override the attributes of that Template as part of the
313 instantiation operation.

314 **4 REST/HTTP Protocol**

315 **4.1 Protocol Definition**

316 All operations are based on the HyperText Transfer Protocol, version 1.1 [[RFC2616](#)]. Each request is
317 sent using an HTTP verb such as PUT, GET, DELETE, HEAD or POST and includes a message body in
318 either JSON or XML format. Each response uses a standard HTTP status code, overloaded with
319 semantics by the context of the particular request that was made. Each entity in the model has a MIME
320 standard ContentType that further contextualizes the operation requests and responses.

321 The entities in the model are identified by URIs and each entity's representation MUST contain a "uri"
322 property that acts as a "self pointer". This URI SHALL be unique within the context of the Provider's
323 implementation. Dereferencing (via an HTTP GET) the URI of an entity will yield a representation of the
324 entity containing attributes and links to associated entities. To begin operations, a client must know the
325 URI to the main entry point of a Cloud Provider - also known as the "Cloud Entry Point" entity. All other
326 entities within the environment shall then be discoverable via the iterative following of links to associated
327 resource within each resource retrieved.

328 4.1.1 Protocol Security

329 Cloud Providers SHALL support secure HTTP connections using TLS. Cloud Providers MAY support non-
330 secure HTTP connections. TLS 1.0, which shall be implemented, is specified in [\[RFC2246\]](#), and the TLS
331 1.1 and TLS 1.2 should be implemented as specified in [\[RFC4346\]](#) and [\[RFC5246\]](#), respectively.

332 To ensure a minimum level of security and interoperability between implementations, all CIMI clients and
333 servers shall support the TLS_DHE_DSS_WITH_3DES_EDE_CBC_SHA cipher suite (hexadecimal value
334 {0x0013}), which is also the mandatory cipher suite for TLS 1.0 (see [\[RFC2246\]](#) Section 9, Mandatory
335 Cipher Suites), as well as the TLS_RSA_WITH_AES_128_CBC_SHA cipher suite (hexadecimal value
336 {0x002F}) shall be implemented, which is the mandatory cipher suite for both TLS 1.1 and TLS 1.2.
337 Finally, the TLS_RSA_WITH_AES_128_CBC_SHA256 cipher suite (hexadecimal value {0x003C}) should
338 be included with all recommended TLS 1.2 implementations to meet the transition to a security strength of
339 112 bits (guidance is provided in NIST Special Publication 800-57 [\[NIST 800-57\]](#) and NIST Special
340 Publication 800-131A [\[NIST 800-131A\]](#)). Implementers are free to include additional cipher suites, but
341 must prefer the mandatory ones in negotiation.

342 4.1.2 XML Namespaces

343 The following table lists the XML namespaces that are used in this specification. The choice of any
344 namespace prefix is arbitrary and not semantically significant.

Prefix	XML Namespaces	Specification
cimi	http://www.dmtf.org/cimi	This specification
xs	http://www.w3.org/2001/XMLSchema	XML Schema XMLSchema - Part 2

345 4.1.3 URI Space

346 While URIs returned by providers are to be treated as opaque by consumers, and consumers MUST NOT
347 make assumptions about the layout of the URIs or the structures of the URIs of the resources, consumer
348 may augment URIs with any well-defined query parameters which are supported by the provider as
349 defined in section 4.1.6. Providers shall not use the CIMI-defined query parameter reserved namespace
350 (i.e. names starting with "CIMI").

351 4.1.4 Media Types

352 In this specification, resource representations and request bodies are encoded in either JSON, as
353 specified in [\[RFC4627\]](#) or in XML.

354 Each type of resource has its own media-type, which matches the pattern `application/.Xxxxx+json`, where
355 "Xxxxx" represents the portion of the identifier unique to a particular representation format for each
356 resource (entity in the model). The identifier MUST either be a DMTF standard identifier as defined in this
357 specification and as registered in accordance to [\[RFC4288\]](#), or it must be a vendor specific identifier that
358 is globally unique (vendor extension).

359 The server implementation shall provide representations of all resources available in both JSON and XML
360 as specified herein. The client implementation may thus use either JSON or XML to communicate with
361 any server implementation.

362 4.1.5 Request Headers

363 This specification uses general-header, request-header, and entity-header headers as defined in HTTP
364 1.1 [\[RFC2616\]](#) in request messages to provide metadata about the message. Applications using
365 messages defined in this specification shall use headers consistent with HTTP 1.1.

366 In addition to headers defined in HTTP 1.1, request messages may include a header defined by this
 367 specification to indicate the set of allowable versions of the CIMI API that server shall use to process the
 368 message.

```
369 X-CIMI-Specification-Version = "X-CIMI-Specification-Version" ":" api-version
370 api-version = "1.0"
```

371 The header allows for a list of *api-version* values to be specified (separated by commas). When more
 372 than one value is present the server shall choose one of those versions of the specification to process the
 373 message. Clients including more than one value are indicating that any of the specified values are
 374 acceptable.

375 If the server is unable to support any of the specified versions then it shall generate a fault and not
 376 process the message. Absence of this header indicates that the server may choose any version of this
 377 specification to process the message.

378 4.1.6 Request Parameters

379 The client can use request parameters in requests to formulate the following

380 **Editors Note: These are example URL parameters for requests that apply across all resource types. This**
 381 **will need to be revisited down the road when we decide what support for this we need.**

382

Table: Request Parameters

Format	Description	Example
?CIMISelect=attr1, attr2,...	Comma separated attribute names indicate that a subset of the resource is being identified. If an attribute is not part of the resource, then it would be ignored. If none of the attributes is part of the resource, then the resource would be returned in its complete form The CIMISelect query parameter name may appear more than once in URI which is semantically equivalent to all of the attribute names appearing as values of a single CIMISelect query parameter. For example: ?CIMISelect=name&CIMISelect=state is equivalent to: ?CIMISelect=name,state	MyMachine?CIMISelect=name,description,state Would return only "name", "description", "state" attributes of the Assembly132.
?[collapse]	This would collapse all the Collection attributes by not returning the individual elements	Only the Collection's uri, name, and total would be returned for all the attributes that are of Collection type
?<attr1>:[collapse]	This would return only attr1, and if attr1 is a Collection, it would be collapsed. If attr1 is not a Collection, it would be ignored	?servers:[collapse] Would return Collection's uri, name, and total only
?[verbose]	This would show all the fields of all the attributes, recursively, including the	For example, /assembly123?[verbose] would

	collections	return the expanded list of all the components, including all the attributes
--	-------------	--

383 The client must URL encode the request parameters.

384 **4.1.7 Response Headers**

385 This specification uses general-header, response-header, and entity-header headers as defined in HTTP
 386 1.1 [RFC2616] in response messages to provide metadata about the message. Applications using
 387 messages defined in this specification shall use headers consistent with HTTP 1.1.

388 In addition to headers defined in HTTP 1.1, response messages shall include a header defined by this
 389 specification to indicate the version of the CIMI API that the server used to process the message.

```
390 X-CIMI-Specification-Version = "X-CIMI-Specification-Version" ":" api-version
391 api-version = "1.0"
```

392 Additionally, if the server supports the Job entity then response messages shall include a header defined
 393 by this specification to indicate the URI for the job created to process the associated request message.

```
394 X-CIMI-Job-URI = "X-CIMI-Job-URI" ":" string
```

395 **4.1.8 HTTP Status Codes**

396 Server implementations will return standard HTTP response codes as described in the following table,
 397 under the conditions listed in the description.

398 **Editors Note:** These are changes from the basic HTTP semantics that are overloaded for the whole
 399 specification. We will remove any status codes that are standard HTTP without overloading.

400 **Table: HTTP Status Codes**

HTTP Status	Description
100 Continue	The client SHOULD continue with its request. This interim response is used to inform the client that the initial part of the request has been received and has not yet been rejected by the platform. The client SHOULD continue by sending the remainder of the request or, if the request has already been completed, ignore this response.
200 OK	The request was successfully completed. If this request created a new resource that is addressable with a URI, and a response body is returned containing a representation of the new resource, a 200 status will be returned with a Location header containing the canonical URI for the newly created resource
201 Created	A request that created a new resource was completed, and no response body containing a representation of the new resource is being returned. A Location header containing the canonical URI for the newly created resource will be returned. Per the HTTP/1.1 specification: <i>The origin server MUST create the resource before returning the 201 status code. If the action cannot be carried out immediately, the server SHOULD respond with 202 (Accepted) response instead.</i>

202 Accepted	<p>The request has been accepted for processing, but the processing has not been completed. Per the HTTP/1.1 specification, the returned entity (if any) SHOULD include an indication of the request's current status. A Location header containing the canonical URI for the not-yet completed resource would be returned along with the Status attribute indicating its progress.</p> <p>If a service implementing this specification supports the Job entity then it SHOULD return the representation of a Job entity in the HTTP body of the response and shall include a 'X-CIMI-Job-URI' HTTP header indicating the URI of the Job entity itself.</p> <p>Per the HTTP/1.1 specification:</p> <p><i>The entity returned with this response SHOULD include an indication of the request's current status and either a pointer to a status monitor or some estimate of when the user can expect the request to be fulfilled</i></p>
400 Bad Request	The request could not be processed because it contains missing or invalid information (such as validation error on an input field, a missing required value, and so on)
401 Unauthorized	The authentication credentials (TBD) included with this request are missing or invalid
403 Forbidden	The server recognized your credentials, but you do not possess authorization to perform this request
404 Not Found	The request specified a URI of a resource that does not exist
405 Method Not Allowed	The HTTP verb specified in the request (DELETE, GET, HEAD, POST, PUT) is not supported for this request URI. This is used in the create/update/delete of MachineConfiguration and MachineImages to indicate that the provider only supports a fixed set of immutable entities.
406 Not Acceptable	The resource identified by this request is not capable of generating a representation corresponding to one of the media types in the Accept header of the request
409 Conflict	A creation or update request could not be completed, because it would cause a conflict in the current state of the resources supported by the platform. This is used in MachineTemplate create/update to indicate that the MachineConfiguration cannot support the given MachineImage, for example.
410 Gone	The requested resource is no longer available at the server and no forwarding address is known. This condition is expected to be considered permanent. Clients with link editing capabilities SHOULD delete references to the Request-URI after user approval. If the server does not know, or has no facility to determine, whether or not the condition is permanent, the status code 404 (Not Found) SHOULD be used instead. This response is cacheable unless indicated otherwise
412 Precondition Failed	The precondition given in one or more of the request-header fields evaluated to false when it was tested on the server. This response code allows the client to place preconditions on the current resource meta-information (header field data) and thus prevent the requested method from being applied to a resource other than the one intended

500 Internal Server Error	The server encountered an unexpected condition which prevented it from fulfilling the request
501 Not Implemented	The server does not (currently) support the functionality required to fulfill the request
503 Service Unavailable	The server is currently unable to handle the request due to temporary overloading or maintenance of the server

401 4.1.9 Serialization of References

402 References, as indicated by the type 'ref' in the model, are defined to be URIs in the REST/HTTP protocol
403 mapping. In the JSON serialization they will appear as type "string" and in the XML serialization they will
404 appear as type "xs:anyURI".

405 References in both JSON and XML have an extensibility point that allows for additional information (such
406 as the target resource to be included "by value") if supported. For example, a reference to a Volume in
407 this specification will appear like this in JSON:

```
408     "volume": { "href": string }
```

409 and this in XML:

```
410     <volume href="xs:anyURI"/>
```

411 For convenience the JSON and XML, as shown above, excludes the implicit extensibility points that would
412 allow for the attributes of the target Volume to be included if desired. So, technically the above should be
413 written as:

```
414     "volume": { "href": string, ... }
```

415 and this in XML:

```
416     <volume href="xs:anyURI"> xs:any* </volume>
```

417 however, for brevity they are excluded.

418 4.1.10 Serialization of Arrays

419 Within this specification, arrays in JSON are serialized with a wrapper property. When serializing arrays,
420 conformant implementations SHALL NOT include empty arrays (i.e. arrays that contain no child
421 properties) in the JSON serialization. For example, an array of references to a list of Volumes attached to
422 a Machine is serialized as:

```
423     "volumes" : [  
424         { "volume": { "href": string },  
425           "attachmentPoint": string,  
426           "protocol": string } +  
427     ], ?
```

428 Notice that the child of the "volumes" property is defined with a "+", meaning at least one child is required.
429 This is done to ensure that the JSON serialization is minimized and only includes the wrapping "volumes"
430 element if, and only if, there are volumes.

431 4.2 Protocol Resource Operations

432 This section defines the set of common REST/HTTP operations that a Cloud Provider might expose. At its
433 core there are four basic CRUD (Create, Read, Update and Delete) operations. The manner in which
434 these are used is consistent across all resources within the model; therefore, their use is defined once

435 and is to be applied consistently. Some resources support specialized operations that do not fit well into
436 a CRUD style of operation and those will all follow a similar high-level pattern but each operation is
437 allowed to have slight variations to accommodate its specific needs. The specifics of these special
438 operations are detailed within the section that defines the resource.

439 When appropriate some of the resource representations will include "operation" properties. These either
440 provide URI references that can be used to perform operations on the resource, or they are URI
441 references to other resources that are related to the current resource. Providers shall only include
442 "operation" properties when the specified operation or related resource is accessible to the current client
443 for that particular resource. This means that based on many factors (e.g. authorization rights of the
444 clients, current state of the resource, etc.) a different set of "operation" properties might be returned on
445 each serialization of the resource.

446 **4.2.1 Operational Principles**

447 **4.2.1.1 Resource Navigation**

448 The retrieval of the representation of a Resource using (GET <ResourceURI>) shall return the attributes
449 of the resource; these attributes might include a set of references to related resources. In that case, it is
450 possible to obtain every related resource by repeatedly applying the GET method on the retrieved
451 references.

452 Example:

453 If a resourceX contains an attribute "attrA" of string type and an attribute "attrB", where the latter
454 references resourceY, the operation:

```
455 GET <ResourceURI_X> HTTP/1.1  
456 Host: ...  
457 Accept: application/CIMI-...  
458 X-CIMI-Specification-Version: 1.0
```

459 returns a message containing the following:

460 **JSON serialization:**

```
461 { "attrA": : "hello",  
462   "attrB": { "href": "http://example.com/uriB" }  
463 }
```

464 **XML serialization:**

```
465 <Resource_X xmlns="http://www.dmtf.org/cimi">  
466   <attrA> hello </attrA>  
467   <attrB href="http://example.com/uriB" />  
468 </Resource_X>
```

469 Following the attrB uri ("uriB"), the operation:

```
470 GET <uriB> HTTP/1.1  
471 Host: ...  
472 Accept: application/CIMI-...  
473 X-CIMI-Specification-Version: 1.0
```

474 returns a message containing the following:

475 **JSON serialization:**

```
476 { "attrY": : "bye" }
```

477 **XML serialization:**

```
478 <Resource_Y xmlns="http://www.dmtf.org/cimi">
479   <attrY> bye </attrY>
480 </Resource_Y>
```

481 **Notes:**

- 482 1) It is possible that the retrived reference obtained with the GET <ResourceURI> operation
 483 does not directly refer to a related resorce, but to a list of homogeneous related
 484 resources. In such case, the result of the GET operation will be a list of URIs, and thus
 485 the result of a further GET on one of such URIs will return the related resource.
- 486 2) this makes it possible to navigate the CIMI resource hierachy with just the knowledge of
 487 the root URI of the Cloud Provider.

488 **4.2.1.2 Operations on a Resource**

489 When it is possible to execute specific operations on the resource, then the response to the GET method
 490 on the resource uri shall contain information to perform such operations, in particular:

- 491 1. The operation name (typology) such as add, delete, edit, start, stop are described using the rel
 492 attribute in the "operation" element.
- 493 2. The URI to perform the above mentioned operation.

494 It is possible to understand that such a field isn't a link to a related resource but instead is an operation by
 495 the presence of the attribute "operations" in JSON or the element "operation" in XML.

496 The operation shall be performed by invoking the REST specific function on the specified URI.

497 **Example:**

498 If for the resource X an operation is provided to edit that resource, the Get <ResourceURI> response
 499 message will contain the following:

500 **JSON serialization:**

```
501 { "operations": [
502   { "rel": "edit", "href": "editURI" }
503 ]
504 }
```

505 **XML serialization:**

```
506 <Resource_X xmlns="http://www.dmtf.org/cimi">
507   <operation rel="edit" href="editURI"/>
508 </Resource_X>
```

509 In this example the operation will be performed with the HTTP PUT on editURI, as follows:

```
510 PUT <editURI> HTTP/1.1
511 Host: ...
512 Accept: application/CIMI-...
513 Content-Type: application/CIMI-...
514 X-CIMI-Specification-Version: 1.0
515
516 <serialization of request to update the resource>
```

517 4.2.2 Common CRUD (Create Read Update and Delete) Operations

518 Each of the resources supported by this protocol will adhere to the interaction patterns defined in the
519 following sections. Section 5 then defines resource specific information such as the serialization of each
520 resource's properties and which specific actions are supported.

521 4.2.2.1 Creating a new Resource

522 To create a new instance of a resource type, an HTTP POST request is sent to a designated "addURI" for
523 that resource type. In many cases, the Collection resource that maintains, or groups, all instances of that
524 resource type will contain an "addLink" property which contains the "addURI" that is to be used.

525 The request will be of the following form:

```
526 POST <addURI> HTTP/1.1
527 Host: ...
528 Accept: application/CIMI-...
529 Content-Type: application/CIMI-...
530 X-CIMI-Specification-Version: 1.0
531
532 <serialization of request to create a new resource>
```

533 The following provides additional constraints on the request message:

534 **X-CIMI-Specification-Version**

535 This optional HTTP header specifies the list of versions of this specification that the server shall
536 choose from to process this message.

537 The response will be of the following form:

```
538 HTTP/1.1 201 Created
539 Location: ...
540 Content-Type: application/CIMI-...
541 X-CIMI-Specification-Version: 1.0
542
543 <serialization of new resource>
```

544 The following provides additional constraints on the response message:

545 **X-CIMI-Specification-Version**

546 This REQUIRED HTTP header specifies the version of this specification that was used to process this
547 message.

548 The HTTP response will also include a status code, as described in the following table:

HTTP Status	Description
201 Created	The new resource was created
202 Accepted	The resource is in the process of being created. Investigate Job to determine the current status of the operation.
400 Bad Request	Invalid parameter or field names in the request.
401 Unauthenticated	Incorrect or missing authentication credentials.
403 Unauthorized	Client lacks the proper authorization to perform this request.

549 4.2.2.2 Reading a Resource

550 To retrieve the representation of resource, an HTTP GET request is sent to the URI of that resource.

551 The request will be of the following form:

```
552 GET <ResourceURI> HTTP/1.1
553 Host: ...
554 Accept: application/CIMI-...
555 X-CIMI-Specification-Version: 1.0
```

556 The following provides additional constraints on the request message:

557 **X-CIMI-Specification-Version**

558 This optional HTTP header specifies the list of versions of this specification that the server shall
559 choose from to process this message.

560 The response will be of the following form:

```
561 HTTP/1.1 200 OK
562 Content-Type: application/CIMI-...
563 X-CIMI-Specification-Version: 1.0
564
565 <serialization of resource>
```

566 The following provides additional constraints on the response message:

567 **X-CIMI-Specification-Version**

568 This REQUIRED HTTP header specifies the version of this specification that was used to process this
569 message.

570 The HTTP response will also include a status code, as described in the following table:

HTTP Status	Description
401 Unauthenticated	Incorrect or missing authentication credentials.
403 Unauthorized	Client lacks the proper authorization to perform this request.

571 **4.2.2.3 Updating a Resource**

572 To update the representation of a resource, an HTTP PUT request is sent to a designated "editURI" for
573 that resource type. In many cases, this "editURI" will be the same as the URI of resource itself - retrieving
574 the resource representation MUST include an "editLink" property, which contains the "editURI" that is to
575 be used, if the requester is allowed to modify the resource.

576 While processing a PUT request if the server detects that an attempt is being made to update a read-
577 only, or immutable, attribute then it SHALL silently ignore that attribute update request and SHALL NOT
578 generate an error. This applies to resource partial updates as well.

579 Due to potential conflicts that might occur due to multiple concurrent updates, Consumers should use the
580 partial update mechanism, defined in 4.2.2.3.1, to avoid mistakenly updating attributes with out-of-date
581 data.

582 The request will be of the following form:

```
583 PUT <editURI> HTTP/1.1
584 Host: ...
585 Accept: application/CIMI-...
586 Content-Type: application/CIMI-...
587 X-CIMI-Specification-Version: 1.0
588
589 <serialization of request to update a resource>
```

590 The following provides additional constraints on the request message:

591 **X-CIMI-Specification-Version**

592 This optional HTTP header specifies the list of versions of this specification that the server shall
593 choose from to process this message.

594 The response will be of the following form:

```
595 HTTP/1.1 200 OK
596 Content-Type: application/CIMI-...
597 X-CIMI-Specification-Version: 1.0
598
599 <serialization of updated resource>
```

600 The following provides additional constraints on the response message:

601 **X-CIMI-Specification-Version**

602 This REQUIRED HTTP header specifies the version of this specification that was used to process this
603 message.

604 The HTTP response message body shall include the updated version of the resource representation.

605 The HTTP response will also include a status code, as described in the following table:

HTTP Status	Description
202 Accepted	The resource is in the process of being created. Investigate Job to determine the current status of the operation.
400 Bad Request	Invalid parameter or field names in the request.
401 Unauthenticated	Incorrect or missing authentication credentials.
403 Unauthorized	Client lacks the proper authorization to perform this request.

606 **4.2.2.3.1 Partial Updates to a Resource**

607 To update only certain top-level attributes of a resource a consumer MAY do so by including only the
608 changes attributes in the representation of the resource within the HTTP request body. When this is done
609 the URI to the resource SHALL include the attributes to be modified as a comma separated list of query
610 parameters - in other words the URI will be of the form:

```
611 http://example.com/resource?CIMISelect=attribute1,attribute2,...
```

612 Only the attributes listed in the URI's query parameters will be modified; attributes not listed in the URI are
613 not directly modified by the request. Note that this does not preclude the modification of one attribute
614 causing side-effects that result in the modification of an attribute not listed in the query parameters.

615 Any attribute listed in the URI but not included within the HTTP request body are reset to a resource
616 specific value (e.g. removed).

617 From an HTTP perspective, the updated subsetted resource is a distinct one; the semantics of a normal
618 HTTP PUT are adhered to - it is a complete replacement update of the specified resource. From the
619 Consumer's perspective, the partial update is interpreted and executed by the Cloud Service Provider,
620 and some part of the resource is/are changed.

621 For example, the following request will update just the name and description attributes of a Machine:

```
622 PUT /machines/myMachine?CIMISelect=name,description HTTP/1.1
623 Host: ...
624 Accept: application/CIMI-Machine
```

```

625 Content-Type: application/CIMI-Machine
626 X-CIMI-Specification-Version: 1.0
627
628 <Machine>
629   <name>My New Machine</name>
630 </Machine>

```

631 In this example, the "name" attribute is set to "My New Machine" and the "description" attribute is erased.

632 4.2.2.4 Deleting a Resource

633 To delete a resource, an HTTP DELETE request is sent to a designated "deleteURI" for that resource
634 type. In many cases, this "deleteURI" will be the same as the URI of resource itself - retrieving the
635 resource representation MUST include a "deleteLink" property, which contains the "deleteURI" that is to
636 be used, if the requester is allowed to delete the resource.

637 The request will be of the following form:

```

638 DELETE <deleteURI> HTTP/1.1
639 Host: ...
640 X-CIMI-Specification-Version: 1.0

```

641 The following provides additional constraints on the request message:

642 **X-CIMI-Specification-Version**

643 This optional HTTP header specifies the list of versions of this specification that the server shall
644 choose from to process this message.

645 The response will be of the following form:

```

646 HTTP/1.1 200 OK
647 X-CIMI-Specification-Version: 1.0

```

648 The following provides additional constraints on the response message:

649 **X-CIMI-Specification-Version**

650 This REQUIRED HTTP header specifies the version of this specification that was used to process this
651 message.

652 The HTTP response will also include a status code, as described in the following table:

HTTP Status	Description
202 Accepted	The resource is in the process of being created. Investigate Job to determine the current status of the operation.
400 Bad Request	Invalid parameter or field names in the request.
401 Unauthenticated	Incorrect or missing authentication credentials.
403 Unauthorized	Client lacks the proper authorization to perform this request.

653 4.2.2.5 Other Operations

654 While some modifications to the resources in the model can be done via a simple update (PUT) operation
655 to the resource's "editURI", sometimes a more complex set of actions need to be taken. In these cases,
656 the operations will be modeled as HTTP POSTs to the operation specific "Link" property/URI of the
657 resource.

658 For each of the resources that define additional operations, a description of the HTTP request and
659 response bodies will be provided. However, the general HTTP interaction will be as described below.

660 The request will be of the following form:

```
661 POST <operationLinkURI> HTTP/1.1  
662 Host: ...  
663 Accept: application/CIMI-...  
664 Content-Type: application/CIMI-...  
665 X-CIMI-Specification-Version: 1.0  
666  
667 <serialization of request to perform some action>
```

668 The following provides additional constraints on the request message:

669 **X-CIMI-Specification-Version**

670 This optional HTTP header specifies the list of versions of this specification that the server shall
671 choose from to process this message.

672 The form of the response will vary depending on the operation and will be defined by the operation itself.

673 Note that the definition of the "Create" operation (see section 4.2.2.1) follows this same pattern - it is just
674 called out for ease of reference.

675 **4.2.2.6 Synchronous Operations**

676 If a Provider supports the Job entity then each incoming PUT, DELETE, POST request SHALL result in a
677 Job entity being created and a reference to that Job entity SHALL be returned back to the client via the X-
678 CIMI-Job-URI HTTP Header in the HTTP response message:

```
679 X-CIMI-Job-URI: <uri-to-Job>
```

680 In this case, the requested operation shall be complete and the Job URI SHALL point to a completed Job.
681 If the Job is not complete, then the server SHALL return a 202 and follow the instructions for
682 Asynchronous Operations.

683 **4.2.2.7 Asynchronous Operations**

684 In some cases, an operation requested by the client may take an undetermined amount of time to
685 complete. For example, creating a new Machine or starting an existing Machine, may take a relatively
686 long time to complete. In these cases, it is not practical to complete these operations within a reasonable
687 HTTP request timeout interval. For these cases, the Provider SHALL return an HTTP "202 Accepted"
688 response code.

689 As with synchronous operations, if a Provider supports the Job entity then it SHALL create a Job entity for
690 the incoming request and return a reference to that Job entity back to the client via the X-CIMI-Job-URI
691 HTTP Header in the HTTP response message. Additionally, in the case of a "202 Accepted" response
692 code and a Job URI being returned, the Provider MAY also return a representation of the Job entity in the
693 body of the HTTP response message. If the request did not include the Job MIME type in the HTTP
694 Accept header, then the encoding style (json vs xml) of the response SHOULD match the encoding style
695 of the request message.

696 Note that the decision as to whether any particular operation will be synchronous or asynchronous is at
697 the server's discretion.

698 **5 Model**

699 This model assumes that a business relationship has already been established between the Cloud
700 Consumer and the Cloud Provider. This relationship may include financial terms, creating separately

701 administered clouds that the consuming organization is paying for, and the establishment of
702 authentication credentials to access the administrative entry point for each cloud. This scope of this
703 model is one separately administered cloud.

704 5.1 Identifiers

705 All identifiers (e.g. entity names, attributes, operations, parameter names) defined by this specification, or
706 defined via an extension, shall adhere to the following:

- 707 • Identifier names shall be treated as case sensitive
- 708 • Identifier names shall only use the following set of characters:
 - 709 ○ Upper case ASCII (U+0041 through U+005A)
 - 710 ○ Lower case ASCII (U+0061 through U+007A)
 - 711 ○ Digits (U+0030 through U+0039)
 - 712 ○ Underscore (U+005F)

713 5.2 Attribute "Properties"

714 Each attribute of the entities in the CIMI model is augmented by a set of "Properties" that further qualify
715 the attribute being defined. The following describes the possible "Properties".

716 **Optionality:**

717 The entity definition tables contain an indicator as to whether the specified attribute (and its
718 corresponding feature) is required to be supported by Cloud providers. Possible values are:

- 719 • Optional - indicates that the specified attribute/feature may be supported
- 720 • Mandatory - indicates that the specified attribute/feature shall be supported
- 721 • Conditional - indicates the specified attribute/feature is mandatory if the condition is satisfied. The
722 condition will be described in the description cell. If the condition is not met, the attribute is
723 optional.

724 **Mutability:**

725 Attributes are either "Immutable" (their values are fixed for the lifetime of the entity), or "Mutable" (their
726 values may change over the lifetime of the entity). Unless otherwise noted, all attributes are mutable.

727 **Writability:**

728 Mutable attributes are either "Writeable" (their value may be changed by the Consumer) or "Read-Only"
729 (their value may only be changed by the Provider). Unless otherwise noted, all mutable attributes are
730 writeable.

731 5.3 Data Types and Values

732 The following describes the data types and values that are using within the model definition tables.

733 **URIs:**

734 Note that the format and syntax of the attributes of type "URI" is defined by RFC 3986 [[RFC3986](#)] with the
735 following, additional constraints: Relative URIs MUST start with a "/", otherwise the URI is assumed to be

736 absolute and no URI processing (to determine the full path) will be performed. Relative URIs are
737 interpreted as being relative to the root URI of the CloudEntryPoint.

738 **Units:**

739 Some of the entities defined by this specification have attributes that describe an amount of something
740 that belongs to, or is associated with that entity. For example, the `Machine` entity has a `memory` attribute
741 which describes “the size of the memory allocated to this machine”. This specification adopts the
742 convention of representing such attributes via a duple consisting of a `quantity` (represented as an
743 integer) and `units` (represented as a string). The allowable values for `units` are listed in the following
744 table. Their meaning is defined in IEC 80000-13:2008 [[IEC 80000-13:2008](#)]. Their numerical equivalents
745 are provided here for convenience:

String	Numerical Value	String	Numerical Value
kilobyte	10 ³	kibibyte	2 ¹⁰
megabyte	10 ⁶	mebibyte	2 ²⁰
gigabyte	10 ⁹	gibibyte	2 ³⁰
terabyte	10 ¹²	tebibyte	2 ⁴⁰
petabyte	10 ¹⁵	pebibyte	2 ⁵⁰
exabyte	10 ¹⁸	exbibyte	2 ⁶⁰
zettabyte	10 ²¹	zebibyte	2 ⁷⁰
yottabyte	10 ²⁴	yobibyte	2 ⁸⁰

746 **5.4 Relationship Semantics**

747 A reference between two entity instances has the semantics of a simple “association”. In particular,
748 unless specified otherwise, (a) the same referred instance can be referred by other entity instances, i.e.
749 be “shared”, and (b) the referred entity instance is not affected when deleting the referring entity instance
750 (i.e. the Delete operation is a “shallow delete” by default).

751 The embedding of a sub-entity inside another entity, has the semantics of a “composition” (or whole-part
752 relationship in UML). In particular, unless specified otherwise: (a) an embedded sub-entity cannot be
753 shared by several entity instances, (b) when deleting an embedding entity instance, the embedded sub-
754 entity instances are also deleted.

755 **5.5 Alternative Model Formats**

756 Since it is expected that this specification will be implemented using a variety of technologies, as a
757 convenience, the definition of the model elements are provided in alternative formats that are easily
758 consumable by technology-specific tooling.

759 This model is available in a CIM/MOF format [CIMI-CIM] as well as a RelaxNG format [CIMI-RNG].

760 **Note: the CIMI-RNG document is not yet available.**

761 In the event of inconsistencies between the various formats, the normative text within this specification
762 takes precedence over the XML Schemas and alternative formats, which in turn take precedence over
763 examples.

764 **5.6 Entities**

765 The following sections detail the attributes of the entities defined by the CIMI model.

766 **5.6.1 Common Attributes**

767 The entities described by this document share the following, common attributes.

Attribute	Type	Description
self	ref	The unique self-reference to this entity; assigned upon entity creation. This attribute value shall be unique in the provider's cloud. Properties: Mandatory / Immutable
name	string	The human readable name of this entity; assigned by the creator as a part of the entity creation input. Properties: Optional / Mutable
description	string	The human readable description of this entity; assigned by the creator as a part of the entity creation input. Properties: Optional / Mutable
created	DateTimeUTC	The timestamp when this entity was created. The format should be unambiguous, and the value is immutable . Properties: Optional / Immutable
properties	map	A list of name/value pairs, some of which may control one or more aspects this entity. Properties may also serve as an extension point, allowing consumers and providers to record configuration and control information for features and capabilities beyond those defined by this specification. The same "name" SHALL NOT be used more than once within a "properties" attribute. Individual properties may be either Mutable or Immutable and, if mutable, Writeable or Read-Only , depending upon the nature of the property and the underlying cloud implementation. Properties: Optional / Mutable

768 **5.7 Entity Metadata**

769 Implementations of this specification SHOULD allow for Consumers to discover the metadata associated
770 with each supported entity. Doing so allows for the discovery of Provider defined constraints on the CIMI
771 defined attributes as well as discovery of any new extension attributes that the Provider may have
772 defined. The mechanism by which this metadata is made available will be protocol specific.

773 Each entity's metadata will contain the following pieces of information:

Name	EntityMetadata																			
Type URI	http://www.dmtf.org/cimi/EntityMetadata																			
Attribute	Type	Description																		
self	ref	The unique self-reference to this entity; assigned upon entity creation. This attribute value is immutable , and shall be unique in the provider's cloud. Properties: Mandatory / Mutable																		
typeURI	URI	A unique URI associated with, and denoting, this entity type. Properties: Mandatory / Mutable																		
name	string	The name of the entity type. Properties: Mandatory / Mutable																		
attributes	attribute[]	A set of Provider defined metadata that can be used by clients to discover any metadata associated with each attribute, as well we the set of extension attributes. Each attribute will contain the following nested data: <table border="1" data-bbox="578 926 1430 1845"> <tr> <td>Name</td> <td colspan="2">attribute</td> </tr> <tr> <td>Data</td> <td>Type</td> <td>Description</td> </tr> <tr> <td>name</td> <td>string</td> <td>The name of the attribute. Properties: Mandatory / Mutable</td> </tr> <tr> <td>namespace</td> <td>URI</td> <td>The namespace in which this attribute is defined. It is recommended that a dereference of this URI returns information about the attribute. This SHALL not be present when describing a CIMI defined attribute, but SHALL be present when describing a non-CIMI defined attribute. Properties: Mandatory / Mutable</td> </tr> <tr> <td>type</td> <td>string</td> <td>The data type of the attribute. This SHALL not be present when describing a CIMI defined attribute, but SHALL be present when describing a non-CIMI defined attribute. Properties: Mandatory / Mutable</td> </tr> <tr> <td>required</td> <td>boolean</td> <td>Indicates whether this entity requires this attribute to be present. When absent the implied value is "false". Properties: Mandatory / Mutable</td> </tr> </table>	Name	attribute		Data	Type	Description	name	string	The name of the attribute. Properties: Mandatory / Mutable	namespace	URI	The namespace in which this attribute is defined. It is recommended that a dereference of this URI returns information about the attribute. This SHALL not be present when describing a CIMI defined attribute, but SHALL be present when describing a non-CIMI defined attribute. Properties: Mandatory / Mutable	type	string	The data type of the attribute. This SHALL not be present when describing a CIMI defined attribute, but SHALL be present when describing a non-CIMI defined attribute. Properties: Mandatory / Mutable	required	boolean	Indicates whether this entity requires this attribute to be present. When absent the implied value is "false". Properties: Mandatory / Mutable
Name	attribute																			
Data	Type	Description																		
name	string	The name of the attribute. Properties: Mandatory / Mutable																		
namespace	URI	The namespace in which this attribute is defined. It is recommended that a dereference of this URI returns information about the attribute. This SHALL not be present when describing a CIMI defined attribute, but SHALL be present when describing a non-CIMI defined attribute. Properties: Mandatory / Mutable																		
type	string	The data type of the attribute. This SHALL not be present when describing a CIMI defined attribute, but SHALL be present when describing a non-CIMI defined attribute. Properties: Mandatory / Mutable																		
required	boolean	Indicates whether this entity requires this attribute to be present. When absent the implied value is "false". Properties: Mandatory / Mutable																		

		<table border="1"> <tr> <td>constraints</td> <td>abstract</td> <td>Type specific data that describes the constraints of this attribute. When absent there are no constraints. Properties: Optional / Mutable</td> </tr> </table> <p>Properties: Optional / Mutable</p>	constraints	abstract	Type specific data that describes the constraints of this attribute. When absent there are no constraints. Properties: Optional / Mutable																					
constraints	abstract	Type specific data that describes the constraints of this attribute. When absent there are no constraints. Properties: Optional / Mutable																								
operations	operation[]	<p>A set of Provider defined operations that can be used by clients to act on the entity.</p> <p>Each operation will contain the following nested data:</p> <table border="1"> <thead> <tr> <th>Name</th> <th>Type</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>operation</td> <td></td> <td></td> </tr> <tr> <td>name</td> <td>string</td> <td>The name of the operation. Properties: Mandatory / Mutable</td> </tr> <tr> <td>uri</td> <td>URI</td> <td>A URI that uniquely identifies the operation at a global level. Properties: Mandatory / Mutable</td> </tr> <tr> <td>description</td> <td>string</td> <td>The human readable description of the semantic of the operation. Properties: Optional / Mutable</td> </tr> <tr> <td>method</td> <td>string</td> <td>The method to use to perform the operation. Properties: Mandatory / Mutable</td> </tr> <tr> <td>inputMessage</td> <td>string</td> <td>The body mimeType of the request message, it may depend on the model format chosen by the provider. Properties: Optional / Mutable</td> </tr> <tr> <td>outputMessage</td> <td>string</td> <td>The body mimeType of the response message, it may depend on the model format chosen by the provider. Properties: Optional / Mutable</td> </tr> </tbody> </table> <p>Properties: Optional / Mutable</p>	Name	Type	Description	operation			name	string	The name of the operation. Properties: Mandatory / Mutable	uri	URI	A URI that uniquely identifies the operation at a global level. Properties: Mandatory / Mutable	description	string	The human readable description of the semantic of the operation. Properties: Optional / Mutable	method	string	The method to use to perform the operation. Properties: Mandatory / Mutable	inputMessage	string	The body mimeType of the request message, it may depend on the model format chosen by the provider. Properties: Optional / Mutable	outputMessage	string	The body mimeType of the response message, it may depend on the model format chosen by the provider. Properties: Optional / Mutable
Name	Type	Description																								
operation																										
name	string	The name of the operation. Properties: Mandatory / Mutable																								
uri	URI	A URI that uniquely identifies the operation at a global level. Properties: Mandatory / Mutable																								
description	string	The human readable description of the semantic of the operation. Properties: Optional / Mutable																								
method	string	The method to use to perform the operation. Properties: Mandatory / Mutable																								
inputMessage	string	The body mimeType of the request message, it may depend on the model format chosen by the provider. Properties: Optional / Mutable																								
outputMessage	string	The body mimeType of the response message, it may depend on the model format chosen by the provider. Properties: Optional / Mutable																								

774 The following describes the serialization of the entity in both JSON and XML:

775 **JSON media type:** application/CIMI-EntityMetadata+json

776 **JSON serialization:**

```
777 { "self": string,
778   "typeURI": URI,
779   "name": string,
780   "attributes" : [
```

```

781     { "name": "string",
782       "namespace": "string", ?
783       "type": "string", ?
784       "required": boolean, ?
785       ...constraints...? } *
786   ], ?
787   "operations" : [
788     { "name": "string"
789       "description": "string", ?
790       "method": "string",
791       "inputMessage": "string", ?
792       "outputMessage": "string" ? }, *
793   ] ?
794   ...
795 }

```

796 **XML media type:** application/CIMI-EntityMetadata+xml

797 **XML serialization:**

```

798 <EntityMetadata xmlns="http://www.dmtf.org/cimi">
799   <self> xs:anyURI </self>
800   <name> xs:string </name>
801   <typeURI> xs:anyURI </typeURI>
802   <attribute name="xs:string" namespace="xs:anyURI"? type="xs:string"
803     required="xs:boolean"? >
804     ...constraints...?
805   </attribute> *
806   <operation name="xs:string" description="xs:string"? method="xs:string"
807     inputMessage="xs:string"? outputMessage="xs:string"? /> *
808   <xs:any>*
809 </EntityMetadata>

```

810 Additional metadata about the entity or attributes MAY be included by the Provider.

811 5.7.1 Attribute Types

812 The following describes the constraint metadata corresponding to the attribute's "type" value.

813 type="string"

814 The JSON SHALL be of the form:

```
815 "values": [ "string", + ] ?
```

816 The XML SHALL be of the form:

```
817 <value> xs:string </value> *
```

818 type="integer"

819 The JSON SHALL be of the form:

```
820 "values": [ integer, + ], ?
821 "ranges": [ { "low": integer, "high": integer }, + ] ?
```

822 The XML SHALL be of the form:

```
823 <value> xs:integer </value> *
824 <range low="xs:integer" high="xs:integer"/> *
```

825 The total value space of an 'integer' attribute is the accumulation of all values and ranges.

826 **type="boolean"**

827 The JSON SHALL be of the form:

```
828 "value": boolean ?
```

829 The XML SHALL be of the form:

```
830 <value> xs:boolean </value> ?
```

831 Only one 'value' is permitted which indicates whether the attribute is required to be either 'true' or 'false'.

832 5.7.2 Examples

833 The following shows a sample metadata document for a VolumeConfiguration entity in XML that as been
834 extended with a "Location" string attribute:

```
835 <EntityMetadata xmlns="http://www.dmtf.org/cimi">
836   <self> http://example.org/types/VC </self>
837   <typeURI> http://www.dmtf.org/cimi/VolumeConfiguration </typeURI>
838   <name> VolumeConfiguration </name>
839   <attribute name="Location" namespace="http://example.org/" type="string"/>
840 </EntityMetadata>
```

841 The following shows the same VolumeConfiguration but the "Location" attribute is restricted to a set of
842 values and is required:

```
843 <EntityMetadata xmlns="http://www.dmtf.org/cimi">
844   <self> http://example.org/types/VC </self>
845   <typeURI> http://www.dmtf.org/cimi/VolumeConfiguration </typeURI>
846   <name> VolumeConfiguration </name>
847   <attribute name="Location" namespace="http://example.org/" type="string"
848     required="true">
849     <value> NYC </value>
850     <value> LAX </value>
851   </attribute>
852 </EntityMetadata>
```

853 The following shows the same VolumeConfiguration serialized in JSON:

```
854 { "self": "http://example.org/types/VC",
855   "typeURI": "http://www.dmtf.org/cimi/VolumeConfiguration",
856   "name": "VolumeConfiguration",
857   "attributes": [
858     { "name": "Location",
859       "namespace": "http://example.org",
860       "type": "string",
861       "required": true,
862       "values": [ "NYC", "LAX" ]
863     }
864   ]
865 }
```

866 The following shows a Volume serialized in JSON wich provides an operation of data compression. In this
867 specific example the method returned (POST) is for the CIMI REST protocol; should another protocol be
868 implemented (e.g SOAP) the "method" will be different :

```
869 { "uri": "http://example.org/types/V",
870   "typeURI": "http://www.dmtf.org/cimi/Volume",
871   "name": "Volume",
872   "operations": [
873     {
874       "name": "compress",
```

875
876
877
878
879
880

```

      "uri": "http://example.org/cimi/action/compress"
      "description": "Compress the data stored in the volume",
      "method": "POST"
    }
  ]
}

```

881 5.8 Cloud Entry Point

882 The Cloud Entry Point represents the entry point into the cloud defined by the CIMI Model. The Cloud
883 Entry Point implements a catalog of entities such as Systems, System Templates, Machines, Machine
884 Templates, etc. that can be queried and browsed by the Consumer.

Name	CloudEntryPoint	
Type URI	http://www.dmf.org/cimi/CloudEntryPoint	
Attribute	Type	Description
systemTemplates	ref	A reference to the System Template Collection of this CloudEntry Point. Properties: Optional / Mutable
systems	ref	A reference to the System Collection of this Cloud Entry Point. Properties: Optional / Mutable
machineTemplates	ref	A reference to the Machine Template Collection of this Cloud Entry Point. Properties: Optional / Mutable
machineConfigs	ref	A reference to the Machine Configuration Collection of this Cloud Entry Point. Properties: Optional / Mutable
machineImages	ref	A reference to the Machine Image Collection of this Cloud Entry Point. Properties: Optional / Mutable
machineAdmins	ref	A reference to the Machine Admin Collection of this Cloud Entry Point. Properties: Optional / Mutable
machines	ref	A reference to the Machine Collection of this Cloud Entry Point. Properties: Optional / Mutable
volumeTemplates	ref	A reference to the Volume Template Collection of this Cloud Entry Point. Properties: Optional / Mutable
volumeConfigs	ref	A reference to the Volume Configuration Collection of this Cloud Entry Point. Properties: Optional / Mutable
volumeImages	ref	A reference to the Volume Image Collection of this Cloud Entry Point. Properties: Optional / Mutable

volumes	ref	A reference to the Volume Collection of this Cloud Entry Point. Properties: Optional / Mutable
networkTemplates	ref	A reference to the Network Template Collection of this Cloud Entry Point. Properties: Optional / Mutable
networkConfigs	ref	A reference to the Network Configuration Collection of this Cloud Entry Point. Properties: Optional / Mutable
networks	ref	A reference to the Network Collection of this Cloud Entry Point. Properties: Optional / Mutable
vspTemplates	ref	A reference to the VSP Template Collection of this Cloud Entry Point. Properties: Optional / Mutable
vspConfigs	ref	A reference to the VSP Configuration Collection of this Cloud Entry Point. Properties: Optional / Mutable
vsps	ref	A reference to the VSP Collection of this Cloud Entry Point. Properties: Optional / Mutable
routingGroups	ref	A reference to the Routing Group Collection of this Cloud Entry Point. Properties: Optional / Mutable
meterTemplates	ref	A reference to the Meter Template Collection of this Cloud Entry Point. Properties: Optional / Mutable
meters	ref	A reference to the Meter Collection of this Cloud Entry Point. Properties: Optional / Mutable
eventLogs	ref	A reference to the Event Log Collection of this Cloud Entry Point. Properties: Optional / Mutable
events	ref	A reference to the Event Collection of this Cloud Entry Point Properties: Optional / Mutable
jobTime	long	This value is Provider specific and is the minimum amount of time a Job will be retained by the system after the completion of the Job. Properties: Optional / Mutable
entityMetadata	ref[]	List of references to EntityMetadata entities supported by the Provider. If an entity does not have any metadata then it will not appear in this list - e.g. it has no constraints beyond what the CIMI specification defines nor does it have any extension attributes. Properties: Optional / Mutable

885 The following describes the serialization of the entity in both JSON and XML:

886 **JSON media type:** application/CIMI-CloudEntryPoint+json

887 **JSON serialization:**

```

888 { "self": string,
889   "name": string, ?
890   "description": string, ?
891   "created": string, ?
892   "properties": { "name": string, + }, ?
893   "systemTemplates": { "href": string }, ?
894   "systems": { "href": string }, ?
895   "machineTemplates": { "href": string }, ?
896   "machineConfigs": { "href": string }, ?
897   "machineImages": { "href": string }, ?
898   "machineAdmins": { "href": string }, ?
899   "machines": { "href": string }, ?
900   "volumeTemplates": { "href": string }, ?
901   "volumeConfigs": { "href": string }, ?
902   "volumeImages": { "href": string }, ?
903   "volumes": { "href": string }, ?
904   "networkTemplates": { "href": string }, ?
905   "networkConfigs": { "href": string }, ?
906   "networks": { "href": string }, ?
907   "vspTemplates": { "href": string }, ?
908   "vspConfigs": { "href": string }, ?
909   "vsps": { "href": string }, ?
910   "routingGroups": { "href": string }, ?
911   "meterTemplates": { "href": string }, ?
912   "meters": { "href": string }, ?
913   "eventLogs": { "href": string }, ?
914   "events": { "href": string }, ?
915   "job_time": number, ?
916   "entityMetadata": [
917     { "href": string }, +
918   ], ?
919   "operations": [
920     { "rel": "edit", "href": string }, ?
921   ] ?
922   ...
923 }
```

924 **XML media type:** application/CIMI-CloudEntryPoint+xml

925 **XML serialization:**

```

926 <CloudEntryPoint xmlns="http://www.dmtf.org/cimi">
927   <self> xs:anyURI </self>
928   <name> xs:string </name> ?
929   <description> xs:string </description> ?
930   <created> xs:dateTime </created>
931   <property name="xs:string"> xs:string </property> *
932   <systemTemplates href="xs:anyURI"/> ?
933   <systems href="xs:anyURI"/> ?
934   <machineTemplates href="xs:anyURI"/> ?
935   <machineConfigs href="xs:anyURI"/> ?
936   <machineImages href="xs:anyURI"/> ?
937   <machineAdmins href="xs:anyURI"/> ?
938   <machines href="xs:anyURI"/> ?
939   <volumeTemplates href="xs:anyURI"/> ?
940   <volumeConfigs href="xs:anyURI"/> ?
941   <volumeImages href="xs:anyURI"/> ?
```

```

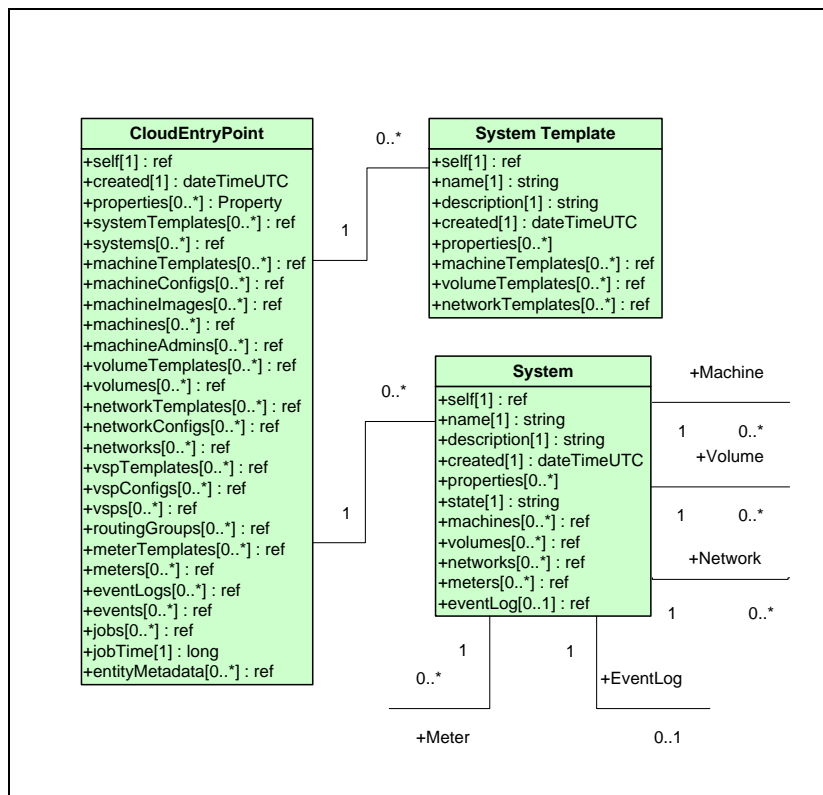
942 <volumes href="xs:anyURI"/> ?
943 <networkTemplates href="xs:anyURI"/> ?
944 <networkConfigs href="xs:anyURI"/> ?
945 <networks href="xs:anyURI"/> ?
946 <vspTemplates href="xs:anyURI"/> ?
947 <vspConfigs href="xs:anyURI"/> ?
948 <vsps href="xs:anyURI"/> ?
949 <routingGroups href="xs:anyURI"/> ?
950 <meterTemplates href="xs:anyURI"/> ?
951 <meters href="xs:anyURI"/> ?
952 <eventLogs href="xs:anyURI"/> ?
953 <events href="xs:anyURI"/> ?
954 <job time> xs:integer </job time>
955 <entityMetadata href="xs:anyURI"/> *
956 <operation rel="edit" href="xs:anyURI"/> ?
957 <xs:any>*
958 </CloudEntryPoint>
    
```

959 **5.8.1 Operations**

960 This entity supports the Read and Update operations.

961 **5.9 System Entities and Relationships**

962 The following diagram illustrates the entities involved in constructing a System and their relationships.
 963 Although this drawing is in the style of an Entity Relationship diagram, the use of UML is neither rigorous
 964 nor normative.



965
 966 **Figure 1 - System Entities**

967 **5.9.1 System Template**

968 The System Template contains configuration values for realizing a System. A System Template can be
 969 used to create multiple Systems.

Name	SystemTemplate	
Type URI	http://www.dmtf.org/cimi/SystemTemplate	
Attribute	Type	Description
volumeTemplates	ref[]	List of volume templates referenced in this System Template. Items in this list are used to create Volumes as part of the System creation process. Properties: Optional / Mutable
machineTemplates	ref[]	List of Machine Templates referenced in this System Template. Items in this list are used to create Machines as part of the System creation process. Properties: Optional / Mutable
networkTemplates	ref[]	List of NetworkTemplates referenced in this System Template. Items in this list are used to create Networks as part of the System creation process. Properties: Optional / Mutable

970 **JSON media type:** application/CIMI-SystemTemplate+json

971 **JSON serialization:**

```

972 { "self": string,
973   "name": string, ?
974   "description": string, ?
975   "created": string, ?
976   "properties": { "name": string, + }, ?
977   "volumeTemplates": [
978     { "href": string }, +
979   ], ?
980   "machineTemplates": [
981     { "href": string }, +
982   ], ?
983   "networkTemplates": [
984     { "href": string }, +
985   ], ?
986   "operations": [
987     { "rel": "edit", "href": string }, ?
988     { "rel": "delete", "href": string } ?
989   ] ?
990   ...
991 }
```

992 **XML media type:** application/CIMI-SystemTemplate+xml

993 **XML serialization:**

```

994 <SystemTemplate xmlns="http://www.dmtf.org/cimi">
995   <self> xs:anyURI </self>
996   <name> xs:string </name> ?
997   <description> xs:string </description> ?
998   <created> xs:string </created>
999   <property name="xs:string"> xs:string </property> *
1000   <volumeTemplate href="xs:anyURI"/> *
```

```

1001 <machineTemplate href="xs:anyURI"/> *
1002 <networkTemplate href="xs:anyURI"/> *
1003 <operation rel="edit" href="xs:anyURI"/> ?
1004 <operation rel="delete" href="xs:anyURI"/> ?
1005 <xs:any>*
1006 </SystemTemplate>
    
```

1007 **5.9.1.1 Operations**

1008 This entity supports the Read, Update and Delete operations. Create is supported via the System
 1009 Template Collection entity.

1010 **5.9.2 System Template Collection**

1011 A System Template Collection entity represents the collection of System Template entities within a
 1012 Provider. This resource can be used to locate and create System Templates.

Name	SystemTemplateCollection	
Type URI	http://www.dmtf.org/cimi/SystemTemplateCollection	
Attribute	Type	Description
systemTemplates	ref[]	An array of references to the set of System Templates in the Provider. Properties: Optional / Mutable

1013 The following describes the serialization of the entity in both JSON and XML:

1014 **JSON media type:** application/CIMI-SystemTemplateCollection+json

1015 **JSON serialization:**

```

1016 { "self": string,
1017   "name": string, ?
1018   "description": string, ?
1019   "created": string, ?
1020   "properties": { "name": string, + }, ?
1021   "systemTemplates": [
1022     { "href": string }, +
1023   ], ?
1024   "operations": [
1025     { "rel": "add", "href": string }, ?
1026     { "rel": "edit", "href": string } ?
1027   ] ?
1028   ...
1029 }
    
```

1030 **XML media type:** application/CIMI-SystemTemplateCollection+xml

1031 **XML serialization:**

```

1032 <SystemTemplateCollection xmlns="http://www.dmtf.org/cimi">
1033   <self> xs:anyURI </self>
1034   <name> xs:string </name> ?
1035   <description> xs:string </description> ?
1036   <created> xs:string </created>
1037   <property name="xs:string"> xs:string </property> *
1038   <systemTemplate href="xs:anyURI"/> *
1039   <operation rel="add" href="xs:anyURI"/> ?
1040   <operation rel="edit" href="xs:anyURI"/> ?
1041   <xs:any>*
    
```

1042 </SystemTemplateCollection>

1043 5.9.2.1 Operations

1044 This entity supports the Read and Update operations. Creation of new System Template entities is
1045 supported via a POST to the "addLink" URI as described in section 4.2.2.1.

1046 5.9.3 System

1047 A System is a realized entity that consists of for example: one or more Machines, Volumes, and Networks
1048 (and others) that could be connected and associated with each other. A System can be operated and
1049 managed as a single entity and usually forms a stack of service. For example, a shopping cart system
1050 consists of machines for web servers and databases, network addresses for public access, and volumes
1051 for database files. A System may directly provide a user facing component, or may provide an
1052 infrastructure component.

Name	System	
Type URI	http://www.dmtf.org/cimi/System	
Attribute	Type	Description
state	string	Current state of the System entity as last known. This is a label containing lifecycle state (e.g. INITIATED, CREATING, CREATED, DESTROYING, DESTROYED). This value is read-only and will change based on the state of the System. Properties: Mandatory / Mutable
machines	ref[]	The list of Machines contained in this System. Adding an item (of type Machine) to this list is logically equivalent to associating the Machine to this System with a "containment relationship". Removing an item from this list is logically equivalent to de-associating the Machine from this System. Properties: Optional / Mutable
volumes	ref[]	The list of Volumes contained in this System. Adding an item (of type Volume) to this list is logically equivalent to associating the Volume to this System with a "containment relationship". Removing an item from this list is logically equivalent to de-associating the Volume from this System. Properties: Optional / Mutable
networks	ref[]	The list of Networks contained in this System. Adding an item (of type Network) to this list is logically equivalent to associating the Network to this System with a "containment relationship". Removing an item from this list is logically equivalent to de-associating the Network from this System. Properties: Optional / Mutable
meters	ref[]	A list of references to Meters monitored for this System. Properties: Optional / Mutable
eventLog	ref	A reference to the EventLog of this System. Properties: Optional / Mutable

1053 **JSON media type:** application/CIMI-System+json

1054 **JSON serialization:**

```

1055 { "self": string,
1056     "name": string, ?
1057     "description": string, ?
1058     "created": string, ?
1059     "properties": { "name": string, + }, ?
1060     "state": string,
1061     "machines": [
1062         { "href": string }, +
1063     ], ?
1064     "volumes": [
1065         { "href": string }, +
1066     ], ?
1067     "networks": [
1068         { "href": string }, +
1069     ], ?
1070     "meters": [
1071         { "href": string }, +
1072     ], ?
1073     "eventLog": { "href": string }, ?
1074     "operations": [
1075         { "rel": "edit", "href": string }, ?
1076         { "rel": "delete", "href": string } ?
1077     ] ?
1078     ...
1079 }
```

1080 **XML media type:** application/CIMI-System+xml

1081 **XML serialization:**

```

1082 <System xmlns="http://www.dmtf.org/cimi">
1083   <self> xs:anyURI </self>
1084   <name> xs:string </name> ?
1085   <description> xs:string </description> ?
1086   <created> xs:string </created>
1087   <property name="xs:string"> xs:string </property> *
1088   <state> xs:string </state>
1089   <machine href="xs:anyURI"/> *
1090   <volume href="xs:anyURI"/> *
1091   <network href="xs:anyURI"/> *
1092   <meter href="xs:anyURI"/> *
1093   <eventLog href="xs:anyURI"/> ?
1094   <operation rel="edit" href="xs:anyURI"/> ?
1095   <operation rel="delete" href="xs:anyURI"/> ?
1096   <xs:any*>
1097 </System>
```

1098 5.9.3.1 Operations

1099 This entity supports the Read, Update and Delete operations. Create is supported via the System
1100 Collection entity.

1101 5.9.4 System Collection

1102 A System Collection entity represents the collection of System entities within a Provider. This entity can
1103 be used to locate and create Systems.

Name	SystemCollection
------	------------------

Type URI	http://www.dmtf.org/cimi/SystemCollection	
Attribute	Type	Description
systems	ref[]	An array of references to the set of Systems in the Provider. Properties: Optional / Mutable

1104 The following describes the serialization of the entity in both JSON and XML:

1105 **JSON media type:** application/CIMI-SystemCollection+json

1106 **JSON serialization:**

```

1107 { "self": string,
1108   "name": string, ?
1109   "description": string, ?
1110   "created": string, ?
1111   "properties": { "name": string, + }, ?
1112   "systems": [
1113     { "href": string }, +
1114   ], ?
1115   "operations": [
1116     { "rel": "add", "href": string }, ?
1117     { "rel": "edit", "href": string } ?
1118   ] ?
1119   ...
1120 }
```

1121 **XML media type:** application/CIMI-SystemCollection+xml

1122 **XML serialization:**

```

1123 <SystemCollection xmlns="http://www.dmtf.org/cimi">
1124   <self> xs:anyURI </self>
1125   <name> xs:string </name> ?
1126   <description> xs:string </description> ?
1127   <created> xs:string </created>
1128   <property name="xs:string"> xs:string </property> *
1129   <system href="xs:anyURI"/> *
1130   <operation rel="add" href="xs:anyURI"/> ?
1131   <operation rel="edit" href="xs:anyURI"/> ?
1132   <xs:any>*
1133 </SystemCollection>
```

1134 5.9.4.1 Operations

1135 This entity supports the Read and Update operations.

1136 The following custom operations are also defined:

1137 **Creating a New System**

1138 **/link@rel:** add

1139 This operation will create a new System.

1140 Input parameters: Either a reference to a System Template or a System Template itself.

1141 Output parameters: A reference to a new System and optionally the representation of the System.

1142 • **HTTP/REST Protocol**

1143 To create a new System a POST is sent to the "add" URI of the SystemCollection where the HTTP
 1144 request body SHALL be as described below. Note this structure allows for certain properties to be
 1145 passed in "by value" or by "reference". The definition of each property can be found in section 5.9.1.

1146 **JSON media type:** application/CIMI-SystemCreate+json

1147 **JSON serialization:**

```

1148 { "name": string,
1149   "description": string, ?
1150   "properties": { "name": string, + }, ?
1151   "systemTemplate" : { "href": string, ?
1152     "properties": { "name": string, + }, ?
1153     "volumeTemplates": [
1154       { "href": string, ?
1155         "name": string, ?
1156         "description": string, ?
1157         "properties": { "name": string, + }, ?
1158         "volumeConfig": { "href": string, ?
1159           "properties": { "name": string, + }, ?
1160           "format": string, ?
1161           "capacity": { "quantity": number, "units": string }, ?
1162           "supportsSnapshots": boolean, ?
1163           "guestInterface": string, ?
1164         },
1165         "volumeImage": { "href": string,
1166           "imageLocation": { "href": string }, ?
1167           "imageData": string, ?
1168           "bootable": boolean ?
1169         } ?
1170       }, +
1171     ], ?
1172   "machineTemplates": [
1173     { "href": string, ?
1174       "name": string, ?
1175       "description": string, ?
1176       "properties": { "name": string, + }, ?
1177       "machineConfig": { "href": string, ?
1178         "cpu": string, ?
1179         "memory": { "quantity": integer, "units": string }, ?
1180         "disks" : [
1181           { "capacity": { "quantity": integer, "units": string },
1182             "guestInterface": string }, +
1183         ] ?
1184       },
1185       "machineImage": { "href": string,
1186         "imageLocation": { "href": string }, ?
1187         "imageData": string, ?
1188       }, ?
1189       "machineAdmin": { "href": string, ?
1190         <provider specific data> ?
1191       }, ?
1192       "volumes": [
1193         { "href": string, "attachmentPoint": string, "protocol": string}, +
1194       ], ?
1195     "volumeTemplates": [
1196       { "href": string, ?
1197         "name": string, ?
1198         "description": string, ?
1199         "properties": { "name": string, + }, ?
1200         "attachmentPoint": string, "protocol": string,
1201         "volumeConfig": { "href": string, ?
1202           "properties": { "name": string, + }, ?

```



```

1203         "format": string, ?
1204         "capacity": { "quantity": number, "units": string }, ?
1205         "supportsSnapshots": boolean, ?
1206         "guestInterface": string ?
1207     }, ?
1208     "volumeImage": { "href": string,
1209         "properties": { "name": string, + }, ?
1210         "imageLocation": { "href": string }, ?
1211         "imageData": string, ?
1212         "bootable": boolean ?
1213     } ?
1214 }, +
1215 ] ?
1216 }
1217 ], ?
1218 "networkTemplates": [
1219     { "href": string, ?
1220       "name": string, ?
1221       "description": string, ?
1222       "properties": { "name": string, + }, ?
1223       "networkConfig": { "href": string },
1224       "routingGroup": { "href": string } ?
1225     }
1226 ] ?
1227 }
1228 ...
1229 }

```

1230 **XML media type:** application/CIMI-SystemCreate+xml

1231 **XML serialization**

```

1232 <SystemCreate>
1233   <name> xs:string </name>
1234   <description> xs:string </description> ?
1235   <property name="xs:string"> xs:string </property> *
1236   <systemTemplate href="xs:anyURI"?>
1237     <property name="xs:string"> xs:string </property> *
1238   </systemTemplate>
1239   <volumeTemplate href="xs:anyURI"? >
1240     <name> xs:string </name>
1241     <description> xs:string </description> ?
1242     <property name="xs:string"> xs:string </property> *
1243     <volumeConfig href="xs:anyURI"? >
1244       <property name="xs:string"> xs:string </property> *
1245       <format> xs:string </format> ?
1246       <capacity quantity="xs:integer" units="xs:string"/> ?
1247       <supportsSnapshots> xs:boolean </supportsSnapshots> ?
1248       <guestInterface> xs:string </guestInterface> ?
1249     </volumeConfig>
1250     <volumeImage href="xs:anyURI">
1251       <property name="xs:string"> xs:string </property> *
1252       <imageLocation href="xs:anyURI"/> ?
1253       <imageData> xs:any* </imageData> ?
1254       <bootable> xs:boolean </bootable> ?
1255     </volumeImage> ?
1256   </volumeTemplate> *
1257 </SystemCreate>
1258 <machineTemplate href="xs:anyURI"? >
1259   <name> xs:string </name>
1260   <description> xs:string </description> ?
1261   <property name="xs:string"> xs:string </property> *
1262   <machineConfig href="xs:anyURI"? >

```

```

1263 <property name="xs:string"> xs:string </property> *
1264 <cpu> xs:string </cpu> ?
1265 <memory quantity="xs:integer" units="xs:string"/>
1266 <disk>
1267     <capacity quantity="xs:integer" units="xs:string">
1268     <guestInterface> xs:string </guestInterface>
1269 </disk> *
1270 </machineConfig>
1271 <machineImage href="xs:anyURI">
1272     <property name="xs:string"> xs:string </property> *
1273     <imageLocation href="xs:anyURI"/> ?
1274     <imageData> xs:string </imageData> ?
1275 </machineImage>
1276 <machineAdmin href="xs:anyURI"? >
1277     xs:any* <!-- provider specific data -->
1278 </machineAdmin> ?
1279
1280 <volume href="xs:anyURI"
1281     attachmentPoint="xs:string" protocol="xs:string" /> *
1282 <volumeTemplate href="xs:anyURI"?
1283     attachmentPoint="xs:string" protocol="xs:string" >
1284     <name> xs:string </name>
1285     <description> xs:string </description> ?
1286     <property name="xs:string"> xs:string </property> *
1287     <volumeConfig href="xs:anyURI">
1288         <property name="xs:string"> xs:string </property> *
1289         <format> xs:string </format> ?
1290         <capacity quantity="xs:integer" units="xs:string"/> ?
1291         <supportsSnapshots> xs:boolean </supportsSnapshots> ?
1292         <guestInterface> xs:string </guestInterface> ?
1293     </volumeConfig>
1294     <volumeImage href="xs:anyURI">
1295         <property name="xs:string"> xs:string </property> *
1296         <imageLocation href="xs:anyURI"/> ?
1297         <imageData> xs:any* </imageData> ?
1298         <bootable> xs:boolean </bootable> ?
1299     </volumeImage> ?
1300 </volumeTemplate> *
1301 </machineTemplate> *
1302
1303 <networkTemplate href="xs:anyURI"? >
1304     <name> xs:string </name>
1305     <description> xs:string </description> ?
1306     <property name="xs:string"> xs:string </property> *
1307     <networkConfig href="xs:anyURI"/>
1308     <routingGroup href="xs:anyURI"/> ?
1309 </networkTemplate> *
1310
1311 </systemTemplate>
1312 <xs:any>*
1313 </SystemCreate>

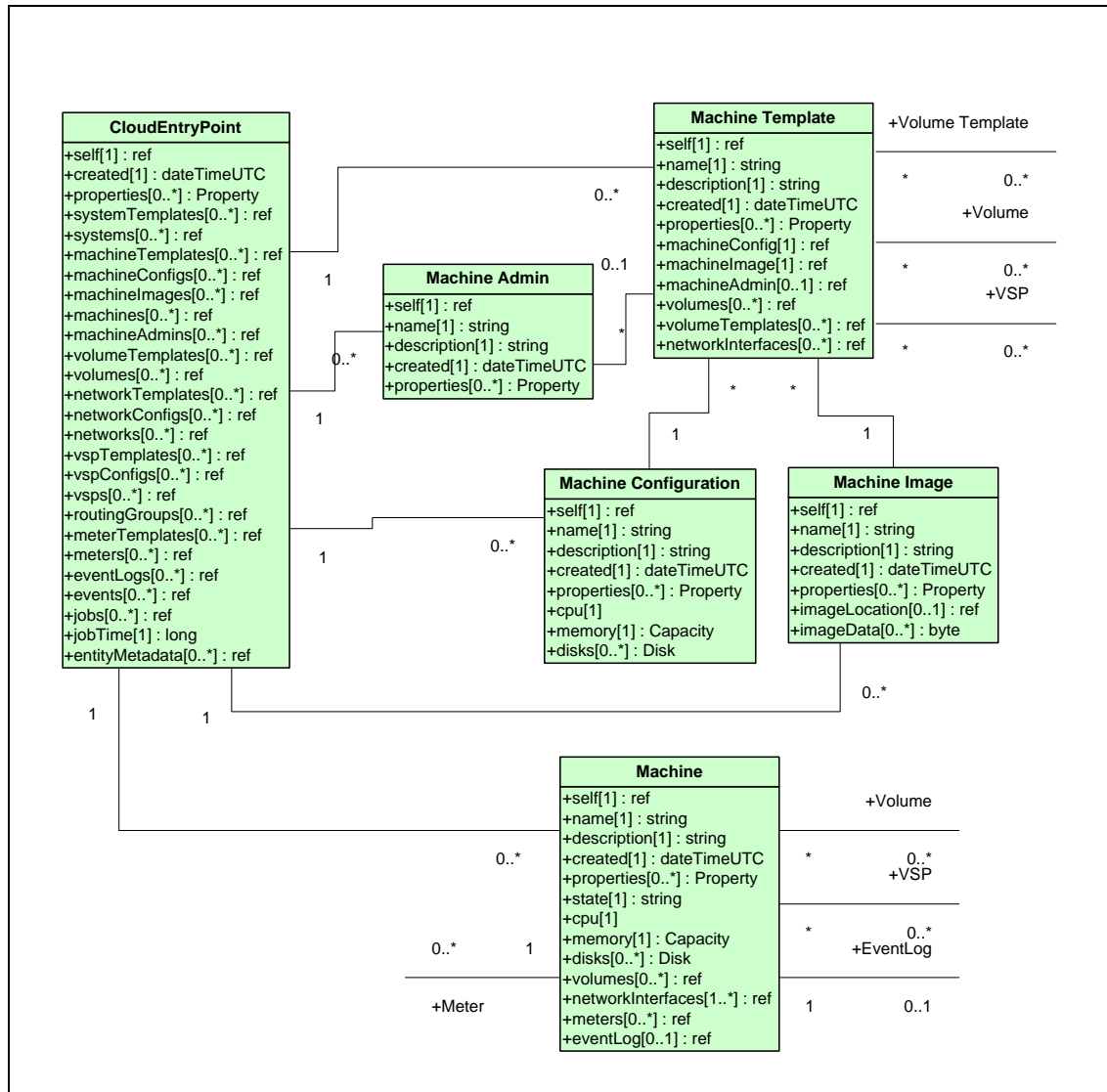
```

1314 The serialization of some reference properties are specified such that a request MAY either include a
1315 reference ("href") to an existing entity or to include the entity "inline" as a set of additional properties.
1316 Requests SHALL NOT include both a reference and the inlined set of properties.

1317 Upon successful processing of the request, the HTTP response body MAY either be empty or contain a
1318 serialization of the System entity.

1319 **5.10 Machine Entities and Relationships**

1320 The following diagram illustrates the entities involved in constructing a Machine and their relationships.
 1321 Although this drawing is in the style of an Entity Relationship diagram, the use of UML is neither rigorous
 1322 nor normative.



1323
 1324 **Figure 2 - Machine Entities**

1325 **5.10.1 Machine Template**

1326 A Machine Template represents the set of metadata and instructions used in the creation of a Machine.

Name	MachineTemplate	
Type URI	http://www.dmtf.org/cimi/MachineTemplate	
Attribute	Type	Description
machineConfig	ref	A reference to the Machine Configuration that will be used to create a Machine from this Machine Template.

		<p>Properties: Mandatory / Mutable</p>															
machineImage	ref	<p>A reference to the Machine Image that will be used to create a Machine from this Machine Template.</p> <p>Properties: Mandatory / Mutable</p>															
machineAdmin	ref	<p>A reference to the Machine Admin that will be used to create the initial login credential for the new Machine.</p> <p>Properties: Optional / Mutable</p>															
volumes	volume[]	<p>A list of references to existing Volumes that will be attached to the Machine during its creation.</p> <p>Each volume attribute has the following sub-attributes which describe aspects of the way in which the Machine will be attached to the Volume:</p> <table border="1"> <thead> <tr> <th>Name</th> <th colspan="2">volume</th> </tr> <tr> <th>Attribute</th> <th>Type</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>attachmentPoint</td> <td>String</td> <td>File system path where the Volume will be attached. Properties: Mandatory / Mutable</td> </tr> <tr> <td>protocol</td> <td>String</td> <td>Protocol that will be used to access this Volume (e.g. NFS, iSCSI). Properties: Mandatory / Mutable</td> </tr> <tr> <td>volume</td> <td>ref</td> <td>Reference to the Volume that will be attached. Properties: Mandatory / Mutable</td> </tr> </tbody> </table> <p>Properties: Optional / Mutable</p>	Name	volume		Attribute	Type	Description	attachmentPoint	String	File system path where the Volume will be attached. Properties: Mandatory / Mutable	protocol	String	Protocol that will be used to access this Volume (e.g. NFS, iSCSI). Properties: Mandatory / Mutable	volume	ref	Reference to the Volume that will be attached. Properties: Mandatory / Mutable
Name	volume																
Attribute	Type	Description															
attachmentPoint	String	File system path where the Volume will be attached. Properties: Mandatory / Mutable															
protocol	String	Protocol that will be used to access this Volume (e.g. NFS, iSCSI). Properties: Mandatory / Mutable															
volume	ref	Reference to the Volume that will be attached. Properties: Mandatory / Mutable															
volumeTemplates	volumeTemplate[]	<p>A list of references to Volume Templates that will be used to create a set of new Volumes that will to be attached to the Machine during its creation.</p> <p>If the Machine is created as part of a System creation, the Volumes created from these templates will be considered as part of that System without the need for these Volume Templates to also be listed in the <code>volumeTemplates</code> attribute of the relevant System Template. If the same Volume Template reference is listed in both the <code>volumeTemplates</code> attribute of a System Template and in the <code>volumeTemplates</code> attribute of a Machine Template contained by that System Template, this means that multiple, distinct Volume instances will be created as part of the overall System creation.</p> <p>Each volumeTemplate attribute has the following sub-attributes which describe aspects of the way in which the Machine will be attached to the Volume instance that will be created from the template:</p>															

		<table border="1"> <tr> <th>Name</th> <td colspan="2">volumeTemplate</td> </tr> <tr> <th>Attribute</th> <th>Type</th> <th>Description</th> </tr> <tr> <td>attachmentPoint</td> <td>String</td> <td>File system path where the Volume will be attached. Properties: Mandatory / Mutable</td> </tr> <tr> <td>protocol</td> <td>String</td> <td>Protocol that will be used to access this Volume (e.g. NFS, iSCSI). Properties: Mandatory / Mutable</td> </tr> <tr> <td>volumeTemplate</td> <td>ref</td> <td>Reference to the Volume Template that will be used to create a new Volume. Properties: Mandatory / Mutable</td> </tr> </table> <p>Properties: Optional / Mutable</p>	Name	volumeTemplate		Attribute	Type	Description	attachmentPoint	String	File system path where the Volume will be attached. Properties: Mandatory / Mutable	protocol	String	Protocol that will be used to access this Volume (e.g. NFS, iSCSI). Properties: Mandatory / Mutable	volumeTemplate	ref	Reference to the Volume Template that will be used to create a new Volume. Properties: Mandatory / Mutable
Name	volumeTemplate																
Attribute	Type	Description															
attachmentPoint	String	File system path where the Volume will be attached. Properties: Mandatory / Mutable															
protocol	String	Protocol that will be used to access this Volume (e.g. NFS, iSCSI). Properties: Mandatory / Mutable															
volumeTemplate	ref	Reference to the Volume Template that will be used to create a new Volume. Properties: Mandatory / Mutable															
networkInterfaces	networkInterface[]	<p>A list of sub-entities that define the network interfaces that will be created on Machines instantiated from this template.</p> <table border="1"> <tr> <th>Name</th> <td colspan="2">networkInterface</td> </tr> <tr> <th>Attribute</th> <th>Type</th> <th>Description</th> </tr> <tr> <td>vsp</td> <td>ref</td> <td>A reference to the VSP (Virtual Switch Port) for this network interface. Note this is a reference to a VSP and not a VSPTemplate. It is expected that VSPs and Networks will be defined separately and prior to the Machines that connect to them. Properties: Optional / Mutable</td> </tr> <tr> <td>hostname</td> <td>string</td> <td>DNS resolvable name associated with this network interface. While this attribute can be specified in most cases it is expected to be supplied by the Provider. Specifying this value is typically only done when the Template is only used for one particular Machine. Properties: Optional / Mutable</td> </tr> <tr> <td>macAddress</td> <td>string</td> <td>Address assigned by the hypervisor when a machine is</td> </tr> </table>	Name	networkInterface		Attribute	Type	Description	vsp	ref	A reference to the VSP (Virtual Switch Port) for this network interface. Note this is a reference to a VSP and not a VSPTemplate. It is expected that VSPs and Networks will be defined separately and prior to the Machines that connect to them. Properties: Optional / Mutable	hostname	string	DNS resolvable name associated with this network interface. While this attribute can be specified in most cases it is expected to be supplied by the Provider. Specifying this value is typically only done when the Template is only used for one particular Machine. Properties: Optional / Mutable	macAddress	string	Address assigned by the hypervisor when a machine is
Name	networkInterface																
Attribute	Type	Description															
vsp	ref	A reference to the VSP (Virtual Switch Port) for this network interface. Note this is a reference to a VSP and not a VSPTemplate. It is expected that VSPs and Networks will be defined separately and prior to the Machines that connect to them. Properties: Optional / Mutable															
hostname	string	DNS resolvable name associated with this network interface. While this attribute can be specified in most cases it is expected to be supplied by the Provider. Specifying this value is typically only done when the Template is only used for one particular Machine. Properties: Optional / Mutable															
macAddress	string	Address assigned by the hypervisor when a machine is															

			<p>created or a unique address can be manually assigned.</p> <p>While this attribute can be specified in most cases it is expected to be supplied by the Provider. Specifying this value is typically only done when the Template is only used for one particular Machine.</p> <p>Properties: Optional / Mutable</p>
		state	<p>string</p> <p>The state of an interface configurable to be “Active” or “Standby”.</p> <p>Properties: Mandatory / Mutable</p>
		protocol	<p>string</p> <p>Selected network protocol such as - IPv4 or IPv6.</p> <p>Properties: Mandatory / Mutable</p>
		allocation	<p>string</p> <p>The option for “Dynamic Host Allocation Protocol” or static.</p> <p>Properties: Mandatory / Mutable</p>
		address	<p>string</p> <p>The IP address assigned to a virtual interface.</p> <p>While this attribute can be specified in most cases it is expected to be supplied by the Provider. Specifying this value is typically only done when the Template is only used for one particular Machine.</p> <p>Properties: Optional / Mutable</p>
		defaultGateway	<p>string</p> <p>An IP address to a firewall or router that serves other networks.</p> <p>Properties: Optional / Mutable</p>
		dns	<p>string</p> <p>The IP address of the Domain Name Service from host name to IP resolution.</p> <p>Properties: Optional / Mutable</p>
		maxTransmissionUnit	<p>integer</p> <p>To set the largest supported</p>

				packet size.
			Properties: Optional / Mutable	
Properties: Optional / Mutable				

1327 The following describes the serialization of the entity in both JSON and XML:

1328 **JSON media type:** application/CIMI-MachineTemplate+json

1329 **JSON serialization:**

```

1330 { "self": string,
1331   "name": string,
1332   "description": string, ?
1333   "created": string, ?
1334   "properties": { "name": string, + }, ?
1335   "machineConfig": { "href": string },
1336   "machineImage": { "href": string },
1337   "machineAdmin": { "href": string }, ?
1338   "volumes": [
1339     { "href": string, "attachmentPoint": string, "protocol": string }, +
1340   ], ?
1341   "volumeTemplates": [
1342     { "href": string, "attachmentPoint": string, "protocol": string }, +
1343   ], ?
1344   "networkInterfaces": [
1345     { "vsp": { "href": string }, "hostname": string, "macAddress": string,
1346       "state": string, "protocol": string, "allocation": string,
1347       "address": string, "defaultGateway": string, "dns": string,
1348       "maxTransmissionUnit": integer }, +
1349   ], ?
1350   "operations": [
1351     { "rel": "edit", "href": string }, ?
1352     { "rel": "delete", "href": string } ?
1353   ] ?
1354   ...
1355 }
```

1356 **XML media type:** application/CIMI-MachineTemplate+xml

1357 **XML serialization:**

```

1358 <MachineTemplate xmlns="http://www.dmtf.org/cimi">
1359   <self> xs:anyURI </self>
1360   <name> xs:string </name>
1361   <description> xs:string </description> ?
1362   <created> xs:string </created>
1363   <property name="xs:string"> xs:string </property> *
1364   <machineConfig href="xs:anyURI"/>
1365   <machineImage href="xs:anyURI"/>
1366   <machineAdmin href="xs:anyURI"/> ?
1367   <volume href="xs:anyURI"
1368     attachmentPoint="xs:string" protocol="xs:string" /> *
1369   <volumeTemplate href="xs:anyURI"
1370     attachmentPoint="xs:string" protocol="xs:string" /> *
1371   <networkInterface>
1372     <vsp href="xs:anyURI"/>
1373     <hostname> xs:string </hostname>
1374     <macAddress> xs:string </macAddress>
1375     <state> xs:string </state>
1376     <protocol> xs:string </protocol>
1377     <allocation> xs:string </alloction>
```

```

1378 <address> xs:string </address>
1379 <defaultGateway> xs:string </defaultGateway>
1380 <dns> xs:string </dns>
1381 <maxTransmissionUnit> xs:integer </maxTransmissionUnit>
1382 </networkInterface> *
1383 <operation rel="edit" href="xs:anyURI"/> ?
1384 <operation rel="delete" href="xs:anyURI"/> ?
1385 <xs:any>*
1386 </MachineTemplate>
    
```

1387 **5.10.1.1 Operations**

1388 This entity supports the Read, Update and Delete operations. Create is supported via the Machine
 1389 Template Collection entity.

1390 **5.10.2 Machine Template Collection**

1391 A Machine Template Collection entity represents the collection of Machine Template entities within a
 1392 Provider. This entity can be used to locate and create Machine Templates.

Name	MachineTemplateCollection	
Type URI	http://www.dmtf.org/cimi/MachineTemplateCollection	
Attribute	Type	Description
machineTemplates	ref[]	An array of references to the set of Machine Templates in the Provider. Properties: Optional / Mutable

1393 The following describes the serialization of the entity in both JSON and XML:

1394 **JSON media type:** application/CIMI-MachineTemplateCollection+json

1395 **JSON serialization:**

```

1396 { "self": string,
1397   "name": string, ?
1398   "description": string, ?
1399   "created": string, ?
1400   "properties": { "name": string, + }, ?
1401   "machineTemplates": [
1402     { "href": string }, +
1403   ], ?
1404   "operations": [
1405     { "rel": "add", "href": string }, ?
1406     { "rel": "edit", "href": string } ?
1407   ] ?
1408   ...
1409 }
    
```

1410 **XML media type:** application/CIMI-MachineTemplateCollection+xml

1411 **XML serialization:**

```

1412 <MachineTemplateCollection xmlns="http://www.dmtf.org/cimi">
1413 <self> xs:anyURI </self>
1414 <name> xs:string </name> ?
1415 <description> xs:string </description> ?
1416 <created> xs:string </created>
1417 <property name="xs:string"> xs:string </property> *
1418 <machineTemplate href="xs:anyURI"/> *
    
```



```

1419 <operation rel="add" href="xs:anyURI"/> ?
1420 <operation rel="edit" href="xs:anyURI"/> ?
1421 <xs:any>*
1422 </MachineTemplateCollection>

```

1423 5.10.2.1 Operations

1424 This entity supports the Read and Update operations. Creation of new Machine Template entities is
 1425 supported via a POST to the "addLink" URI as described in section 4.2.2.1.

1426 5.10.3 Machine Configuration

1427 The Machine Configuration entity represents the set of configuration values that define the (virtual)
 1428 hardware resources of a to-be-realized Machine Instance. Machine Configurations are created by
 1429 Providers and MAY, at the Providers discretion, be created by Consumers.

Name	MachineConfiguration										
Type URI	http://www.dmtf.org/cimi/MachineConfiguration										
Attribute	Type	Description									
cpu	<i>TBD</i>	Indicate the amount of CPU (based on standard CPU measurement) that a Machine realized from this configuration would have, by default. Properties: Mandatory / Mutable									
memory	structure	Indicates the amount of RAM that a Machine realized from this configuration will have. This attribute has the following sub-attributes which serve to describe it: <table border="1" data-bbox="495 1087 1442 1455"> <thead> <tr> <th>Attribute</th> <th>Type</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>quantity</td> <td>integer</td> <td>A numerical quantity expressed as an integer. Properties: Mandatory / Mutable</td> </tr> <tr> <td>units</td> <td>string</td> <td>An enumerated value that expresses the unit of measurement used. Allowable values are byte, kibibyte, mebibyte, gibibyte, tebibyte, pebibyte, exbibyte, zebibyte, and yobibyte. Properties: Mandatory / Mutable</td> </tr> </tbody> </table>	Attribute	Type	Description	quantity	integer	A numerical quantity expressed as an integer. Properties: Mandatory / Mutable	units	string	An enumerated value that expresses the unit of measurement used. Allowable values are byte , kibibyte , mebibyte , gibibyte , tebibyte , pebibyte , exbibyte , zebibyte , and yobibyte . Properties: Mandatory / Mutable
Attribute	Type	Description									
quantity	integer	A numerical quantity expressed as an integer. Properties: Mandatory / Mutable									
units	string	An enumerated value that expresses the unit of measurement used. Allowable values are byte , kibibyte , mebibyte , gibibyte , tebibyte , pebibyte , exbibyte , zebibyte , and yobibyte . Properties: Mandatory / Mutable									
disks	disk[]	Contains the list of metadata of the disks that will be created upon the instantiation of a Machine from this configuration. The disks are local storages to the Machine. Each disks attribute has the following sub-attributes: <table border="1" data-bbox="495 1690 1442 1883"> <thead> <tr> <th>Name</th> <td colspan="2">disk</td> </tr> <tr> <th>Attribute</th> <th>Type</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>capacity</td> <td></td> <td>Indicates the initial capacity of the disk described by this attribute. This property has the following, sub-attributes.</td> </tr> </tbody> </table>	Name	disk		Attribute	Type	Description	capacity		Indicates the initial capacity of the disk described by this attribute. This property has the following, sub-attributes.
Name	disk										
Attribute	Type	Description									
capacity		Indicates the initial capacity of the disk described by this attribute. This property has the following, sub-attributes.									

Attribute	Type	Description
quantity	integer	A numerical quantity expressed as an integer. Properties: Mandatory / Mutable
units	string	An enumerated value that expresses the unit of measurement used. Allowable values are byte, kilobyte, megabyte, gigabyte, terabyte, petabyte, exabyte, zettabyte, and yottabyte. Properties: Mandatory / Mutable
Properties: Mandatory / Mutable		
format	string	The format/type of this disk (e.g. ext4, NTFS). Properties: Mandatory / Mutable
attachmentPoint	string	File system path where this disk is attached. Properties: Mandatory / Mutable
Properties: Optional / Mutable		

1430 Note: The disk attributes "format" and "attachmentPoint" will not appear on Machine entities because
 1431 once the Machine is created the user of the Machine will be able to modify those attributes of a disk
 1432 without the Provider's knowledge - therefore it is no longer an aspect of the Machine that the Provider can
 1433 reliably manage.

1434 **JSON media type:** application/CIMI-MachineConfiguration+json

1435 **JSON serialization:**

```

1436 { "self": string,
1437   "name": string,
1438   "description": string, ?
1439   "created": string, ?
1440   "properties": { "name": string, + }, ?
1441   "cpu": string,
1442   "memory": { "quantity": integer, "units": string },
1443   "disks" : [
1444     { "capacity": { "quantity": integer, "units": string }, +
1445   ], ?
1446   "operations": [
1447     { "rel": "edit", "href": string }, ?
1448     { "rel": "delete", "href": string } ?
1449   ] ?
1450   ...
1451 }
```

1452 **XML media type:** application/CIMI-MachineConfiguration+xml

1453 **XML serialization:**

```

1454 <MachineConfiguration xmlns="http://www.dmtf.org/cimi">
1455   <self> xs:anyURI </self>
```

```

1456 <name> xs:string </name>
1457 <description> xs:string </description> ?
1458 <created> xs:string </created>
1459 <property name="xs:string"> xs:string </property> *
1460 <cpu> xs:string </cpu>
1461 <memory quantity="xs:integer" units="xs:string"/>
1462 <disk>
1463   <capacity quantity="xs:integer" units="xs:string">
1464 </disk> *
1465 <operation rel="edit" href="xs:anyURI"/> ?
1466 <operation rel="delete" href="xs:anyURI"/> ?
1467 <xs:any>*
1468 </MachineConfiguration>

```

1469 5.10.3.1 Operations

1470 This entity supports the Read, Update and Delete operations. Create is supported via the Machine
1471 Configuration Collection entity.

1472 5.10.4 Machine Configuration Collection

1473 A Machine Configuration Collection entity represents the collection of Machine Configuration entities
1474 within a Provider. This entity can be used to locate and create Machine Configurations.

Name	MachineConfigurationCollection	
Type URI	http://www.dmtf.org/cimi/MachineConfigurationCollection	
Attribute	Type	Description
machineConfigurations	ref[]	An array of references to the set of Machine Configurations in the Provider. Properties: Optional / Mutable

1475 The following describes the serialization of the entity in both JSON and XML:

1476 **JSON media type:** application/CIMI-MachineConfigurationCollection+json

1477 **JSON serialization:**

```

1478 { "self": string,
1479   "name": string, ?
1480   "description": string, ?
1481   "created": string, ?
1482   "properties": { "name": string, + }, ?
1483   "machineConfigurations": [
1484     { "href": string }, +
1485   ], ?
1486   "operations": [
1487     { "rel": "add", "href": string }, ?
1488     { "rel": "edit", "href": string } ?
1489   ] ?
1490   ...
1491 }

```

1492 **XML media type:** application/CIMI-MachineConfigurationCollection+xml

1493 **XML serialization:**

```

1494 <MachineConfigurationCollection xmlns="http://www.dmtf.org/cimi">
1495   <self> xs:anyURI </self>

```

```

1496 <name> xs:string </name> ?
1497 <description> xs:string </description> ?
1498 <created> xs:string </created>
1499 <property name="xs:string"> xs:string </property> *
1500 <machineConfiguration href="xs:anyURI"/> *
1501 <operation rel="add" href="xs:anyURI"/> ?
1502 <operation rel="edit" href="xs:anyURI"/> ?
1503 <xs:any>*
1504 </MachineConfigurationCollection>
    
```

1505 **5.10.4.1 Operations**

1506 This entity supports the Read and Update operations. Creation of new Machine Configuration entities is
 1507 supported via a POST to the "addLink" URI as described in section 4.2.2.1.

1508 **5.10.5 Machine Image**

1509 This entity represents the information (e.g. an Open Virtualization Format (OVF) package) necessary for
 1510 hardware virtualized resources to create a Machine Instance; it contains configuration data such as
 1511 startup instructions, including possible combinations of the following:

- 1512 • the software image (i.e. a copy of an installed Machine) which is to be instantiated on the disk
 1513 and other virtual resources
- 1514 • installation software, which, when executed on the hardware (virtual) resources, builds the
 1515 machine instance
- 1516 • both a disk image and a set of software and parameters in order to install new components not
 1517 included in original disk image

Name	MachineImage	
Type URI	http://www.dmtf.org/cimi/MachineImage	
Attribute	Type	Description
imageLocation	ref	A reference to the location of the binary data that makes up this image. Either this attribute or imageData SHALL be present; however both values SHALL NOT be present simultaneously. Properties: Optional / Immutable
imageData	byte[]	The binary data that makes up this image. Either this attribute or imageLocation SHALL be present; however both values SHALL NOT be present simultaneously. Properties: Optional / Immutable

1518 The following describes the serialization of the entity in both JSON and XML:

1519 JSON media type: **application/CIMI-MachineImage+json**

1520 **JSON serialization:**

```

1521 { "self": string,
1522   "name": string,
1523   "description": string, ?
1524   "created": string, ?
1525   "properties": { "name": string, + }, ?
1526   "imageLocation": { "href": string }, ?
1527   "imageData": string, ?
    
```

```

1528     "operations": [
1529       { "rel": "edit", "href": string }, ?
1530       { "rel": "delete", "href": string } ?
1531     ] ?
1532     ...
1533   }

```

1534 **XML media type:** application/CIMI-MachineImage+xml

1535 **XML serialization:**

```

1536 <MachineImage xmlns="http://www.dmtf.org/cimi">
1537   <self> xs:anyURI </self>
1538   <name> xs:string </name>
1539   <description> xs:string </description> ?
1540   <created> xs:string </created>
1541   <property name="xs:string"> xs:string </property> *
1542   <imageLocation href="xs:anyURI"/> ?
1543   <imageData> xs:string </imageData> ?
1544   <operation rel="edit" href="xs:anyURI"/> ?
1545   <operation rel="delete" href="xs:anyURI"/> ?
1546   <xs:any*>
1547 </MachineImage>

```

1548 5.10.5.1 Operations

1549 This entity supports the Read, Update and Delete operations. Create is supported via the Machine Image
1550 Collection entity.

1551 5.10.6 Machine Image Collection

1552 A Machine Image Collection entity represents the collection of Machine Image entities within a Provider.
1553 This entity can be used to locate and create Machine Images.

Name	MachineImageCollection	
Type URI	http://www.dmtf.org/cimi/MachineImageCollection	
Attribute	Type	Description
machineImages	ref[]	An array of references to the set of Machine Images in the Provider. Properties: Optional / Mutable

1554 The following describes the serialization of the entity in both JSON and XML:

1555 **JSON media type:** application/CIMI-MachineImageCollection+json

1556 **JSON serialization:**

```

1557 { "self": string,
1558   "name": string, ?
1559   "description": string, ?
1560   "created": string, ?
1561   "properties": { "name": string, + }, ?
1562   "machineImages": [
1563     { "href": string }, +
1564   ], ?
1565   "operations": [
1566     { "rel": "add", "href": string }, ?
1567     { "rel": "edit", "href": string } ?
1568   ] ?

```

1569 }
 1570 }

1571 **XML media type:** application/CIMI-MachineImageCollection+xml

1572 **XML serialization:**

```

1573 <MachineImageCollection xmlns="http://www.dmtf.org/cimi">
1574   <self> xs:anyURI </self>
1575   <name> xs:string </name> ?
1576   <description> xs:string </description> ?
1577   <created> xs:string </created>
1578   <property name="xs:string"> xs:string </property> *
1579   <machineImage href="xs:anyURI"/> *
1580   <operation rel="add" href="xs:anyURI"/> ?
1581   <operation rel="edit" href="xs:anyURI"/> ?
1582   <xs:any>*
1583 </MachineImageCollection>
    
```

1584 **5.10.6.1 Operations**

1585 This entity supports the Read and Update operations. Creation of new Machine Image entities is
 1586 supported via a POST to the "addLink" URI as described in section 4.2.2.1.

1587 **5.10.7 Machine**

1588 An instantiated compute resource that encapsulates both CPU and Memory.

Name	Machine	
Type URI	http://www.dmtf.org/cimi/Machine	
Attribute	Type	Description
state	string	Indicates the operational state of the Machine. Allowable values include: STARTED: The Machine is available and ready for use. Allowable actions when in this state are: stop and restart . STOPPED: This is the virtual equivalent of powering off a physical Machine. There is no saved CPU or memory state. Allowable actions when in this state are: start and restart . Providers may define additional values. This value is read-only and will change based on the state of the Machine. Properties: Mandatory / Mutable
cpu	TBD	The size of the CPU allocated to this Machine to be used. This should adhere to the standard unit of measurement. For example, a Machine with 4 unit worth of CPU would allow the processes in the Machine to use up to 4 units worth of CPU (and be charged thereof). When this value is increased, it implies that the Machine is allocated more CPU to use, and vice versa when the value is decreased.

		<p>Properties: Mandatory / Mutable</p>																	
memory	structure	<p>The size of the memory (RAM) allocated to this Machine.</p> <p>When this value is increased, it implies that the Machine is allocated more RAM, and vice versa when the value is decreased.</p> <p>This attribute has the following sub-attributes which serve to describe it:</p> <table border="1"> <thead> <tr> <th>Attribute</th> <th>Type</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>quantity</td> <td>integer</td> <td> A numerical quantity expressed as an integer. Properties: Mandatory / Mutable </td> </tr> <tr> <td>units</td> <td>string</td> <td> An enumerated value that expresses the unit of measurement used. Allowable values are byte, kibibyte, mebibyte, gibibyte, tebibyte, pebibyte, exbibyte, zebibyte, and yobibyte. Properties: Mandatory / Mutable </td> </tr> </tbody> </table> <p>Properties: Mandatory / Mutable</p>	Attribute	Type	Description	quantity	integer	A numerical quantity expressed as an integer. Properties: Mandatory / Mutable	units	string	An enumerated value that expresses the unit of measurement used. Allowable values are byte , kibibyte , mebibyte , gibibyte , tebibyte , pebibyte , exbibyte , zebibyte , and yobibyte . Properties: Mandatory / Mutable								
Attribute	Type	Description																	
quantity	integer	A numerical quantity expressed as an integer. Properties: Mandatory / Mutable																	
units	string	An enumerated value that expresses the unit of measurement used. Allowable values are byte , kibibyte , mebibyte , gibibyte , tebibyte , pebibyte , exbibyte , zebibyte , and yobibyte . Properties: Mandatory / Mutable																	
disks	disk[]	<p>The list of disks (local storages) that are part of the Machine. Adding an element to this list creates a disk.</p> <p>Each disk attribute has the following sub-attributes which describe aspects of the disk:</p> <table border="1"> <thead> <tr> <th>Name</th> <td>disk</td> </tr> <tr> <th>Attribute</th> <th>Type</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>capacity</td> <td colspan="2"> Indicates the initial capacity of the disk described by this attribute. This property has the following, sub-attributes. <table border="1"> <thead> <tr> <th>Attribute</th> <th>Type</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>quantity</td> <td>integer</td> <td> A numerical quantity expressed as an integer. Properties: Mandatory / Mutable </td> </tr> <tr> <td>units</td> <td>string</td> <td> An enumerated value that expresses the unit of measurement used. Allowable values are byte, kilobyte, megabyte, gigabyte, terabyte, petabyte, exabyte, zettabyte, and yottabyte. Properties: Mandatory / Mutable </td> </tr> </tbody> </table> </td> </tr> </tbody> </table>	Name	disk	Attribute	Type	Description	capacity	Indicates the initial capacity of the disk described by this attribute. This property has the following, sub-attributes. <table border="1"> <thead> <tr> <th>Attribute</th> <th>Type</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>quantity</td> <td>integer</td> <td> A numerical quantity expressed as an integer. Properties: Mandatory / Mutable </td> </tr> <tr> <td>units</td> <td>string</td> <td> An enumerated value that expresses the unit of measurement used. Allowable values are byte, kilobyte, megabyte, gigabyte, terabyte, petabyte, exabyte, zettabyte, and yottabyte. Properties: Mandatory / Mutable </td> </tr> </tbody> </table>		Attribute	Type	Description	quantity	integer	A numerical quantity expressed as an integer. Properties: Mandatory / Mutable	units	string	An enumerated value that expresses the unit of measurement used. Allowable values are byte , kilobyte , megabyte , gigabyte , terabyte , petabyte , exabyte , zettabyte , and yottabyte . Properties: Mandatory / Mutable
Name	disk																		
Attribute	Type	Description																	
capacity	Indicates the initial capacity of the disk described by this attribute. This property has the following, sub-attributes. <table border="1"> <thead> <tr> <th>Attribute</th> <th>Type</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>quantity</td> <td>integer</td> <td> A numerical quantity expressed as an integer. Properties: Mandatory / Mutable </td> </tr> <tr> <td>units</td> <td>string</td> <td> An enumerated value that expresses the unit of measurement used. Allowable values are byte, kilobyte, megabyte, gigabyte, terabyte, petabyte, exabyte, zettabyte, and yottabyte. Properties: Mandatory / Mutable </td> </tr> </tbody> </table>		Attribute	Type	Description	quantity	integer	A numerical quantity expressed as an integer. Properties: Mandatory / Mutable	units	string	An enumerated value that expresses the unit of measurement used. Allowable values are byte , kilobyte , megabyte , gigabyte , terabyte , petabyte , exabyte , zettabyte , and yottabyte . Properties: Mandatory / Mutable								
Attribute	Type	Description																	
quantity	integer	A numerical quantity expressed as an integer. Properties: Mandatory / Mutable																	
units	string	An enumerated value that expresses the unit of measurement used. Allowable values are byte , kilobyte , megabyte , gigabyte , terabyte , petabyte , exabyte , zettabyte , and yottabyte . Properties: Mandatory / Mutable																	

		<table border="1"> <tr> <td colspan="2">Properties: Mandatory / Mutable</td> </tr> </table> <p>Properties: Optional / Mutable</p>	Properties: Mandatory / Mutable																							
Properties: Mandatory / Mutable																										
volumes	volume[]	<p>The list of networked volumes that are attached to this Machine.</p> <p>Adding a Volume to this list means that the Machine has some access to the data on the Volume. Removing a Volume from this list means that the Machine no longer has access to the data on the Volume.</p> <p>Each volume attribute has the following sub-attributes which describe aspects of the way in which the Machine is attached to the Volume:</p> <table border="1"> <tr> <td>Name</td> <td>volume</td> </tr> </table> <p>See "volume" in "5.10.1 Machine Template" for the definition of this sub-entity.</p> <p>Properties: Optional / Mutable</p>	Name	volume																						
Name	volume																									
networkInterfaces	networkInterface[]	<p>A list of sub-entities that define the network interfaces on this Machine.</p> <table border="1"> <tr> <td>Name</td> <td colspan="2">networkInterface</td> </tr> <tr> <td>Attribute</td> <td>Type</td> <td>Description</td> </tr> <tr> <td>vsp</td> <td>URI</td> <td>A reference to the VSP (Virtual Switch Port) for this network interface. Properties: Mandatory / Mutable</td> </tr> <tr> <td>hostname</td> <td>string</td> <td>DNS resolvable name associated with this network interface. Properties: Mandatory / Mutable</td> </tr> <tr> <td>macAddress</td> <td>string</td> <td>Address assigned by the hypervisor when a machine is created or a unique address can be manually assigned. Properties: Optional / Mutable</td> </tr> <tr> <td>state</td> <td>string</td> <td>The state of an interface configurable to be "Active" or "Standby". Properties: Mandatory / Mutable</td> </tr> <tr> <td>protocol</td> <td>string</td> <td>Selected network protocol such as - IPv4 or IPv6. Properties: Mandatory / Mutable</td> </tr> <tr> <td>allocation</td> <td>string</td> <td>The option for "Dynamic Host Allocation Protocol" or static. Properties: Machine / Mutable</td> </tr> </table>	Name	networkInterface		Attribute	Type	Description	vsp	URI	A reference to the VSP (Virtual Switch Port) for this network interface. Properties: Mandatory / Mutable	hostname	string	DNS resolvable name associated with this network interface. Properties: Mandatory / Mutable	macAddress	string	Address assigned by the hypervisor when a machine is created or a unique address can be manually assigned. Properties: Optional / Mutable	state	string	The state of an interface configurable to be "Active" or "Standby". Properties: Mandatory / Mutable	protocol	string	Selected network protocol such as - IPv4 or IPv6. Properties: Mandatory / Mutable	allocation	string	The option for "Dynamic Host Allocation Protocol" or static. Properties: Machine / Mutable
Name	networkInterface																									
Attribute	Type	Description																								
vsp	URI	A reference to the VSP (Virtual Switch Port) for this network interface. Properties: Mandatory / Mutable																								
hostname	string	DNS resolvable name associated with this network interface. Properties: Mandatory / Mutable																								
macAddress	string	Address assigned by the hypervisor when a machine is created or a unique address can be manually assigned. Properties: Optional / Mutable																								
state	string	The state of an interface configurable to be "Active" or "Standby". Properties: Mandatory / Mutable																								
protocol	string	Selected network protocol such as - IPv4 or IPv6. Properties: Mandatory / Mutable																								
allocation	string	The option for "Dynamic Host Allocation Protocol" or static. Properties: Machine / Mutable																								

		address	string	The IP address assigned to a virtual interface. Properties: Mandatory / Mutable
		defaultGateway	string	An IP address to a firewall or router that serves other networks. Properties: Optional / Mutable
		dns	string	The IP address of the Domain Name Service from host name to IP resolution. Properties: Optional / Mutable
		maxTransmissionUnit	integer	To set the largest supported packet size. Properties: Optional / Mutable
		Properties: Mandatory / Mutable		
meters	ref[]	A list of references to Meters monitored for this Machine. Properties: Optional / Mutable		
eventLog	ref	A reference to the EventLog of this Machine. Properties: Optional / Mutable		

1589 The following describes the serialization of the entity in both JSON and XML:

1590 **JSON media type:** application/CIMI-Machine+json

1591 **JSON serialization:**

```

1592 { "self": string,
1593   "name": string,
1594   "description": string, ?
1595   "created": string, ?
1596   "properties": { "name": string, + }, ?
1597   "state": string,
1598   "networkInterfaces": [
1599     { "network": { "href": string }, "address": string, "hostname", string }, +
1600   ], ?
1601   "cpu": string,
1602   "memory": { "quantity": integer, "units": string },
1603   "disks" : [
1604     { "capacity": { "quantity": integer, "units": string } }, +
1605   ], ?
1606   "volumes" : [
1607     { "volume": { "href": string },
1608       "attachmentPoint": string,
1609       "protocol": string } +
1610   ], ?
1611   "networkInterfaces": [
1612     { "vsp": { "href": string }, "hostname": string, "macAddress": string,
1613       "state": string, "protocol": string, "allocation": string,
1614       "address": string, "defaultGateway": string, "dns": string,
1615       "maxTransmissionUnit": integer }, +

```

```

1616 ], ?
1617 "meters": [
1618   { "href": string }, +
1619 ], ?
1620 "eventLog": { "href": string }, ?
1621 "operations": [
1622   { "rel": "edit", "href": string }, ?
1623   { "rel": "delete", "href": string }, ?
1624   { "rel": "http://www.dmtf.org/cimi/action/start", "href": string }, ?
1625   { "rel": "http://www.dmtf.org/cimi/action/stop", "href": string }, ?
1626   { "rel": "http://www.dmtf.org/cimi/action/restart", "href": string } ?
1627 ]
1628 ...
1629 }

```

1630 **XML media type:** application/CIMI-Machine+xml

1631 **XML serialization:**

```

1632 <Machine xmlns="http://www.dmtf.org/cimi">
1633   <self> xs:anyURI </self>
1634   <name> xs:string </name>
1635   <description> xs:string </description> ?
1636   <created> xs:string </created>
1637   <property name="xs:string"> xs:string </property> *
1638   <state> xs:string </state>
1639   <networkInterface network="xs:anyURI" address="xs:string"
1640     hostname="xs:string"> *
1641   <cpu> xs:string </cpu>
1642   <memory quantity="xs:integer" units="xs:string"/>
1643   <disk>
1644     <capacity quantity="xs:integer" units="xs:string"/>
1645   </disk> *
1646   <volume href="xs:anyURI" attachmentPoint="xs:string" protocol="xs:string"/> *
1647   <networkInterface>
1648     <vsp href="xs:anyURI"/>
1649     <hostname> xs:string </hostname>
1650     <macAddress> xs:string </macAddress>
1651     <state> xs:string </state>
1652     <protocol> xs:string </protocol>
1653     <allocation> xs:string </allocation>
1654     <address> xs:string </address>
1655     <defaultGateway> xs:string </defaultGateway>
1656     <dns> xs:string </dns>
1657     <maxTransmissionUnit> xs:integer </maxTransmissionUnit>
1658   </networkInterface> *
1659   <meter href="xs:anyURI"/> *
1660   <eventLog href="xs:anyURI"/> ?
1661   <operation rel="edit" href="xs:anyURI"/> ?
1662   <operation rel="delete" href="xs:anyURI"/> ?
1663   <operation rel="http://www.dmtf.org/cimi/action/start" href="xs:anyURI"/> ?
1664   <operation rel="http://www.dmtf.org/cimi/action/stop" href="xs:anyURI"/> ?
1665   <operation rel="http://www.dmtf.org/cimi/action/restart" href="xs:anyURI"/> ?
1666   <xs:any>*
1667 </Machine>

```

1668 5.10.7.1 Operations

1669 This entity supports the Read, Update and Delete operations. Create is supported via the Machine
1670 Collection entity.

1671 The following custom operations are also defined:

1672 Starting a Machine

1673 **/link@rel:** http://www.dmtf.org/cimi/action/start

1674 This operation will start a Machine.

1675 Input parameters: None.

1676 Output parameters: None.

1677 Upon successful completion of this operation the Machine will be in the "STARTED" state. Starting a
1678 machine is the virtual equivalent of powering on a physical machine. There is no restored CPU or Memory
1679 state, so the guest OS will typically perform its boot or installation tasks.

1680 • HTTP/REST Protocol

1681 To start a Machine a POST is sent to the "http://www.dmtf.org/cimi/start" URI of the Machine where the
1682 HTTP request body SHALL be as described below.

1683 **JSON media type:** application/CIMI-Action+json

1684 **JSON serialization:**

```
1685 { "action": "http://www.dmtf.org/cimi/action/start" ,  
1686   "properties": { "name": string, + } ?  
1687   ...  
1688 }
```

1689 **XML media type:** application/CIMI-Action+xml

1690 **XML serialization**

```
1691 <Action xmlns="http://www.dmtf.org/cimi">  
1692   <action> http://www.dmtf.org/cimi/action/start </action>  
1693   <property name="xs:string"> xs:string </property> *  
1694   <xs:any>*</xs:any>  
1695 </Action>
```

1696 Upon successful processing of the request, the HTTP response body will be empty.

1697 Stopping a Machine

1698 **/link@rel:** http://www.dmtf.org/cimi/action/stop

1699 This operation will stop, or shutdown, a Machine.

1700 Input parameters: None.

1701 Output parameters: None.

1702 Upon successful completion of this operation the Machine will be in the "STOPPED" state. Stopping a
1703 machine is the virtual equivalent of powering off a physical machine. There is no saved CPU or Memory
1704 state.

1705 • HTTP/REST Protocol

1706 To stop a Machine a POST is sent to the "http://www.dmtf.org/cimi/stop" URI of the Machine where the
1707 HTTP request body SHALL be as described below.

1708 **JSON media type:** application/CIMI-Action+json

1709 **JSON serialization:**

```
1710 { "action": "http://www.dmtf.org/cimi/action/stop" ,
1711     "properties": { "name": string, + } ?
1712     ...
1713 }
```

1714 **XML media type:** application/CIMI-Action+xml

1715 **XML serialization**

```
1716 <Action xmlns="http://www.dmtf.org/cimi">
1717     <action> http://www.dmtf.org/cimi/action/stop </action>
1718     <property name="xs:string"> xs:string </property> *
1719     <xs:any>*
1720 </Action>
```

1721 Upon successful processing of the request, the HTTP response body will be empty.

1722 **Restarting a Machine**

1723 **/link@rel:** http://www.dmtf.org/cimi/action/restart

1724 This operation will restart a Machine. If the Machine is in the "STARTED" state then this operation will
1725 have the semantic effect of executing the "stop" and then "start" operations. If the Machine is in the
1726 "STOPPED" state then this operation will have the semantic effect of executing the "start" operation.

1727 Input parameters: None.

1728 Output parameters: None.

1729 Upon successful completion of this operation the Machine will be in the "STARTED" state. Restarting a
1730 machine is the virtual equivalent of powering off, then on a physical machine. There is no restored CPU
1731 or Memory state, so the guest OS will typically perform it's boot or installation tasks.

1732 • **HTTP/REST Protocol**

1733 To restart a Machine a POST is sent to the "http://www.dmtf.org/cimi/restart" URI of the Machine where
1734 the HTTP request body SHALL be as described below.

1735 **JSON media type:** application/CIMI-Action+json

1736 **JSON serialization:**

```
1737 { "action": "http://www.dmtf.org/cimi/action/restart" ,
1738     "properties": { "name": string, + } ?
1739     ...
1740 }
```

1741 **XML media type:** application/CIMI-Action+xml

1742 **XML serialization**

```
1743 <Action xmlns="http://www.dmtf.org/cimi">
1744     <action> http://www.dmtf.org/cimi/action/restart </action>
1745     <property name="xs:string"> xs:string </property> *
1746     <xs:any>*
1747 </Action>
```

1748 Upon successful processing of the request, the HTTP response body will be empty.

1749 **5.10.8 Machine Collection**

1750 A Machine Collection resource represents the collection of Machine entities within a Provider. This
 1751 resource can be used to locate and create Machines.

Name	MachineCollection	
Type URI	http://www.dmtf.org/cimi/MachineCollection	
Attribute	Type	Description
machines	ref[]	An array of references to the set of Machines in the Provider. Properties: Optional / Mutable

1752 The following describes the serialization of the entity in both JSON and XML:

1753 **JSON media type:** application/CIMI-MachineCollection+json

1754 **JSON serialization:**

```

1755 { "self": string,
1756   "name": string, ?
1757   "description": string, ?
1758   "created": string, ?
1759   "properties": { "name": string, + }, ?
1760   "machines": [
1761     { "href": string }, +
1762   ], ?
1763   "operations": [
1764     { "rel": "add", "href": string }, ?
1765     { "rel": "edit", "href": string } ?
1766   ] ?
1767   ...
1768 }
```

1769 **XML media type:** application/CIMI-MachineCollection+xml

1770 **XML serialization:**

```

1771 <MachineCollection xmlns="http://www.dmtf.org/cimi">
1772   <self> xs:anyURI </self>
1773   <name> xs:string </name> ?
1774   <description> xs:string </description> ?
1775   <created> xs:string </created>
1776   <property name="xs:string"> xs:string </property> *
1777   <machine href="xs:anyURI"/> *
1778   <operation rel="add" href="xs:anyURI"/> ?
1779   <operation rel="edit" href="xs:anyURI"/> ?
1780   <xs:any>*
1781 </MachineCollection>
```

1782 **5.10.8.1 Operations**

1783 This entity supports the Read and Update operations.

1784 The following custom operations are also defined:

1785 **• Creating a New Machine**

1786 **/link@rel:** add

1787 This operation will create a new Machine.

1788 Input parameters: Either a reference to a Machine Template or a Machine Template itself.

1789 Output parameters: A reference to a new Machine and optionally the representation of the Machine.

1790 • **HTTP/REST Protocol**

1791 To create a new Machine a POST is sent to the "add" URI of the MachineCollection where the HTTP
 1792 request body SHALL be as described below. Note this structure allows for certain properties to be
 1793 passed in "by value" or by "reference". The definition of each property can be found in section 5.10.1.

1794 **JSON media type:** application/CIMI-MachineCreate+json

1795 **JSON serialization:**

```

1796 { "name": string,
1797   "description": string, ?
1798   "properties": { "name": string, + }, ?
1799   "machineTemplate": { "href": string, ?
1800     "properties": { "name": string, + }, ?
1801     "machineConfig": { "href": string, ?
1802       "cpu": string, ?
1803       "memory": { "quantity": integer, "units": string }, ?
1804       "disks" : [
1805         { "capacity": { "quantity": integer, "units": string },
1806           "guestInterface": string }, +
1807       ] ?
1808     },
1809     "machineImage": { "href": string,
1810       "imageLocation": { "href": string }, ?
1811       "imageData": string, ?
1812     }, ?
1813     "machineAdmin": { "href": string, ?
1814       <provider specific data> ?
1815     }, ?
1816     "volumes": [
1817       { "href": string, "attachmentPoint": string, "protocol": string}, +
1818     ], ?
1819     "volumeTemplates": [
1820       { "href": string, ?
1821         "attachmentPoint": string, "protocol": string,
1822         "volumeConfig": { "href": string, ?
1823           "properties": { "name": string, + }, ?
1824           "format": string, ?
1825           "capacity": { "quantity": number, "units": string }, ?
1826           "supportsSnapshots": boolean, ?
1827           "guestInterface": string ?
1828         }, ?
1829         "volumeImage": { "href": string,
1830           "properties": { "name": string, + }, ?
1831           "imageLocation": { "href": string }, ?
1832           "imageData": string, ?
1833           "bootable": boolean ?
1834         } ?
1835       }, +
1836     ]
1837   }
1838   ...
1839 }
```

1840 **XML media type:** application/CIMI-MachineCreate+xml

1841 **XML serialization**

```

1842 <MachineCreate>
1843   <name> xs:string </name>
1844   <description> xs:string </description> ?
1845   <property name="xs:string"> xs:string </property> *
1846
1847   <machineTemplate href="xs:anyURI"? >
1848     <property name="xs:string"> xs:string </property> *
1849     <machineConfig href="xs:anyURI"? >
1850       <property name="xs:string"> xs:string </property> *
1851       <cpu> xs:string </cpu> ?
1852       <memory quantity="xs:integer" units="xs:string"/>
1853       <disk>
1854         <capacity quantity="xs:integer" units="xs:string">
1855           <guestInterface> xs:string </guestInterface>
1856         </disk> *
1857     </machineConfig>
1858     <machineImage href="xs:anyURI">
1859       <property name="xs:string"> xs:string </property> *
1860       <imageLocation href="xs:anyURI"/> ?
1861       <imageData> xs:string </imageData> ?
1862     </machineImage>
1863     <machineAdmin href="xs:anyURI"? >
1864       xs:any* <!-- provider specific data -->
1865     </machineAdmin> ?
1866     <volume href="xs:anyURI"
1867       attachmentPoint="xs:string" protocol="xs:string" /> *
1868     <volumeTemplate href="xs:anyURI"?
1869       attachmentPoint="xs:string" protocol="xs:string" >
1870       <property name="xs:string"> xs:string </property> *
1871       <volumeConfig href="xs:anyURI">
1872         <property name="xs:string"> xs:string </property> *
1873         <format> xs:string </format> ?
1874         <capacity quantity="xs:integer" units="xs:string"/> ?
1875         <supportsSnapshots> xs:boolean </supportsSnapshots> ?
1876         <guestInterface> xs:string </guestInterface> ?
1877       </volumeConfig>
1878       <volumeImage href="xs:anyURI">
1879         <property name="xs:string"> xs:string </property> *
1880         <imageLocation href="xs:anyURI"/> ?
1881         <imageData> xs:any* </imageData> ?
1882         <bootable> xs:boolean </bootable> ?
1883       </volumeImage> ?
1884     </volumeTemplate> *
1885   </machineTemplate>
1886
1887   <xs:any>*
1888
1889 </MachineCreate>

```

1890 The serialization of some reference properties are specified such that a request MAY either include a
 1891 reference ("href") to an existing entity or to include the entity "inline" as a set of additional properties.
 1892 Requests SHALL NOT include both a reference and the inlined set of properties.

1893 Upon successful processing of the request, the HTTP response body MAY either be empty or contain a
 1894 serialization of the Machine entity.

1895 **5.10.9 Machine Admin**

1896 A Machine Admin entity contains the information required to create the initial administrative super- user of
 1897 a newly created Machine. Due to the variation between operating systems and Providers, this
 1898 specification does not manadate one particular set of attributes that all implementations need to support.
 1899 However, Providers are expected to extend this entity with additional attributes to meet their
 1900 requirements.

1901 For example, a Provider might extend this entity with username and password attributes, which would
 1902 then be the login information for new Machines. These extension attributes would appear as siblings to
 1903 the common attributes like 'name' and 'description'.

Name	MachineAdmin	
Type URI	http://www.dmtf.org/cimi/MachineAdmin	
Attribute	Type	Description

1904 Some common extension attributes that Providers might use include:

1905 **UserName/Password:**

Attribute	Type	Description
userName	string	The initial superuser's user name. Properties: Mandatory / Mutable
password	string	Initial superuser's password. Properties: Mandatory / Mutable

1906 **Public Key:**

Attribute	Type	Description
key	byte[]	The digit of the public key for the initial superuser. Properties: Mandatory / Mutable

1907 **JSON media type:** application/CIMI-MachineAdmin+json

1908 **JSON serialization:**

```

1909 { "self": string,
1910   "name": string, ?
1911   "description": string, ?
1912   "created": string, ?
1913   "properties": { "name": string, + }, ?
1914   "operations": [
1915     { "rel": "edit", "href": string } ?
1916     { "rel": "delete", "href": string } ?
1917   ] ?
1918   ...
1919 }
```


1920 **XML media type:** application/CIMI-MachineAdmin+xml

1921 **XML serialization:**

```

1922 <MachineAdmin xmlns="http://www.dmtf.org/cimi">
1923   <self> xs:anyURI </self>
1924   <name> xs:string </name> ?
1925   <description> xs:string </description> ?
1926   <created> xs:string </created>
1927   <property name="xs:string"> xs:string </property> *
1928   <operation rel="edit" href="xs:anyURI"/> ?
1929   <operation rel="delete" href="xs:anyURI"/> ?
1930   <xs:any>*
1931 </MachineAdmin>
    
```

1932 **5.10.9.1 Operations**

1933 This entity supports the Read, Update and Delete operations. Create is supported via the Machine Admin
 1934 Collection entity.

1935 **5.10.10 Machine Admin Collection**

1936 A Machine Admin Collection entity represents the collection of Machine Admin entities within a Provider.
 1937 This entity can be used to locate and create MachineAdmins.

Name	MachineAdminCollection	
Type URI	http://www.dmtf.org/cimi/MachineAdminCollection	
Attribute	Type	Description
machineAdmins	ref[]	An array of references to the set of Machine Admins in the provider. Properties: Optional / Mutable

1938 The following describes the serialization of the entity in both JSON and XML:

1939 **JSON media type:** application/CIMI-MachineAdminCollection+json

1940 **JSON serialization:**

```

1941 { "self": string,
1942   "name": string, ?
1943   "description": string, ?
1944   "created": string, ?
1945   "properties": { "name": string, + }, ?
1946   "machineAdmins": [
1947     { "href": string }, +
1948   ], ?
1949   "operations": [
1950     { "rel": "add", "href": string }, ?
1951     { "rel": "edit", "href": string } ?
1952   ] ?
1953   ...
1954 }
    
```

1955 **XML media type:** application/CIMI-MachineAdminCollection+xml

1956 **XML serialization:**

```

1957 <MachineAdminCollection xmlns="http://www.dmtf.org/cimi">
1958   <self> xs:anyURI </self>
    
```

```

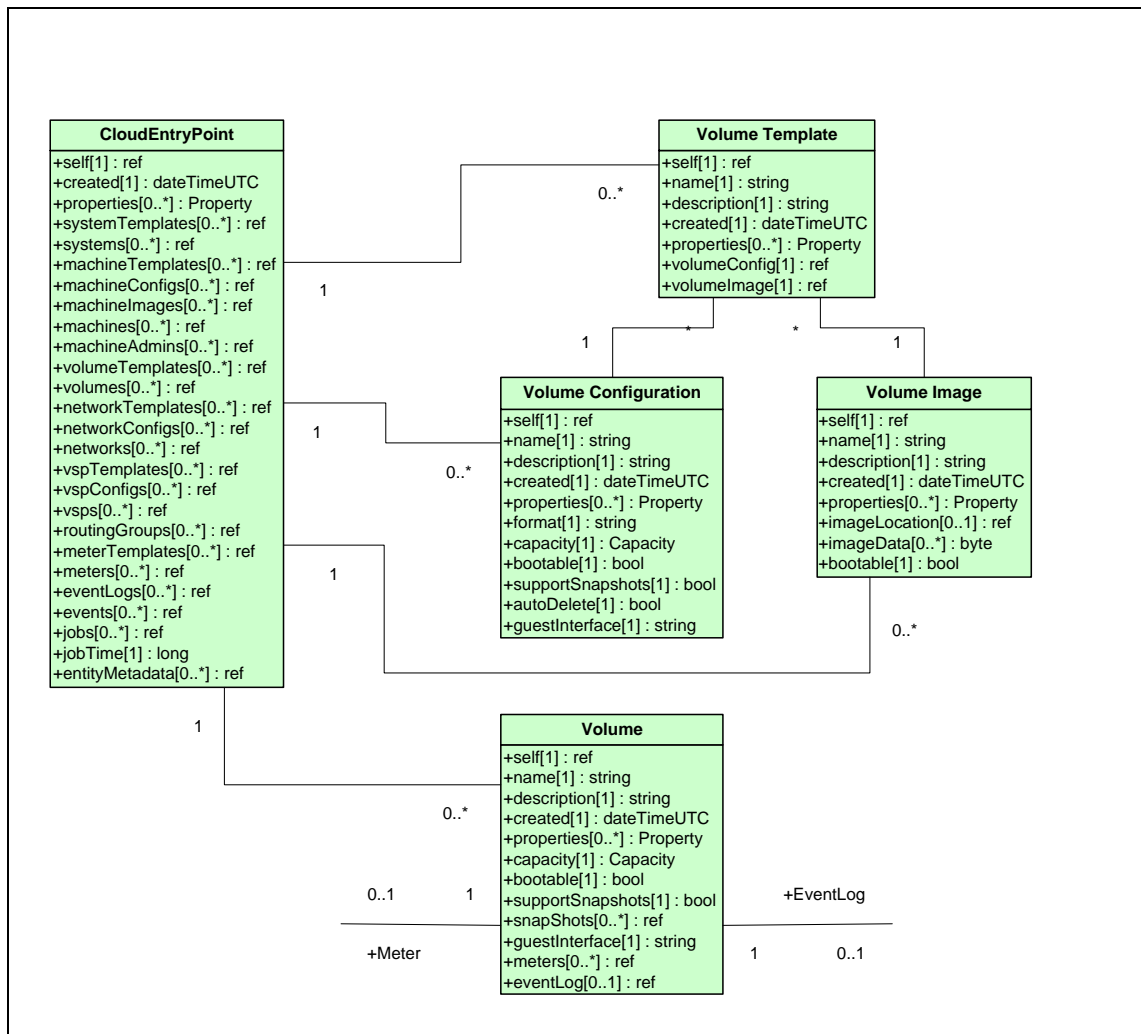
1959 <name> xs:string </name> ?
1960 <description> xs:string </description> ?
1961 <created> xs:string </created>
1962 <property name="xs:string"> xs:string </property> *
1963 <machineAdmin href="xs:anyURI"/> *
1964 <operation rel="add" href="xs:anyURI"/> ?
1965 <operation rel="edit" href="xs:anyURI"/> ?
1966 <xs:any>*
1967 </MachineAdminCollection>
    
```

1968 **5.10.10.1 Operations**

1969 This entity supports the Read and Update operations. Creation of new Machine Admin entities is
 1970 supported via a POST to the "addLink" URI as described in section 4.2.2.1.

1971 **5.11 Volume Entities and Relationships**

1972 The following diagram illustrates the entities involved in constructing a Volume and their relationships.
 1973 Although this drawing is in the style of an Entity Relationship diagram, the use of UML is neither rigorous
 1974 nor normative.



1975
 1976 **Figure 3 - Volume Entities**

1977 **5.11.1 Volume Template**

1978 This entity captures the configuration values for realizing a Volume. A Volume Template may be used to
1979 create multiple Volumes.

Name	VolumeTemplate	
Type URI	http://www.dmtf.org/cimi/VolumeTemplate	
Attribute	Type	Description
volumeConfig	ref	A reference to the Volume Configuration that will be used to create a Volume from this Volume Template. Properties: Mandatory / Mutable
volumeImage	ref	A reference to the Volume Image that will be used to create a Volume from this Volume Template. Properties: Optional / Mutable

1980 The following describes the serialization of the entity in both JSON and XML:

1981 **JSON media type:** application/CIMI-VolumeTemplate+json

1982 **JSON serialization:**

```
1983 { "self": string,
1984   "name": string, ?
1985   "description": string, ?
1986   "created": string, ?
1987   "properties": { "name": string, + }, ?
1988   "volumeConfig": { "href": string },
1989   "volumeImage": { "href": string },
1990   "operations": [
1991     { "rel": "edit", "href": string }, ?
1992     { "rel": "delete", "href": string } ?
1993   ] ?
1994   ...
1995 }
```

1996 **XML media type:** application/CIMI-VolumeTemplate+xml

1997 **XML serialization:**

```
1998 <VolumeTemplate xmlns="http://www.dmtf.org/cimi">
1999   <self> xs:anyURI </self>
2000   <name> xs:string </name> ?
2001   <description> xs:string </description> ?
2002   <created> xs:string </created>
2003   <property name="xs:string"> xs:string </property> *
2004   <volumeConfig href="xs:anyURI"/>
2005   <volumeImage href="xs:anyURI"/>
2006   <operation rel="edit" href="xs:anyURI"/> ?
2007   <operation rel="delete" href="xs:anyURI"/> ?
2008   <xs:any>*
2009 </VolumeTemplate>
```

2010 **5.11.1.1 Operations**

2011 This entity supports the Read, Update and Delete operations. Create is supported via the Volume
2012 Template Collection entity.

2013 5.11.2 Volume Template Collection

2014 A Volume Template Collection entity represents the collection of VolumeTemplate entities within a
2015 Provider. This entity can be used to locate and create Volume Templates.

Name	VolumeTemplateCollection	
Type URI	http://www.dmtf.org/cimi/VolumeTemplateCollection	
Attribute	Type	Description
volumeTemplates	ref[]	An array of references to the set of Volume Templates in the Provider. Properties: Optional / Mutable

2016 The following describes the serialization of the entity in both JSON and XML:

2017 **JSON media type:** application/CIMI-VolumeTemplateCollection+json

2018 **JSON serialization:**

```
2019 { "self": string,
2020   "name": string, ?
2021   "description": string, ?
2022   "created": string, ?
2023   "properties": { "name": string, + }, ?
2024   "volumeTemplates": [
2025     { "href": string }, +
2026   ], ?
2027   "operations": [
2028     { "rel": "add", "href": string }, ?
2029     { "rel": "edit", "href": string } ?
2030   ] ?
2031   ...
2032 }
```

2033 **XML media type:** application/CIMI-VolumeTemplateCollection+xml

2034 **XML serialization:**

```
2035 <VolumeTemplateCollection xmlns="http://www.dmtf.org/cimi">
2036   <self> xs:anyURI </self>
2037   <name> xs:string </name> ?
2038   <description> xs:string </description> ?
2039   <created> xs:string </created>
2040   <property name="xs:string"> xs:string </property> *
2041   <volumeTemplate href="xs:anyURI"/> *
2042   <operation rel="add" href="xs:anyURI"/> ?
2043   <operation rel="edit" href="xs:anyURI"/> ?
2044   <xs:any>*
2045 </VolumeTemplateCollection>
```

2046 5.11.2.1 Operations

2047 This entity supports the Read and Update operations. Creation of new Volume Template entities is
2048 supported via a POST to the "addLink" URI as described in section 4.2.2.1.

2049 5.11.3 Volume Configuration

2050 The Volume Configuration entity represents the set of configuration values needed to create a Volume
2051 with certain characteristics. Volume Configurations are created by Providers and MAY, at the Providers
2052 discretion, be created by Consumers.

Name	VolumeConfiguration										
Type URI	http://www.dmtf.org/cimi/VolumeConfiguration										
Attribute	Type	Description									
format	String	The format of the file system that will be placed on Volumes created from this configuration. This attribute is only meaningful for Volume Configurations that describe block devices. This attribute is optional; the absence of this attribute indicates that Volumes created from this configuration will not be formatted with a file system. Example values: "ext4", "ntfs". Properties: Mandatory / Mutable									
capacity	structure	The default size, when limited, of the Volume created from this Volume Configuration. This attribute has the following, sub-attributes. <table border="1" data-bbox="597 762 1429 1192"> <thead> <tr> <th>Attribute</th> <th>Type</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>quantity</td> <td>integer</td> <td>A numerical quantity expressed as an integer. Properties: Mandatory / Mutable</td> </tr> <tr> <td>units</td> <td>String</td> <td>An enumerated value that expresses the unit of measurement used. Allowable values are byte, kilobyte, megabyte, gigabyte, terabyte, petabyte, exabyte, zettabyte, and yottabyte. Properties: Mandatory / Mutable</td> </tr> </tbody> </table> Properties: Mandatory / Mutable	Attribute	Type	Description	quantity	integer	A numerical quantity expressed as an integer. Properties: Mandatory / Mutable	units	String	An enumerated value that expresses the unit of measurement used. Allowable values are byte, kilobyte, megabyte, gigabyte, terabyte, petabyte, exabyte, zettabyte, and yottabyte . Properties: Mandatory / Mutable
Attribute	Type	Description									
quantity	integer	A numerical quantity expressed as an integer. Properties: Mandatory / Mutable									
units	String	An enumerated value that expresses the unit of measurement used. Allowable values are byte, kilobyte, megabyte, gigabyte, terabyte, petabyte, exabyte, zettabyte, and yottabyte . Properties: Mandatory / Mutable									
supportsSnapshots	boolean	This property indicates whether Volumes created from this Volume Configuration will support the ability to take snapshots. Properties: Mandatory / Mutable									
guestInterface	String	This property indicates the interface that will be offered to a Machine instances by Volumes created from this Volume Configuration. Properties: Mandatory / Mutable									

2053 The following describes the serialization of the entity in both JSON and XML:

2054 **JSON media type:** application/CIMI-VolumeConfiguration+json

2055 **JSON serialization:**

```

2056 { "self": string,
2057   "name": string, ?
2058   "description": string, ?
2059   "created": string, ?
2060   "properties": { "name": string, + }, ?
2061   "format": string,
2062   "capacity": { "quantity": number, "units": string },

```

```

2063     "supportsSnapshots": boolean,
2064     "guestInterface": string,
2065     "operations": [
2066       { "rel": "edit", "href": string }, ?
2067       { "rel": "delete", "href": string } ?
2068     ] ?
2069     ...
2070   }

```

2071 **XML media type:** application/CIMI-VolumeConfiguration+xml

2072 **XML serialization:**

```

2073 <VolumeConfiguration xmlns="http://www.dmtf.org/cimi">
2074   <self> xs:anyURI </self>
2075   <name> xs:string </name> ?
2076   <description> xs:string </description> ?
2077   <created> xs:string </created>
2078   <property name="xs:string"> xs:string </property> *
2079   <format> xs:string </format>
2080   <capacity quantity="xs:integer" units="xs:string"/>
2081   <supportsSnapshots> xs:boolean </supportsSnapshots>
2082   <guestInterface> xs:string </guestInterface>
2083   <operation rel="edit" href="xs:anyURI"/> ?
2084   <operation rel="delete" href="xs:anyURI"/> ?
2085   <xs:any>*
2086 </VolumeConfiguration>

```

2087 **5.11.3.1 Operations**

2088 This entity supports the Read, Update and Delete operations. Create is supported via the Volume
 2089 Configuration Collection entity.

2090 **5.11.4 Volume Configuration Collection**

2091 A Volume Configuration Collection entity represents the collection of Volume Configuration entities within
 2092 a Provider. This entity can be used to locate and create Volume Configurations.

Name	VolumeConfigurationCollection	
Type URI	http://www.dmtf.org/cimi/VolumeConfigurationCollection	
Attribute	Type	Description
volumeConfigurations	ref[]	An array of references to the set of Volume Configurations in the Provider. Properties: Optional / Mutable

2093 The following describes the serialization of the entity in both JSON and XML:

2094 **JSON media type:** application/CIMI-VolumeConfigurationCollection+json

2095 **JSON serialization:**

```

2096 { "self": string,
2097   "name": string, ?
2098   "description": string, ?
2099   "created": string, ?
2100   "properties": { "name": string, + }, ?
2101   "volumeConfigurations": [
2102     { "href": string }, +

```

```

2103     ], ?
2104     "operations": [
2105         { "rel": "add", "href": string }, ?
2106         { "rel": "edit", "href": string } ?
2107     ] ?
2108     ...
2109 }

```

2110 **XML media type:** application/CIMI-VolumeConfigurationCollection+xml

2111 **XML serialization:**

```

2112 <VolumeConfigurationCollection xmlns="http://www.dmtf.org/cimi">
2113   <self> xs:anyURI </self>
2114   <name> xs:string </name> ?
2115   <description> xs:string </description> ?
2116   <created> xs:string </created>
2117   <property name="xs:string"> xs:string </property> *
2118   <volumeConfiguration href="xs:anyURI"/> *
2119   <operation rel="add" href="xs:anyURI"/> ?
2120   <operation rel="edit" href="xs:anyURI"/> ?
2121   <xs:any>*
2122 </VolumeConfigurationCollection>

```

2123 5.11.4.1 Operations

2124 This entity supports the Read and Update operations. Creation of new Volume Image entities is
 2125 supported via a POST to the "addLink" URI as described in section 4.2.2.1.

2126 5.11.5 Volume Image

2127 This entity represents an image that could be place on a pre-loaded volume.

Name	VolumelImage	
Type URI	http://www.dmtf.org/cimi/VolumelImage	
Attribute	Type	Description
imageLocation	ref	A reference to the location of the binary data that makes up this image. Either this attribute or imageData SHALL be present; however, both values SHALL NOT be present simultaneously. Properties: Optional / Immutable
imageData	byte[]	The binary data that makes up this image. Either this attribute or imageLocation SHALL be present; however, both values SHALL NOT be present simultaneously. Properties: Optional / Immutable
bootable	boolean	This property indicates whether Volumes created from this Volume Configuration will be bootable. Properties: Mandatory / Mutable

2128 The following describes the serialization of the entity in both JSON and XML:

2129 **JSON media type:** application/CIMI-VolumeImage+json

2130 **JSON serialization:**

```

2131 { "self": string,
2132     "name": string, ?
2133     "description": string, ?
2134     "created": string, ?
2135     "properties": { "name": string, + }, ?
2136     "imageLocation": { "href": string }, ?
2137     "imageData": string, ?
2138     "bootable": boolean,
2139     "operations": [
2140         { "rel": "edit", "href": string }, ?
2141         { "rel": "delete", "href": string } ?
2142     ] ?
2143     ...
2144 }
```

2145 **XML media type:** application/CIMI-VolumeImage+xml

2146 **XML serialization:**

```

2147 <VolumeImage xmlns="http://www.dmtf.org/cimi">
2148   <self> xs:anyURI </self>
2149   <name> xs:string </name> ?
2150   <description> xs:string </description> ?
2151   <created> xs:string </created>
2152   <property name="xs:string"> xs:string </property> *
2153   <imageLocation href="xs:anyURI"/> ?
2154   <imageData> xs:any* </imageData> ?
2155   <bootable> xs:boolean </bootable>
2156   <operation rel="edit" href="xs:anyURI"/> ?
2157   <operation rel="delete" href="xs:anyURI"/> ?
2158   <xs:any*>
2159 </VolumeImage>
```

2160 **5.11.5.1 Operations**

2161 This entity supports the Read, Update and Delete operations. Create is supported via the Volume Image
 2162 Collection entity.

2163 **5.11.6 Volume Image Collection**

2164 A Volume Image Collection entity represents the collection of VolumeImage entities within a Provider.
 2165 This entity can be used to locate and create Volume Images.

Name	VolumeImageCollection	
Type URI	http://www.dmtf.org/cimi/VolumeImageCollection	
Attribute	Type	Description
volumeImages	ref[]	An array of references to the set of Volume Images in the Provider. Properties: Optional / Mutable

2166 The following describes the serialization of the entity in both JSON and XML:

2167 **JSON media type:** application/CIMI-VolumeImageCollection+json

2168 **JSON serialization:**

```

2169 { "self": string,
2170   "name": string, ?
2171   "description": string, ?
2172   "created": string, ?
2173   "properties": { "name": string, + }, ?
2174   "volumeImages": [
2175     { "href": string }, +
2176   ], ?
2177   "operations": [
2178     { "rel": "add", "href": string }, ?
2179     { "rel": "edit", "href": string } ?
2180   ] ?
2181   ...
2182 }
```

2183 **XML media type:** application/CIMI-VolumeImageCollection+xml

2184 **XML serialization:**

```

2185 <VolumeImageCollection xmlns="http://www.dmtf.org/cimi">
2186   <self> xs:anyURI </self>
2187   <name> xs:string </name> ?
2188   <description> xs:string </description> ?
2189   <created> xs:string </created>
2190   <property name="xs:string"> xs:string </property> *
2191   <volumeImage href="xs:anyURI"/> *
2192   <operation rel="add" href="xs:anyURI"/> ?
2193   <operation rel="edit" href="xs:anyURI"/> ?
2194   <xs:any>*
2195 </VolumeImageCollection>
```

2196 **5.11.6.1 Operations**

2197 This entity supports the Read and Update operations. Creation of new Volume Image entities is
 2198 supported via a POST to the "addLink" URI as described in section 4.2.2.1.

2199 **5.11.7 Volume**

2200 A Volume represents storage at either the block or file-system level. Volumes can be attached to
 2201 Machines. Once attached, Volumes can be accessed by processes on that Machine.

Name	Volume		
Type URI	http://www.dmtf.org/cimi/Volume		
Attribute	Type	Description	
capacity	structure	The maximum size, when limited, of the Volume.	
		When this value is increased, the Volume can contain more data. Decreasing this value may require evaluations.	
		This attribute has the following, sub-attributes.	
	Attribute	Type	Description
	quantity	integer	A numerical quantity expressed as an

		integer. Properties: Mandatory / Mutable
	units	string An enumerated value that expresses the unit of measurement used. Allowable values are byte, kilobyte, megabyte, gigabyte, terabyte, petabyte, exabyte, zettabyte, and yottabyte. Properties: Mandatory / Mutable
		Properties: Mandatory / Mutable
bootable	boolean	This property indicates whether this Volume is bootable. Properties: Mandatory / Mutable
supportsSnapshots	boolean	This property indicates whether the Volume supports the ability to take snapshots. Properties: Mandatory / Mutable
snapShots	ref[]	A list of references to Volume Images that represent snapshots taken from the Volume. Properties: Optional / Mutable
guestInterface	String	This property indicates the interface offered to a Machine instance to gain access to the storage contents. Properties: Mandatory / Mutable
meters	ref[]	A list of references to Meters monitored for this Volume. Properties: Optional / Mutable
eventLog	ref	A reference to the EventLog of this Volume. Properties: Optional / Mutable

2202 The following describes the serialization of the entity in both JSON and XML:

2203 **JSON media type:** application/CIMI-Volume+json

2204 **JSON serialization:**

```

2205 { "self": string,
2206   "name": string, ?
2207   "description": string, ?
2208   "created": string, ?
2209   "properties": { "name": string, + }, ?
2210   "capacity": { "quantity": number, "units": string },
2211   "bootable": boolean,
2212   "supportsSnapshots": boolean,
2213   "snapshots": [
2214     { "href": string }, +
2215   ], ?
2216   "guestInterface": string,
2217   "meters": [
2218     { "href": string }, +

```

```

2219     ], ?
2220     "eventLog": { "href": string }, ?
2221     "operations": [
2222       { "rel": "edit", "href": string }, ?
2223       { "rel": "delete", "href": string } ?
2224     ] ?
2225     ...
2226   }

```

2227 **XML media type:** application/CIMI-Volume+xml

2228 **XML serialization:**

```

2229 <Volume xmlns="http://www.dmtf.org/cimi">
2230   <self> xs:anyURI </self>
2231   <name> xs:string </name> ?
2232   <description> xs:string </description> ?
2233   <created> xs:string </created>
2234   <property name="xs:string"> xs:string </property> *
2235   <capacity quantity="xs:integer" units="xs:string"/>
2236   <bootable> xs:boolean </bootable>
2237   <supportsSnapshots> xs:boolean </supportsSnapshots>
2238   <shapshot href="xs:anyURI"/> *
2239   <guestInterface> xs:string </guestInterface>
2240   <meter href="xs:anyURI"/> *
2241   <eventLog href="xs:anyURI"/> ?
2242   <operation rel="edit" href="xs:anyURI"/> ?
2243   <operation rel="delete" href="xs:anyURI"/> ?
2244   <xs:any*>
2245 </Volume>

```

2246 5.11.7.1 Operations

2247 This entity supports the Read, Update and Delete operations. Create is supported via the Volume
2248 Collection entity.

2249 5.11.8 Volume Collection

2250 A Volume Collection entity represents the collection of Volumes within a Provider. This entity can be used
2251 to locate and create Volumes.

Name	VolumeCollection	
Type URI	http://www.dmtf.org/cimi/VolumeCollection	
Attribute	Type	Description
volumes	ref[]	An array of references to the set of Volumes in the provider. Properties: Optional / Mutable

2252 The following describes the serialization of the entity in both JSON and XML:

2253 **JSON media type:** application/CIMI-VolumeCollection+json

2254 **JSON serialization:**

```

2255 { "self": string,
2256   "name": string, ?
2257   "description": string, ?
2258   "created": string, ?
2259   "properties": { "name": string, + }, ?

```

```

2260     "volumes": [
2261         { "href": string }, +
2262     ], ?
2263     "operations": [
2264         { "rel": "add", "href": string }, ?
2265         { "rel": "edit", "href": string } ?
2266     ] ?
2267     ...
2268 }

```

2269 **XML media type:** application/CIMI-VolumeCollection+xml

2270 **XML serialization:**

```

2271 <VolumeCollection xmlns="http://www.dmtf.org/cimi">
2272   <self> xs:anyURI </self>
2273   <name> xs:string </name> ?
2274   <description> xs:string </description> ?
2275   <created> xs:string </created>
2276   <property name="xs:string"> xs:string </property> *
2277   <volume href="xs:anyURI"/> *
2278   <operation rel="add" href="xs:anyURI"/> ?
2279   <operation rel="edit" href="xs:anyURI"/> ?
2280   <xs:any>*
2281 </VolumeCollection>

```

2282 5.11.8.1 Operations

2283 This entity supports the Read and Update operations.

2284 The following custom operations are also defined:

2285 Creating a New Volume

2286 **/link@rel:** add

2287 This operation will create a new Volume.

2288 Input parameters: Either a reference to a Volume Template or a Volume Template itself.

2289 Output parameters: A reference to a new Volume and optionally the representation of the Volume.

2290 • HTTP/REST Protocol

2291 To create a new Volume a POST is sent to the "add" URI of the VolumeCollection where the HTTP
 2292 request body SHALL be as described below. Note this structure allows for certain properties to be
 2293 passed in "by value" or by "reference". The definition of each property can be found in section 5.11.1.

2294 **JSON media type:** application/CIMI-VolumeCreate+json

2295 **JSON serialization:**

```

2296 { "name": string,
2297   "description": string, ?
2298   "properties": { "name": string, + }, ?
2299   "volumeTemplate": { "href": string, ?
2300     "properties": { "name": string, + }, ?
2301     "volumeConfig": { "href": string, ?
2302       "properties": { "name": string, + }, ?
2303       "format": string, ?
2304       "capacity": { "quantity": number, "units": string }, ?
2305       "supportsSnapshots": boolean, ?
2306       "guestInterface": string, ?

```

```

2307     },
2308     "volumeImage": { "href": string,
2309       "imageLocation": { "href": string }, ?
2310       "imageData": string, ?
2311       "bootable": boolean ?
2312     } ?
2313   }
2314   ...
2315 }

```

2316 **XML media type:** application/CIMI-VolumeCreate+xml

2317 **XML serialization**

```

2318 <VolumeCreate>
2319   <name> xs:string </name>
2320   <description> xs:string </description> ?
2321   <property name="xs:string"> xs:string </property> *
2322   <volumeTemplate href="xs:anyURI"? >
2323     <property name="xs:string"> xs:string </property> *
2324     <volumeConfig href="xs:anyURI"? >
2325       <property name="xs:string"> xs:string </property> *
2326       <format> xs:string </format> ?
2327       <capacity quantity="xs:integer" units="xs:string"/> ?
2328       <supportsSnapshots> xs:boolean </supportsSnapshots> ?
2329       <guestInterface> xs:string </guestInterface> ?
2330     </volumeConfig>
2331     <volumeImage href="xs:anyURI">
2332       <property name="xs:string"> xs:string </property> *
2333       <imageLocation href="xs:anyURI"/> ?
2334       <imageData> xs:any* </imageData> ?
2335       <bootable> xs:boolean </bootable> ?
2336     </volumeImage> ?
2337   </volumeTemplate>
2338   <xs:any>*
2339 </VolumeCreate>

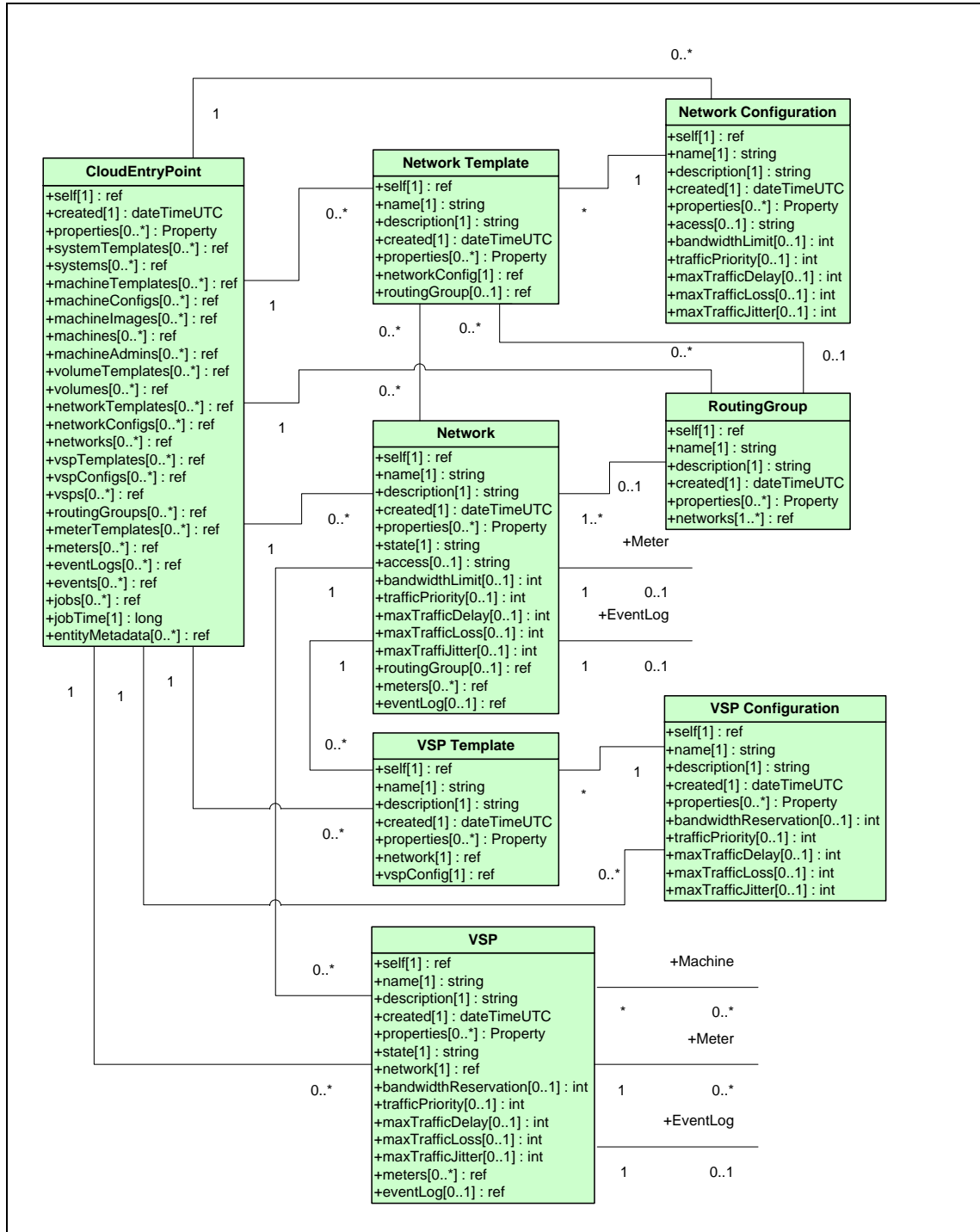
```

2340 The serialization of some reference properties are specified such that a request MAY either include a
 2341 reference ("href") to an existing entity or to include the entity "inline" as a set of additional properties.
 2342 Requests SHALL NOT include both a reference and the inlined set of properties.

2343 Upon successful processing of the request, the HTTP response body MAY either be empty or contain a
 2344 serialization of the Volume entity.

2345 5.12 Network Entities and Relationships

2346 The following diagram illustrates the entities involved in constructing Networks and their Virtual Switch
 2347 Ports (VSPs) and their relationships. Although this drawing is in the style of an Entity Relationship
 2348 diagram, the use of UML is neither rigorous nor normative.



2349

2350 **Figure 4 - Network Entities**

2351 **5.12.1 Network Template**

2352 The Network Template is a set of configuration values for realizing a Network. An instance of Network
 2353 Template may be used to create multiple Networks.

Name	NetworkTemplate
-------------	-----------------

Type URI	http://www.dmtf.org/cimi/NetworkTemplate	
Attribute	Type	Description
networkConfig	ref	A reference to the Network Configuration that will be used to create a Network from this Network Template. Properties: Mandatory / Mutable
routingGroup	ref	A reference to a RoutingGroup that the net Network will be part of. Note that Networks route to themselves, therefore this attribute will only appear in cases where the Network that will be created from this template routes to one or more additional Networks. Properties: Optional / Mutable

2354 The following describes the serialization of the entity in both JSON and XML:

2355 **JSON media type:** application/CIMI-NetworkTemplate+json

2356 **JSON serialization:**

```

2357 { "self": string,
2358   "name": string, ?
2359   "description": string, ?
2360   "created": string, ?
2361   "properties": { "name": string, + }, ?
2362   "networkConfig": { "href": string },
2363   "routingGroup": { "href": string }, ?
2364   "operations": [
2365     { "rel": "edit", "href": string }, ?
2366     { "rel": "delete", "href": string } ?
2367   ] ?
2368   ...
2369 }
```

2370 **XML media type:** application/CIMI-NetworkTemplate+xml

2371 **XML serialization:**

```

2372 <NetworkTemplate xmlns="http://www.dmtf.org/cimi">
2373   <self> xs:anyURI </self>
2374   <name> xs:string </name> ?
2375   <description> xs:string </description> ?
2376   <created> xs:string </created>
2377   <property name="xs:string"> xs:string </property> *
2378   <networkConfig href="xs:anyURI"/>
2379   <routingGroup href="xs:anyURI"/> ?
2380   <operation rel="edit" href="xs:anyURI"/> ?
2381   <operation rel="delete" href="xs:anyURI"/> ?
2382   <xs:any>*
2383 </NetworkTemplate>
```

2384 5.12.1.1 Operations

2385 This entity supports the Read, Update and Delete operations. Create is supported via the Network
2386 Template Collection entity.

2387 5.12.2 Network Template Collection

2388 A Network Template Collection entity represents the collection of NetworkTemplateas within a Provider.
2389 This resource can be used to locate and create NetworkTemplates.

Name	NetworkTemplateCollection	
Type URI	http://www.dmtf.org/cimi/NetworkTemplateCollection	
Attribute	Type	Description
networkTemplates	ref[]	An array of references to the set of Network Templates in the Provider. Properties: Optional / Mutable

2390 The following describes the serialization of the entity in both JSON and XML:

2391 **JSON media type:** application/CIMI-NetworkTemplateCollection+json

2392 **JSON serialization:**

```
2393 { "self": string,
2394   "name": string, ?
2395   "description": string, ?
2396   "created": string, ?
2397   "properties": { "name": string, + }, ?
2398   "networkTemplates": [
2399     { "href": string }, +
2400   ], ?
2401   "operations": [
2402     { "rel": "add", "href": string }, ?
2403     { "rel": "edit", "href": string } ?
2404   ] ?
2405   ...
2406 }
```

2407 **XML media type:** application/CIMI-NetworkTemplateCollection+xml

2408 **XML serialization:**

```
2409 <NetworkTemplateCollection xmlns="http://www.dmtf.org/cimi">
2410   <self> xs:anyURI </self>
2411   <name> xs:string </name> ?
2412   <description> xs:string </description> ?
2413   <created> xs:string </created>
2414   <property name="xs:string"> xs:string </property> *
2415   <networkTemplate href="xs:anyURI"/> *
2416   <operation rel="add" href="xs:anyURI"/> ?
2417   <operation rel="edit" href="xs:anyURI"/> ?
2418   <xs:any>*
2419 </NetworkTemplateCollection>
```

2420 5.12.2.1 Operations

2421 This entity supports the Read and Update operations. Creation of new Network Template entities is
2422 supported via a POST to the "addLink" URI as described in section 4.2.2.1.

2423 5.12.3 Network Configuration

2424 The set of configuration values representing the information needed to create a Network with certain
2425 characteristics.

Name	NetworkConfiguration	
Type URI	http://www.dmtf.org/cimi/NetworkConfiguration	
Attribute	Type	Description
access	string	An indicator of whether or not the Machine entity has access to a Public or Private network. An indication of Public represents an open and Internet routable network. An indication of Private identifies a local non-routed network. Properties: Optional / Mutable
bandwidthLimit	integer	Maximum allowable bandwidth. Properties: Optional / Mutable
trafficPriority	integer	Indicates priority of traffic on this network. Properties: Optional / Mutable
maxTrafficDelay	integer	The requested maximum delay for end to end transmission specified in nanoseconds with uint64 (i.e. latency). Properties: Optional / Mutable
maxTrafficLoss	integer	The requested maximum percentage traffic loss for end to end transmission with uint8, min 0 max 100. The percentage of traffic lost in the transmission traffic. A value of zero indicates that a lossless transmission is requested. A value of 100 indicates a best effort transmission. The default value is 100. Properties: Optional / Mutable
maxTrafficJitter	integer	The requested maximum jitter for end to end transmission with uint32 when traffic is packetized. The variation between packets arriving specified in nanoseconds with uint64. Properties: Optional / Mutable

2426 The following describes the serialization of the entity in both JSON and XML:

2427 **JSON media type:** application/CIMI-NetworkConfiguration+json

2428 **JSON serialization:**

```

2429 { "self": string,
2430   "name": string, ?
2431   "description": string, ?
2432   "created": string, ?
2433   "properties": { "name": string, + }, ?
2434   "access": string, ?
2435   "bandwidthLimit": number, ?
2436   "trafficPriority": number, ?
2437   "maxTrafficDelay": number, ?
2438   "maxTrafficLoss": number, ?
2439   "maxTrafficJitter": number, ?
2440   "operations": [
2441     { "rel": "edit", "href": string }, ?
2442     { "rel": "delete", "href": string } ?
2443   ] ?

```

2444 } ...
2445 }

2446 **XML media type:** application/CIMI-NetworkConfiguration+xml

2447 **XML serialization:**

```
2448 <NetworkConfiguration xmlns="http://www.dmtf.org/cimi">
2449   <self> xs:anyURI </self>
2450   <name> xs:string </name> ?
2451   <description> xs:string </description> ?
2452   <created> xs:string </created>
2453   <property name="xs:string"> xs:string </property> *
2454   <access> xs:string </access> ?
2455   <bandwidthLimit> xs:string <bandwidthLimit> ?
2456   <trafficPriority> xs:integer </trafficPriority> ?
2457   <maxTrafficDelay> xs:integer </maxTrafficDelay> ?
2458   <maxTrafficLoss> xs:integer </maxTrafficLoss> ?
2459   <maxTrafficJitter> xs:integer </maxTrafficJitter> ?
2460   <operation rel="edit" href="xs:anyURI"/> ?
2461   <operation rel="delete" href="xs:anyURI"/> ?
2462   <xs:any*>
2463 </NetworkConfiguration>
```

2464 **5.12.3.1 Operations**

2465 This entity supports the Read, Update and Delete operations. Create is supported via the Network
2466 Configuration Collection entity.

2467 **5.12.4 Network Configuration Collection**

2468 A Network Configuration Collection entity represents the collection of Network Configurations within a
2469 Provider. This entity can be used to locate and create Network Configurations.

Name	NetworkConfigurationCollection	
Type URI	http://www.dmtf.org/cimi/NetworkConfigurationCollection	
Attribute	Type	Description
networkConfigurations	ref[]	An array of references to the set of Network Configurations in the provider. Properties: Optional / Mutable

2470 The following describes the serialization of the entity in both JSON and XML:

2471 **JSON media type:** application/CIMI-NetworkConfigurationCollection+json

2472 **JSON serialization:**

```
2473 { "self": string,
2474   "name": string, ?
2475   "description": string, ?
2476   "created": string, ?
2477   "properties": { "name": string, + }, ?
2478   "networkConfigurations": [
2479     { "href": string }, +
2480   ], ?
2481   "operations": [
2482     { "rel": "add", "href": string }, ?
2483     { "rel": "edit", "href": string } ?
```

```

2484     ] ?
2485     ...
2486 }

```

2487 **XML media type:** application/CIMI-NetworkConfigurationCollection+xml

2488 **XML serialization:**

```

2489 <NetworkConfigurationCollection xmlns="http://www.dmtf.org/cimi">
2490   <self> xs:anyURI </self>
2491   <name> xs:string </name> ?
2492   <description> xs:string </description> ?
2493   <created> xs:string </created>
2494   <property name="xs:string"> xs:string </property> *
2495   <networkConfiguration href="xs:anyURI"/> *
2496   <operation rel="add" href="xs:anyURI"/> ?
2497   <operation rel="edit" href="xs:anyURI"/> ?
2498   <xs:any>*
2499 </NetworkCollectionCollection>

```

2500 5.12.4.1 Operations

2501 This entity supports the Read and Update operations. Creation of new Network Collection entities is
 2502 supported via a POST to the "addLink" URI as described in section 4.2.2.1.

2503 5.12.5 Network

2504 A Network is a realized entity that represents an abstraction of a layer 2 broadcast domain.

Name	Network	
Type URI	http://www.dmtf.org/cimi/Network	
Attribute	Type	Description
state	string	Indicates the operational state of the Network. For example, STARTED, STOPPED, SUSPENDED are operational states. This value is read-only and will change based on the state of the Network. Properties: Mandatory / Mutable
access	string	An indicator of whether or not the Machine entity has access to a Public or Private network. An indication of Public represents an open and Internet routable network. An indication of Private identifies a local non-routed network. Properties: Optional / Mutable
bandwidthLimit	integer	Maximum allowable bandwidth. Properties: Optional / Mutable
trafficPriority	integer	Indicates priority of traffic on this network. Properties: Optional / Mutable
maxTrafficDelay	integer	The requested maximum delay for end to end transmission specified in nanoseconds with uint64 (i.e. latency). Properties: Optional / Mutable

maxTrafficLoss	integer	The requested maximum percentage traffic loss for end to end transmission with uint8, min 0 max 100. The percentage of traffic lost in the transmission traffic. A value of zero indicates that a lossless transmission is requested. A value of 100 indicates a best effort transmission. The default value is 100. Properties: Optional / Mutable
maxTrafficJitter	integer	The requested maximum jitter for end to end transmission with uint32 when traffic is packetized. The variation between packets arriving specified in nanoseconds with uint64. Properties: Optional / Mutable
routingGroup	ref	A reference to a RoutingGroup that this Network is part of. Note that Networks route to themselves, therefore this attribute will only appear in cases where the Network routes to one or more additional Networks. Properties: Optional / Mutable
meters	ref[]	A list of references to Meters monitored for this Network. Properties: Optional / Mutable
eventLog	ref	A reference to the EventLog of this Network. Properties: Optional / Mutable

2505 The following describes the serialization of the entity in both JSON and XML:

2506 **JSON media type:** application/CIMI-Network+json

2507 **JSON serialization:**

```

2508 { "self": string,
2509   "name": string, ?
2510   "description": string, ?
2511   "created": string, ?
2512   "properties": { "name": string, + }, ?
2513   "state": string,
2514   "access": string, ?
2515   "bandwidthLimit": number, ?
2516   "trafficPriority": number, ?
2517   "maxTrafficDelay": number, ?
2518   "maxTrafficLoss": number, ?
2519   "maxTrafficJitter": number, ?
2520   "routingGroup": { "href": string }, ?
2521   "meters": [
2522     { "href": string }, +
2523   ], ?
2524   "eventLog": { "href": string }, ?
2525   "operations": [
2526     { "rel": "edit", "href": string }, ?
2527     { "rel": "delete", "href": string } ?
2528   ] ?
2529   ...
2530 }
```

2531 **XML media type:** application/CIMI-Network+xml

2532 **XML serialization:**

```

2533 <Network xmlns="http://www.dmtf.org/cimi">
2534   <self> xs:anyURI </self>
2535   <name> xs:string </name> ?
2536   <description> xs:string </description> ?
2537   <created> xs:string </created>
2538   <property name="xs:string"> xs:string </property> *
2539   <state> xs:string </state>
2540   <access> xs:string </access> ?
2541   <bandwidthLimit> xs:integer </bandwidthLimit> ?
2542   <trafficPriority> xs:integer </trafficPriority> ?
2543   <maxTrafficDelay> xs:integer </maxTrafficDelay> ?
2544   <maxTrafficLoss> xs:integer </maxTrafficLoss> ?
2545   <maxTrafficJitter> xs:integer </maxTrafficJiffer> ?
2546   <routingGroup href="xs:anyURI"/> ?
2547   <meter href="xs:anyURI"/> *
2548   <eventLog href="xs:anyURI"/> ?
2549   <operation rel="edit" href="xs:anyURI"/> ?
2550   <operation rel="delete" href="xs:anyURI"/> ?
2551   <xs:any>*
2552 </Network>

```

2553 5.12.5.1 Operations

2554 This entity supports the Read, Update and Delete operations. Create is supported via the Network
2555 Collection entity.

2556 5.12.6 Network Collection

2557 A Network Collection entity represents the collection of Networks within a Provider. This entity can be
2558 used to locate and create Networks.

Name	NetworkCollection	
Type URI	http://www.dmtf.org/cimi/NetworkCollection	
Attribute	Type	Description
networks	ref[]	An array of references to the set of Networks in the Provider. Properties: Optional / Mutable

2559 The following describes the serialization of the entity in both JSON and XML:

2560 **JSON media type:** application/CIMI-NetworkCollection+json

2561 **JSON serialization:**

```

2562 { "self": string,
2563   "name": string, ?
2564   "description": string, ?
2565   "created": string, ?
2566   "properties": { "name": string, + }, ?
2567   "networks": [
2568     { "href": string }, +
2569   ], ?
2570   "operations": [
2571     { "rel": "add", "href": string }, ?
2572     { "rel": "edit", "href": string } ?

```

```
2573 ] ?
2574 ...
2575 }
```

2576 **XML media type:** application/CIMI-NetworkCollection+xml

2577 **XML serialization:**

```
2578 <NetworkCollection xmlns="http://www.dmtf.org/cimi">
2579   <self> xs:anyURI </self>
2580   <name> xs:string </name> ?
2581   <description> xs:string </description> ?
2582   <created> xs:string </created>
2583   <property name="xs:string"> xs:string </property> *
2584   <network href="xs:anyURI"/> *
2585   <operation rel="add" href="xs:anyURI"/> ?
2586   <operation rel="edit" href="xs:anyURI"/> ?
2587   <xs:any>*
2588 </NetworkCollection>
```

2589 5.12.6.1 Operations

2590 This entity supports the Read and Update operations.

2591 The following custom operations are also defined:

2592 **Creating a New Network**

2593 **/link@rel:** add

2594 This operation will create a new Network.

2595 Input parameters: Either a reference to a Network Template or a Network Template itself.

2596 Output parameters: A reference to a new Network and optionally the representation of the Network.

2597 • HTTP/REST Protocol

2598 To create a new Network a POST is sent to the "add" URI of the NetworkCollection where the HTTP
2599 request body SHALL be as described below. Note this structure allows for certain properties to be passed
2600 in "by value" or by "reference". The definition of each property can be found in section 5.12.1.

2601 **JSON media type:** application/CIMI-NetworkCreate+json

2602 **JSON serialization:**

```
2603 { "name": string,
2604   "description": string, ?
2605   "properties": { "name": string, + }, ?
2606   "networkTemplate": { "href": string, ?
2607     "properties": { "name": string, + } ?
2608     "networkConfig": { "href": string, ?
2609       "properties": { "name": string, + }, ?
2610       "access": string, ?
2611       "bandwidthLimit": number, ?
2612       "trafficPriority": number, ?
2613       "maxTrafficDelay": number, ?
2614       "maxTrafficLoss": number, ?
2615       "maxTrafficJitter": number, ?
2616     },
2617     "routingGroup": { "href": string } ?
2618   } ?
2619 }
```

2620 ...
2621 }

2622 **XML media type:** application/CIMI-NetworkCreate+xml

2623 **XML serialization**

```
2624 <NetworkCreate>
2625   <name> xs:string </name>
2626   <description> xs:string </description> ?
2627   <property name="xs:string"> xs:string </property> *
2628   <networkTemplate href="xs:anyURI"? >
2629     <property name="xs:string"> xs:string </property> *
2630     <networkConfig href="xs:anyURI"? >
2631       <access> xs:string </access> ?
2632       <bandwidthLimit> xs:string <bandwidthLimit> ?
2633       <trafficPriority> xs:integer </trafficPriority> ?
2634       <maxTrafficDelay> xs:integer </maxTrafficDelay> ?
2635       <maxTrafficLoss> xs:integer </maxTrafficLoss> ?
2636       <maxTrafficJitter> xs:integer </maxTrafficJitter> ?
2637     </networkConfig>
2638     <routingGroup href="xs:anyURI"/> ?
2639   </networkTemplate>
2640   <xs:any>*
2641 </NetworkCreate>
```

2642 The serialization of some reference properties are specified such that a request MAY either include a
2643 reference ("href") to an existing entity or to include the entity "inline" as a set of additional properties.
2644 Requests SHALL NOT include both a reference and the inlined set of properties.

2645 Upon successful processing of the request, the HTTP response body MAY either be empty or contain a
2646 serialization of the Network entity.

2647 5.12.7 VSP (Virtual Switch Port) Template

2648 The VSP Template is a set of Configuration values for realizing a VSP. A VSP Template may be used to
2649 create multiple VSPs.

Name	VSPTemplate	
Type URI	http://www.dmtf.org/cimi/VSPTemplate	
Attribute	Type	Description
network	ref	A reference to the network to be associated with this VSP. Properties: Mandatory / Mutable
vspConfig	ref	A reference to the VSP Configuration that will be used to create a VSP from this VSP Template. Properties: Mandatory / Mutable

2650 The following describes the serialization of the entity in both JSON and XML:

2651 **JSON media type:** application/CIMI-VSPTemplate+json

2652 **JSON serialization:**

```
2653 { "self": string,
2654   "name": string, ?
2655   "description": string, ?
```

```

2656     "created": string, ?
2657     "properties": { "name": string, + }, ?
2658     "network": { "href": string },
2659     "vspConfig": { "href": string },
2660     "operations": [
2661       { "rel": "edit", "href": string }, ?
2662       { "rel": "delete", "href": string } ?
2663     ] ?
2664     ...
2665   }

```

2666 **XML media type:** application/CIMI-VSPTemplate+xml

2667 **XML serialization:**

```

2668 <VSPTemplate xmlns="http://www.dmtf.org/cimi">
2669   <self> xs:anyURI </self>
2670   <name> xs:string </name> ?
2671   <description> xs:string </description> ?
2672   <created> xs:string </created>
2673   <property name="xs:string"> xs:string </property> *
2674   <network href="xs:anyURI"/>
2675   <vspConfig href="xs:anyURI"/>
2676   <operation rel="edit" href="xs:anyURI"/> ?
2677   <operation rel="delete" href="xs:anyURI"/> ?
2678   <xs:any>*
2679 </VSPTemplate>

```

2680 **5.12.7.1 Operations**

2681 This entity supports the Read, Update and Delete operations. Create is supported via the VSP Template
 2682 Collection entity.

2683 **5.12.8 VSP (Virtual Switch Port) Template Collection**

2684 A VSP Template Collection entity represents the collection of VSP Templates within a Provider. This
 2685 entity can be used to locate and create VSP Templates.

Name	VSPTempalteCollection	
Type URI	http://www.dmtf.org/cimi/VSPTemplateCollection	
Attribute	Type	Description
vspTemplates	ref[]	An array of references to the set of VSP Templates in the Provider. Properties: Optional / Mutable

2686 The following describes the serialization of the entity in both JSON and XML:

2687 **JSON media type:** application/CIMI-VSPTemplateCollection+json

2688 **JSON serialization:**

```

2689 { "self": string,
2690   "name": string, ?
2691   "description": string, ?
2692   "created": string, ?
2693   "properties": { "name": string, + }, ?
2694   "vspTemplates": [
2695     { "href": string }, +
2696   ], ?

```



```

2697     "operations": [
2698       { "rel": "add", "href": string }, ?
2699       { "rel": "edit", "href": string } ?
2700     ] ?
2701     ...
2702   }

```

2703 **XML media type:** application/CIMI-VSPTemplateCollection+xml

2704 **XML serialization:**

```

2705 <VSPTemplateCollection xmlns="http://www.dmtf.org/cimi">
2706   <self> xs:anyURI </self>
2707   <name> xs:string </name> ?
2708   <description> xs:string </description> ?
2709   <created> xs:string </created>
2710   <property name="xs:string"> xs:string </property> *
2711   <vspTemplate href="xs:anyURI"/> *
2712   <operation rel="add" href="xs:anyURI"/> ?
2713   <operation rel="edit" href="xs:anyURI"/> ?
2714   <xs:any>*
2715 </VSPTemplateCollection>

```

2716 5.12.8.1 Operations

2717 This entity supports the Read and Update operations. Creation of new VSP Template entities is
 2718 supported via a POST to the "addLink" URI as described in section 4.2.2.1.

2719 5.12.9 VSP (Virtual Switch Port) Configuration

2720 The set of configuration values representing the information needed to create a VSP with certain
 2721 characteristics.

Name	VSPConfiguration	
Type URI	http://www.dmtf.org/cimi/VSPConfiguration	
Attribute	Type	Description
bandwidthReservation	integer	Minimum Bandwidth requirements. Properties: Optional / Mutable
trafficPriority	integer	Indicates priority of traffic on this network. Properties: Optional / Mutable
maxTrafficDelay	integer	The requested maximum delay for end to end transmission specified in nanoseconds with uint64 (i.e. latency). Properties: Optional / Mutable
maxTrafficLoss	integer	The requested maximum percentage traffic loss for end to end transmission with uint8, min 0 max 100. The percentage of traffic lost in the transmission traffic. A value of zero indicates that a lossless transmission is requested. A value of 100 indicates a best effort transmission. The default value is 100. Properties: Optional / Mutable

maxTrafficJitter	integer	<p>The requested maximum jitter for end to end transmission with uint32 when traffic is packetized. The variation between packets arriving specified in nanoseconds with uint64.</p> <p>Properties: Optional / Mutable</p>
------------------	---------	---

2722 The following describes the serialization of the entity in both JSON and XML:

2723 **JSON media type:** application/CIMI-VSPConfiguration+json

2724 **JSON serialization:**

```

2725 { "self": string,
2726   "name": string,
2727   "description": string, ?
2728   "created": string, ?
2729   "properties": { "name": string, + }, ?
2730   "bandwidthReservation": number, ?
2731   "trafficPriority": number, ?
2732   "maxTrafficDelay": number, ?
2733   "maxTrafficLoss": number, ?
2734   "maxTrafficJitter": number, ?
2735   "operations": [
2736     { "rel": "edit", "href": string }, ?
2737     { "rel": "delete", "href": string } ?
2738   ] ?
2739   ...
2740 }
```

2741 **XML media type:** application/CIMI-VSPConfiguration+xml

2742 **XML serialization:**

```

2743 <VSPConfiguration xmlns="http://www.dmtf.org/cimi">
2744   <self> xs:anyURI </self>
2745   <name> xs:string </name>
2746   <description> xs:string </description> ?
2747   <created> xs:string </created>
2748   <property name="xs:string"> xs:string </property> *
2749   <bandwidthReservation> xs:integer </bandwidthReservation> ?
2750   <trafficPriority> xs:integer </trafficPriority> ?
2751   <maxTrafficDelay> xs:integer </maxTrafficDelay> ?
2752   <maxTrafficLoss> xs:integer </maxTrafficLoss> ?
2753   <maxTrafficJitter> xs:integer </maxTrafficJitter> ?
2754   <operation rel="edit" href="xs:anyURI"/> ?
2755   <operation rel="delete" href="xs:anyURI"/> ?
2756   <xs:any>*
2757 </VSPConfiguration>
```

2758 **5.12.9.1 Operations**

2759 This entity supports the Read, Update and Delete operations. Create is supported via the VSP
 2760 Configuration Collection entity.

2761 **5.12.10 VSP (Virtual Switch Port) Configuration Collection**

2762 A VSP Configuration Collection entity represents the collection of VSP Configurations within a Provider.
 2763 This entity can be used to locate and create VSP Configurations.

Name	VSPConfigurationCollection
-------------	----------------------------

Type URI	http://www.dmtf.org/cimi/VSPConfigurationCollection	
Attribute	Type	Description
vspConfigurations	ref[]	An array of references to the set of VSP Configurations in the Provider. Properties: Optional / Mutable

2764 The following describes the serialization of the entity in both JSON and XML:

2765 **JSON media type:** application/CIMI-VSPConfigurationCollection+json

2766 **JSON serialization:**

```

2767 { "self": string,
2768   "name": string, ?
2769   "description": string, ?
2770   "created": string, ?
2771   "properties": { "name": string, + }, ?
2772   "vspConfigurations": [
2773     { "href": string }, +
2774   ], ?
2775   "operations": [
2776     { "rel": "add", "href": string }, ?
2777     { "rel": "edit", "href": string } ?
2778   ] ?
2779   ...
2780 }
```

2781 **XML media type:** application/CIMI-VSPConfigurationCollection+xml

2782 **XML serialization:**

```

2783 <VSPConfigurationCollection xmlns="http://www.dmtf.org/cimi">
2784   <self> xs:anyURI </self>
2785   <name> xs:string </name> ?
2786   <description> xs:string </description> ?
2787   <created> xs:string </created>
2788   <property name="xs:string"> xs:string </property> *
2789   <vspConfiguration href="xs:anyURI"/> *
2790   <operation rel="add" href="xs:anyURI"/> ?
2791   <operation rel="edit" href="xs:anyURI"/> ?
2792   <xs:any>*
2793 </VSPConfigurationCollection>
```

2794 5.12.10.1 Operations

2795 This entity supports the Read and Update operations. Creation of new VSP Configuration entities is
 2796 supported via a POST to the "addLink" URI as described in section 4.2.2.1.

2797 5.12.11 VSP (Virtual Switch Port)

2798 A VSP represents the connection parameters of a network port.

Name	VSP	
Type URI	http://www.dmtf.org/cimi/VSP	
Attribute	Type	Description
state	string	An indicator of whether or not a specified port is on or off [Default =

		Enabled]. This value is read-only and will change based on the state of the VSP. Properties: Mandatory / Mutable
network	ref	A reference to the network associated with this VSP. Properties: Mandatory / Mutable
bandwidthReservation	integer	Minimum Bandwidth requirements. Properties: Optional / Mutable
trafficPriority	integer	Indicates priority of traffic on this network. Properties: Optional / Mutable
maxTrafficDelay	integer	The requested maximum delay for end to end transmission specified in nanoseconds with uint64 (i.e. latency). Properties: Optional / Mutable
maxTrafficLoss	integer	The requested maximum percentage traffic loss for end to end transmission with uint8, min 0 max 100. The percentage of traffic lost in the transmission traffic. A value of zero indicates that a lossless transmission is requested. A value of 100 indicates a best effort transmission. The default value is 100. Properties: Optional / Mutable
maxTrafficJitter	integer	The requested maximum jitter for end to end transmission with uint32 when traffic is packetized. The variation between packets arriving specified in nanoseconds with uint64. Properties: Mandatory / Mutable
meters	ref[]	A list of references to Meters monitored for this VSP. Properties: Optional / Mutable
eventLog	ref	A reference to the EventLog of this VSP. Properties: Optional / Mutable

2799 The following describes the serialization of the entity in both JSON and XML:

2800 **JSON media type:** application/CIMI-VSP+json

2801 **JSON serialization:**

```

2802 { "self": string,
2803   "name": string,
2804   "description": string, ?
2805   "created": string, ?
2806   "properties": { "name": string, + }, ?
2807   "network": { "href": string },
2808   "state": string, ?
2809   "bandwidthLimit": number, ?
2810   "trafficPriority": number, ?

```

```

2811     "maxTrafficDelay": number, ?
2812     "maxTfafficLoss": number, ?
2813     "maxTrafficJitter": number, ?
2814     "meters": [
2815         { "href": string }, +
2816     ], ?
2817     "eventLog": { "href": string }, ?
2818     "operations": [
2819         { "rel": "edit", "href": string }, ?
2820         { "rel": "delete", "href": string } ?
2821     ] ?
2822     ...
2823 }
    
```

2824 **XML media type:** application/CIMI-VSP+xml

2825 **XML serialization:**

```

2826 <VSP xmlns="http://www.dmtf.org/cimi">
2827   <self> xs:anyURI </self>
2828   <name> xs:string </name>
2829   <description> xs:string </description> ?
2830   <created> xs:string </created>
2831   <property name="xs:string"> xs:string </property> *
2832   <network href="xs:anyURI">
2833   <state> xs:string </state> ?
2834   <bandwidthLimit> xs:integer </bandwidthLimit> ?
2835   <trafficPriority> xs:integer </trafficPriority> ?
2836   <maxTrafficDelay> xs:integer </maxTrafficDelay> ?
2837   <maxTrafficLoss> xs:integer </maxTrafficLoss> ?
2838   <maxTrafficJitter> xs:integer </maxTrafficJiffer> ?
2839   <meter href="xs:anyURI"/> *
2840   <eventLog" href="xs:anyURI"/> ?
2841   <operation rel="edit" href="xs:anyURI"/> ?
2842   <operation rel="delete" href="xs:anyURI"/> ?
2843   <xs:any>*
2844 </VSP>
    
```

2845 **5.12.11.1 Operations**

2846 This entity supports the Read, Update and Delete operations. Create is supported via the VSP Collection
 2847 entity.

2848 **5.12.12 VSP (Virtual Switch Port) Collection**

2849 A VSP Collection entity represents the collection of VSPs within a Provider. This entity can be used to
 2850 locate and create VSPs.

Name	VSPCollection	
Type URI	http://www.dmtf.org/cimi/VSPCollection	
Attribute	Type	Description
vsp	ref[]	An array of references to the set of VSPs in the Provider. Properties: Optional / Mutable

2851 The following describes the serialization of the entity in both JSON and XML:

2852 **JSON media type:** application/CIMI-VSPCollection+json

2853 **JSON serialization:**

```
2854 { "self": string,
2855   "name": string, ?
2856   "description": string, ?
2857   "created": string, ?
2858   "properties": { "name": string, + }, ?
2859   "vsps": [
2860     { "href": string }, +
2861   ], ?
2862   "operations": [
2863     { "rel": "add", "href": string }, ?
2864     { "rel": "edit", "href": string } ?
2865   ] ?
2866   ...
2867 }
```

2868 **XML media type:** application/CIMI-VSPCollection+xml

2869 **XML serialization:**

```
2870 <VSPCollection xmlns="http://www.dmtf.org/cimi">
2871   <self> xs:anyURI </self>
2872   <name> xs:string </name> ?
2873   <description> xs:string </description> ?
2874   <created> xs:string </created>
2875   <property name="xs:string"> xs:string </property> *
2876   <vsp href="xs:anyURI"/> *
2877   <operation rel="add" href="xs:anyURI"/> ?
2878   <operation rel="edit" href="xs:anyURI"/> ?
2879   <xs:any>*
2880 </VSPCollection>
```

2881 5.12.12.1 Operations

2882 This entity supports the Read and Update operations.

2883 The following custom operations are also defined:

2884 **Creating a New VSP**

2885 **/link@rel:** add

2886 This operation will create a new VSP.

2887 Input parameters: Either a reference to a VSP Template or a VSP Template itself.

2888 Output parameters: A reference to a new VSP and, optionally, the representation of the VSP.

2889 • HTTP/REST Protocol

2890 To create a new VSP a POST is sent to the "add" URI of the VSPCollection where the HTTP request
2891 body SHALL be as described below. Note this structure allows for certain properties to be passed in "by
2892 value" or by "reference". The definition of each property can be found in section 5.12.7.

2893 **JSON media type:** application/CIMI-VSPCreate+json

2894 **JSON serialization:**

```
2895 { "name": string,
2896   "description": string, ?
```

```

2897 "properties": { "name": string, + }, ?
2898 "vspTemplate": { "href": string, ?
2899   "properties": { "name": string, + } ?
2900   "network": { "href": string }, ?
2901   "vspConfig": { "href": string, ?
2902     "properties": { "name": string, + }, ?
2903     "bandwidthReservation": number, ?
2904     "trafficPriority": number, ?
2905     "maxTrafficDelay": number, ?
2906     "maxTrafficLoss": number, ?
2907     "maxTrafficJitter": number ?
2908   } ?
2909 }
2910 ...
2911 }
    
```

2912 **XML media type:** application/CIMI-VSPCreate+xml

2913 **XML serialization**

```

2914 <NVSPCreate>
2915   <name> xs:string </name>
2916   <description> xs:string </description> ?
2917   <property name="xs:string"> xs:string </property> *
2918   <vspTemplate href="xs:anyURI"? >
2919     <property name="xs:string"> xs:string </property> *
2920     <network href="xs:anyURI"/> ?
2921     <vspConfig href="xs:anyURI">
2922       <property name="xs:string"> xs:string </property> *
2923       <bandwidthReservation> xs:integer </bandwidthReservation> ?
2924       <trafficPriority> xs:integer </trafficPriority> ?
2925       <maxTrafficDelay> xs:integer </maxTrafficDelay> ?
2926       <maxTrafficLoss> xs:integer </maxTrafficLoss> ?
2927       <maxTrafficJitter> xs:integer </maxTrafficJitter> ?
2928     </vspConfig> ?
2929   </vspTemplate>
2930   <xs:any>*
2931 </NVSPCreate>
    
```

2932 The serialization of some reference properties are specified such that a request MAY either include a
 2933 reference ("href") to an existing entity or to include the entity "inline" as a set of additional properties.
 2934 Requests SHALL NOT include both a reference and the inlined set of properties.

2935 Upon successful processing of the request, the HTTP response body MAY either be empty or contain a
 2936 serialization of the VSP entity.

2937 **5.12.13 Routing Group**

2938 A Routing Group represents a collection of Networks that route to each other..

Name	RoutingGroup	
Type URI	http://www.dmtf.org/cimi/RoutingGroup	
Attribute	Type	Description
networks	ref[]	An array of references to the networks in this Routing Group. Properties: Optional / Mutable

2939 The following describes the serialization of the entity in both JSON and XML:

2940 **JSON media type:** application/CIMI-RoutingGroup+json

2941 **JSON serialization:**

```

2942 { "self": string,
2943     "name": string,
2944     "description": string, ?
2945     "created": string, ?
2946     "properties": { "name": string, + }, ?
2947     "networks": [
2948         { "href": string }, +
2949     ], ?
2950     "operations": [
2951         { "rel": "edit", "href": string }, ?
2952         { "rel": "delete", "href": string } ?
2953     ] ?
2954     ...
2955 }
```

2956 **XML media type:** application/CIMI-RoutingGroup+xml

2957 **XML serialization:**

```

2958 <VSP xmlns="http://www.dmtf.org/cimi">
2959   <self> xs:anyURI </self>
2960   <name> xs:string </name>
2961   <description> xs:string </description> ?
2962   <created> xs:string </created>
2963   <property name="xs:string"> xs:string </property> *
2964   <network href="xs:anyURI"> *
2965   <operation rel="edit" href="xs:anyURI"/> ?
2966   <operation rel="delete" href="xs:anyURI"/> ?
2967   <xs:any*>
2968 </VSP>
```

2969 **5.12.13.1 Operations**

2970 This entity supports the Read, Update and Delete operations. Create is supported via the RoutingGroup
 2971 Collection entity.

2972 **5.12.14 Routing Group Collection**

2973 A Routing Group Collection entity represents the collection of Routing Groups within a Provider. This
 2974 entity can be used to locate and create Routing Groups.

Name	RoutingGroupCollection	
Type URI	http://www.dmtf.org/cimi/RoutingGroupCollection	
Attribute	Type	Description
routingGroups	ref[]	An array of references to the set of RoutingGroups in the Provider. Properties: Optional / Mutable

2975 The following describes the serialization of the entity in both JSON and XML:

2976 **JSON media type:** application/CIMI-RoutingGroupCollection+json

2977 **JSON serialization:**

```

2978 { "self": string,
```



```

2979     "name": string, ?
2980     "description": string, ?
2981     "created": string, ?
2982     "properties": { "name": string, + }, ?
2983     "routingGroups": [
2984         { "href": string }, +
2985     ], ?
2986     "operations": [
2987         { "rel": "add", "href": string }, ?
2988         { "rel": "edit", "href": string } ?
2989     ] ?
2990     ...
2991 }

```

2992 **XML media type:** application/CIMI-RoutingGroupCollection+xml

2993 **XML serialization:**

```

2994 <RoutingGroupCollection xmlns="http://www.dmtf.org/cimi">
2995   <self> xs:anyURI </self>
2996   <name> xs:string </name> ?
2997   <description> xs:string </description> ?
2998   <created> xs:string </created>
2999   <property name="xs:string"> xs:string </property> *
3000   <routingGroup href="xs:anyURI"/> *
3001   <operation rel="add" href="xs:anyURI"/> ?
3002   <operation rel="edit" href="xs:anyURI"/> ?
3003   <xs:any>*
3004 </RoutingGroupCollection>

```

3005 5.12.14.1 Operations

3006 This entity supports the Read and Update operations.

3007 The following custom operations are also defined:

3008 **Creating a New RoutingGroup**

3009 **/link@rel:** add

3010 This operation will create a new RoutingGroup.

3011 Input parameters: A RoutingGroup definition.

3012 Output parameters: A reference to a new RoutingGroup and, optionally, the representation of the
3013 RoutingGroup.

3014 • HTTP/REST Protocol

3015 To create a new RoutingGroup a POST is sent to the "add" URI of the RoutingGroupCollection where the
3016 HTTP request body SHALL be as described below. The definition of each property can be found in
3017 section 5.12.13.

3018 **JSON media type:** application/CIMI-RoutingGroupCreate+json

3019 **JSON serialization:**

```

3020 { "name": string,
3021   "description": string, ?
3022   "properties": { "name": string, + }, ?
3023   "networks": [
3024     { "href": string }, +
3025   ] ?

```

3026
3027

```
    ...
  }
```

3028 **XML media type:** application/CIMI-RoutingGroupCreate+xml

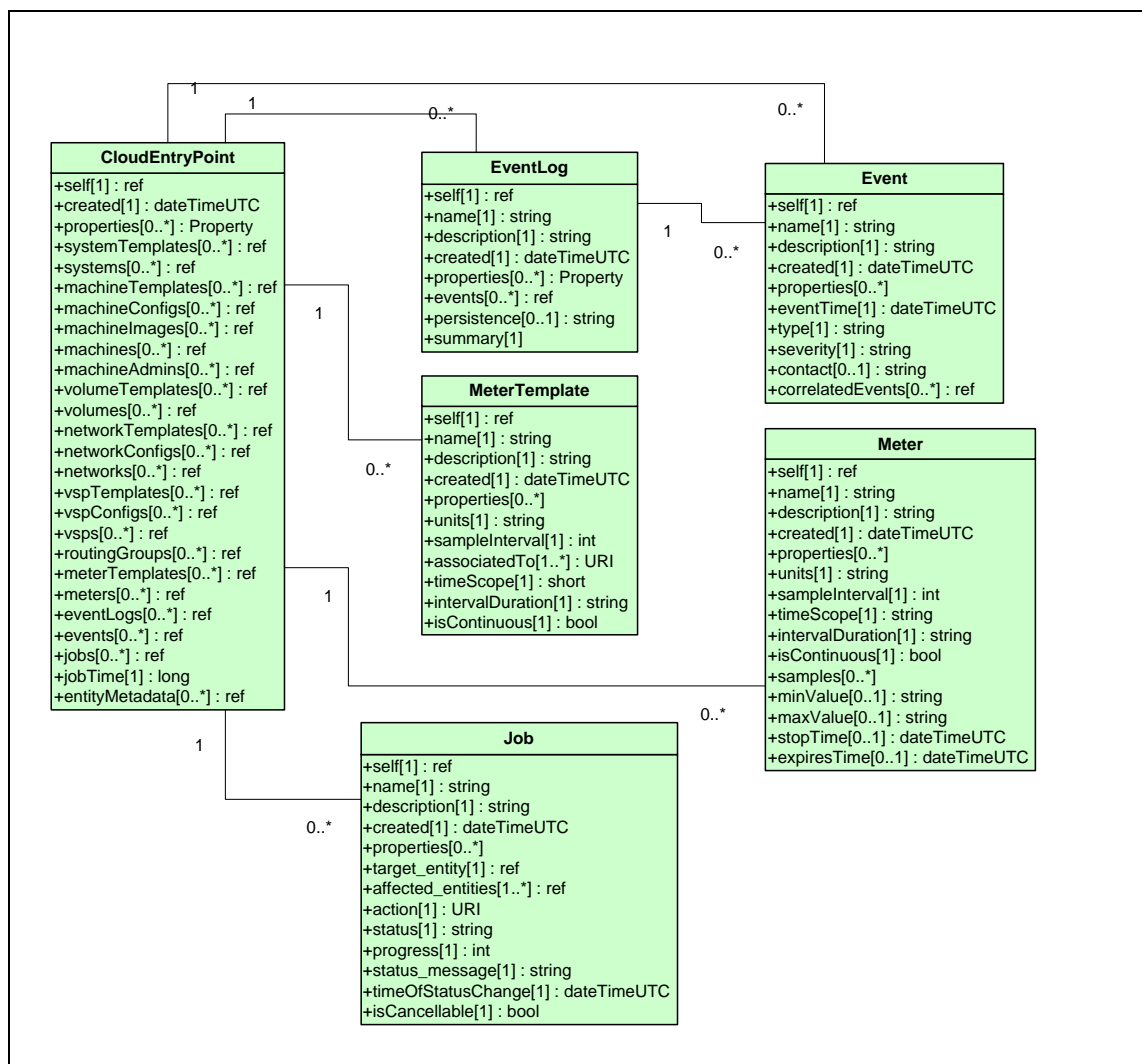
3029 **XML serialization**

```
3030 <RoutingGroupCreate>
3031   <name> xs:string </name>
3032   <description> xs:string </description> ?
3033   <property name="xs:string"> xs:string </property> *
3034   <network href="xs:anyURI"/> *
3035   <xs:any>*
3036 </RoutingGroupCreate>
```

3037 Upon successful processing of the request, the HTTP response body MAY either be empty or contain a
3038 serialization of the RoutingGroup entity.

3039 **5.13 Monitoring Entities and Relationships**

3040 The following diagram illustrates the entities involved in tracking the progress of operations as well as
3041 metering and monitoring the status of other entities. Although this drawing is in the style of an Entity
3042 Relationship diagram, the use of UML is neither rigorous nor normative.



3043

3044 **Figure 5 - Monitoring Entities**

3045 **5.13.1 Job**

3046 This entity represents a process (i.e. a sequence of one or more operations directed to accomplish a
3047 specific goal) performed by the Provider.

3048 If a Provider supports exposing Job entities to Consumers then each request from a Consumer that would
3049 result in a change to the environment MUST result in a Job entity being created and a reference to that
3050 Job entity MUST be made available to the requestingConsumer. Providers MAY create additional Job
3051 entities for Provider initiated operations.

Name	Job	
Type URI	http://www.dmtf.org/cimi/Job	
Attribute	Type	Description
targetEntity	ref	A reference to the top-level entity upon which the operation is being performed. Typically, this would be the instance which was specified when the operation was initiated. This attribute value is

		read-only. Properties: Mandatory / Mutable
affectedEntities	ref[]	An array of references to each of the entities upon which the operation is being performed. This array always includes the reference specified in the target_entity attribute. The values in this array are read-only . Properties: Mandatory / Mutable
action	URI	A URI that indicates the type of action being performed. Properties: Mandatory / Immutable
status	string	The current status of the process associated with this operation. Sample values for this include "running", "failed", "success", and "cancelled". This attribute value is read-only . Properties: Mandatory / Mutable
progress	integer	An integer value in the range 0 ... 100 that indicates the progress of this Job. This attribute value is read-only . Properties: Mandatory / Mutable
statusMessage	string	This attribute is a human-readable string that provides information about the operation. It is used to further qualify or provide additional information about the current status of the operation. For example, this may indicate the reason why the operation failed, or whether the operation was cancelled by the Consumer or the Provider. This attribute value is read-only . Properties: Mandatory / Mutable
timeOfStatusChange	DateTimeUTC	A timestamp indicating the last time that the status of the operation changed. This attribute value is read-only . Properties: Mandatory / Mutable
isCancellable	boolean	Specifies whether the task being performed by this Job object supports the Cancel operation. Properties: Mandatory / Immutable

3052 The following describes the serialization of the entity in both JSON and XML:

3053 **JSON media type:** application/CIMI-Job+json

3054 **JSON serialization:**

```

3055 { "self": string,
3056   "name": string,
3057   "description": string, ?
3058   "created": string, ?
3059   "properties": { "name": string, + }, ?
3060   "targetEntity": { "href": string },
3061   "affectedEntities": [
3062     { "href": string }, +
3063   ], ?
3064   "action": string,
```

```

3065     "status": string,
3066     "progress": integer,
3067     "statusMessage": string,
3068     "timeOfStatusChange": date,
3069     "isCancellable": boolean,
3070     "operations": [
3071       { "rel": "edit", "href": string }, ?
3072       { "rel": "delete", "href": string } ?
3073     ] ?
3074     ...
3075   }

```

3076 **XML media type:** application/CIMI-Job+xml

3077 **XML serialization:**

```

3078 <Job xmlns="http://www.dmtf.org/cimi">
3079   <self> xs:anyURI </self>
3080   <name> xs:string </name>
3081   <description> xs:string </description> ?
3082   <created> xs:string </created>
3083   <property name="xs:string"> xs:string </property> *
3084   <targetEntity href="xs:anyURI"/>
3085   <affectedEntity> xs:anyURI </affectedEntity> *
3086   <action> xs:anyURI </action>
3087   <status> xs:string </status>
3088   <progress> xs:integer </progress>
3089   <statusMessage> xs:string </statusMessage>
3090   <timeOfStatusChange> xs:dateTime </timeOfStatusChange>
3091   <isCancellable> xs:boolean </isCancellable>
3092   <operation rel="edit" href="xs:anyURI"/> ?
3093   <operation rel="delete" href="xs:anyURI"/> ?
3094   <xs:any*>
3095 </Job>

```

3096 5.13.1.1 Operations

3097 This entity supports the Read, Update and Delete operations.

3098 5.13.2 Job Collection

3099 A Job Collection entity represents the collection of Jobs within a Provider. This resource can be used to
3100 locate Jobs.

Name	JobCollection	
Type URI	http://www.dmtf.org/cimi/JobCollection	
Attribute	Type	Description
jobs	ref[]	An array of references to the set of Jobs in the Provider. Properties: Optional / Mutable

3101 The following describes the serialization of the entity in both JSON and XML:

3102 **JSON media type:** application/CIMI-JobCollection+json

3103 **JSON serialization:**

```

3104 { "self": string,
3105   "name": string, ?

```

```

3106 "description": string, ?
3107 "created": string, ?
3108 "properties": { "name": string, + }, ?
3109 "jobs": [
3110   { "href": string }, +
3111 ], ?
3112 "operations": [
3113   { "rel": "edit", "href": string } ?
3114 ] ?
3115 ...
3116 }
    
```

3117 **XML media type:** application/CIMI-JobCollection+xml

3118 **XML serialization:**

```

3119 <JobCollection xmlns="http://www.dmtf.org/cimi">
3120   <self> xs:anyURI </self>
3121   <name> xs:string </name> ?
3122   <description> xs:string </description> ?
3123   <created> xs:string </created>
3124   <property name="xs:string"> xs:string </property> *
3125   <job href="xs:anyURI"/> *
3126   <operation rel="edit" href="xs:anyURI"/> ?
3127   <xs:any>*
3128 </JobCollection>
    
```

3129 **5.13.2.1 Operations**

3130 This entity supports the Read and Update operations.

3131 **5.13.3 Meter Template**

3132 A Meter Template represents the definition of a Meter. A Meter Template can only be created by the
 3133 Provider.

Name	MeterTemplate	
Type URI	http://www.dmtf.org/cimi/MeterTemplate	
Attribute	Type	Description
units	string	Name of the used units, e.g. kilobits per second, CPU usage percentage, etc. This attribute is read-only . Properties: Mandatory / Mutable
sampleInterval	integer	It indicates the time between consecutive samples in seconds. This attribute is read-only . Properties: Mandatory / Mutable
associatedTo	URI[]	An array of URIs that indicate the entities to which a Meter created from this template can be applied. The value space of these URIs is identical to that of EntityMetadata.typeURI - a URI that uniquely identifies an entity type. This attribute is read-only . Properties: Mandatory / Mutable
timeScope	string	It indicates the time scope to which the Meter value applies.

		Two possible values: "Point" indicates that the Meter applies to a point in time. "Interval" indicates that the Meter applies to a time interval. For instance, it would be possible to define a MeterTemplate which purpose is to provide the daily average CPU usage. This attribute is read-only . Properties: Mandatory / Mutable
intervalDuration	string	It indicates the interval duration when the timeScope is set to "Interval". Possible values: hourly, daily, weekly, monthly or yearly. This attribute is read-only . Properties: Mandatory / Mutable
isContinuous	boolean	It indicates whether or not the Meter value is continuous or scalar. Performance Meters are an example of a linear metric. This attribute is read-only . Properties: Mandatory / Mutable

3134 The following describes the serialization of the entity in both JSON and XML:

3135 **JSON media type:** application/CIMI-MeterTemplate+json

3136 **JSON serialization:**

```

3137 { "self": string,
3138   "name": string, ?
3139   "description": string, ?
3140   "created": string, ?
3141   "properties": { "name": string, + }, ?
3142   "units": string,
3143   "sampleInterval": integer,
3144   "associatedTo": [
3145     { "href": string }, +
3146   ], ?
3147   "timeScope": string,
3148   "intervalDuration": string,
3149   "isContinuous": boolean,
3150   "operations": [
3151     { "rel": "edit", "href": string }, ?
3152     { "rel": "delete", "href": string } ?
3153   ] ?
3154   ...
3155 }
```

3156 **XML media type:** application/CIMI-MeterTemplate+xml

3157 **XML serialization:**

```

3158 <MeterTemplate xmlns="http://www.dmtf.org/cimi">
3159   <self> xs:anyURI </self>
3160   <name> xs:string </name> ?
3161   <description> xs:string </description> ?
3162   <created> xs:string </created>
3163   <property name="xs:string"> xs:string </property> *
3164   <units> xs:string </units>
3165   <sampleInterval> xs:integer </sampleInterval>
3166   <associatedTo href="xs:anyURI"/> *
3167   <timeScope> xs:string </timeScope>
3168   <intervalDuration> xs:string </intervalDuration>
3169   <isContinuous> xs:boolean </isContinuous>
```

```

3170     <operation rel="edit" href="xs:anyURI"/> ?
3171     <operation rel="delete" href="xs:anyURI"/> ?
3172     <xs:any>*
3173 </MeterTemplate>

```

3174 5.13.3.1 Operations

3175 This entity supports the Read, Update and Delete operations. Create is supported via the Meter Template
 3176 entity.

3177 5.13.4 Meter Template Collection

3178 A Meter Template Collection entity represents the collection of Meter Templates within a Provider. This
 3179 entity can be used to locate Meter Templates.

Name	MeterTemplateCollection	
Type URI	http://www.dmtf.org/cimi/MeterTemplateCollection	
Attribute	Type	Description
meterTemplates	ref[]	An array of references to the set of Meter Templates in the Provider. Properties: Optional / Mutable

3180 The following describes the serialization of the entity in both JSON and XML:

3181 **JSON media type:** application/CIMI-MeterTemplateCollection+json

3182 **JSON serialization:**

```

3183 { "self": string,
3184   "name": string, ?
3185   "description": string, ?
3186   "created": string, ?
3187   "properties": { "name": string, + }, ?
3188   "meterTemplates": [
3189     { "href": string }, +
3190   ], ?
3191   "operations": [
3192     { "rel": "add", "href": string } ?
3193     { "rel": "edit", "href": string } ?
3194   ] ?
3195   ...
3196 }

```

3197 **XML media type:** application/CIMI-MeterTemplateCollection+xml

3198 **XML serialization:**

```

3199 <MeterTemplateCollection xmlns="http://www.dmtf.org/cimi">
3200   <self> xs:anyURI </self>
3201   <name> xs:string </name> ?
3202   <description> xs:string </description> ?
3203   <created> xs:string </created>
3204   <property name="xs:string"> xs:string </property> *
3205   <meterTemplate href="xs:anyURI"/> *
3206   <operation rel="add" href="xs:anyURI"/> ?
3207   <operation rel="edit" href="xs:anyURI"/> ?
3208   <xs:any>*
3209 </MeterTemplateCollection>

```


3210 **5.13.4.1 Operations**

3211 This entity supports the Read and Update operations. Creation of new Meter Template entities is
 3212 supported via a POST to the "addLink" URI as described in section 4.2.2.1.

3213 **5.13.5 Meter**

3214 This entity represents an available Meter of some property associated to a given entity.

Name	Meter										
Type URI	http://www.dmtf.org/cimi/Meter										
Attribute	Type	Description									
units	string	Name of the used units, e.g. kilobits per second, CPU usage percentage, etc. This attribute is read-only . Properties: Mandatory / Mutable									
sampleInterval	integer	It indicates the time between consecutive samples in seconds. This attribute is read-only . Properties: Mandatory / Mutable									
timeScope	string	It indicates the time scope to which this meter's value applies. Two possible values: "Point" indicates that the Meter applies to a point in time. "Interval" indicates that the Meter applies to a time interval. For instance, it would be possible to define a MeterTemplate which purpose is to provide the daily average CPU usage. This attribute is read-only . Properties: Mandatory / Mutable									
intervalDuration	string	It indicates the interval duration when the timeScope is set to "Interval". Possible values: hourly, daily, weekly, monthly or yearly. This attribute is read-only . Properties: Mandatory / Mutable									
isContinuous	boolean	It indicates whether or not the Meter value is continuous or scalar. Performance Meters are an example of a linear metric. This attribute is read-only . Properties: Mandatory / Mutable									
samples	sample[]	A list of taken samples Each sample attribute has the following sub-attributes: <table border="1" data-bbox="609 1612 1435 1892"> <tr> <td>Name</td> <td colspan="2">sample</td> </tr> <tr> <td>Attribute</td> <td>Type</td> <td>Description</td> </tr> <tr> <td>timeStamp</td> <td>DateTimeUTC</td> <td>It indicates when the measure was taken (timeScope="Point"). When the timeScope is "Interval", it indicates the end of the time interval.</td> </tr> </table>	Name	sample		Attribute	Type	Description	timeStamp	DateTimeUTC	It indicates when the measure was taken (timeScope="Point"). When the timeScope is "Interval", it indicates the end of the time interval.
Name	sample										
Attribute	Type	Description									
timeStamp	DateTimeUTC	It indicates when the measure was taken (timeScope="Point"). When the timeScope is "Interval", it indicates the end of the time interval.									

		<table border="1"> <tr> <td></td> <td></td> <td>Properties: Mandatory / Mutable</td> </tr> <tr> <td>value</td> <td>string</td> <td>It indicates the sampled value of the measure. Properties: Mandatory / Mutable</td> </tr> </table> <p>Properties: Optional / Mutable</p>			Properties: Mandatory / Mutable	value	string	It indicates the sampled value of the measure. Properties: Mandatory / Mutable
		Properties: Mandatory / Mutable						
value	string	It indicates the sampled value of the measure. Properties: Mandatory / Mutable						
minValue	string	It indicates the expected minimal measure value. This attribute is read-only . Properties: Optional / Mutable						
maxValue	string	It indicates the expected maximum measure value. This attribute is read-only . Properties: Optional / Mutable						
stopTime	dateTimeUTC	It indicates a time from which the meter stops tracking samples. This attribute is writable . Properties: Optional / Mutable						
expiresTime	dateTimeUTC	It indicates the time from which the Meter is not monitored anymore. It implies the deletion of the Meter after this time. This attribute is writable . Properties: Optional / Mutable						

3215 The following describes the serialization of the entity in both JSON and XML:

3216 **JSON media type:** application/CIMI-Meter+json

3217 **JSON serialization:**

```

3218 { "self": string,
3219   "name": string, ?
3220   "description": string, ?
3221   "created": string, ?
3222   "properties": { "name": string, + }, ?
3223   "units": string,
3224   "sampleInterval": integer,
3225   "timeScope": string,
3226   "intervalDuration": string,
3227   "isContinuous": boolean,
3228   "samples": [
3229     { "timestamp": string, "value": string }, +
3230   ], ?
3231   "minValue": string, ?
3232   "maxValue": string, ?
3233   "stopTime": string, ?
3234   "expiresTime": string, ?
3235   "operations": [
3236     { "rel": "edit", "href": string }, ?
3237     { "rel": "delete", "href": string } ?
3238   ] ?
3239   ...
3240 }
```

3241 **XML media type:** application/CIMI-Meter+xml

3242 **XML serialization:**

```

3243 <Meter xmlns="http://www.dmtf.org/cimi">
3244   <self> xs:anyURI </self>
3245   <name> xs:string </name> ?
3246   <description> xs:string </description> ?
3247   <created> xs:string </created>
3248   <property name="xs:string"> xs:string </property> *
3249   <units> xs:string </units>
3250   <sampleInterval> xs:integer </sampleInterval>
3251   <timeScope> xs:string <timeScope>
3252   <intervalDuration xs:string </intervalDuration>
3253   <isContinuous> xs:boolean </isContinuous>
3254   <sample timestamp="xs:dateTime" value="xs:string"/> *
3255   <minValue> xs:string </minValue> ?
3256   <maxValue> xs:string </maxValue> ?
3257   <stopTime> xs:dateTime </stopTime> ?
3258   <expiresTime> xs:dateTime </expiresTime> ?
3259   <operation rel="edit" href="xs:anyURI"/> ?
3260   <operation rel="delete" href="xs:anyURI"/> ?
3261   <xs:any*>
3262 </Meter>

```

3263 5.13.5.1 Operations

3264 This entity supports the Read, Update and Delete operations. Create is supported via the Meter
3265 Collection entity.

3266 The following custom operations are also defined:

3267 **Starting a Meter**

3268 **/link@rel:** http://www.dmtf.org/cimi/action/start

3269 This operation will start a Meter.

3270 Input parameters: None.

3271 Output parameters: None.

3272 Upon successful completion of this operation the Meter will begin to record samples related to its
3273 associated resource.

- 3274 • **HTTP/REST Protocol**

3275 To start a Meter a POST is sent to the "http://www.dmtf.org/cimi/start" URI of the Meter where the HTTP
3276 request body SHALL be as described below.

3277 **JSON media type:** application/CIMI-Action+json

3278 **JSON serialization:**

```

3279 { "action": "http://www.dmtf.org/cimi/action/start" ,
3280   "properties": { "name": string, + } ?
3281   ...
3282 }

```

3283 **XML media type:** application/CIMI-Action+xml

3284 **XML serialization**

```
3285 <Action xmlns="http://www.dmtf.org/cimi">
3286   <action> http://www.dmtf.org/cimi/action/start </action>
3287   <property name="xs:string"> xs:string </property> *
3288   <xs:any>*
3289 </Action>
```

3290 Upon successful processing of the request, the HTTP response body will be empty.

3291 **Stopping a Meter**

3292 **/link@rel:** http://www.dmtf.org/cimi/action/stop

3293 This operation will stop a Meter.

3294 Input parameters: None.

3295 Output parameters: None.

3296 Upon successful completion of this operation the Meter will no longer be recording samples related to its
3297 associated resource.

3298 • **HTTP/REST Protocol**

3299 To stop a Meter a POST is sent to the "http://www.dmtf.org/cimi/start" URI of the Meter where the HTTP
3300 request body SHALL be as described below.

3301 **JSON media type:** application/CIMI-Action+json

3302 **JSON serialization:**

```
3303 { "action": "http://www.dmtf.org/cimi/action/stop" ,
3304   "properties": { "name": string, + } ?
3305   ...
3306 }
```

3307 **XML media type:** application/CIMI-Action+xml

3308 **XML serialization**

```
3309 <Action xmlns="http://www.dmtf.org/cimi">
3310   <action> http://www.dmtf.org/cimi/action/stop </action>
3311   <property name="xs:string"> xs:string </property> *
3312   <xs:any>*
3313 </Action>
```

3314 Upon successful processing of the request, the HTTP response body will be empty.

3315 **5.13.6 Meter Collection**

3316 A Meter Collection entity represents the collection of Meters within a Provider. This entity can be used to
3317 locate and create Meters.

Name	MeterCollection	
Type URI	http://www.dmtf.org/cimi/MeterCollection	
Attribute	Type	Description

meters	ref[]	An array of references to the set of Meters in the Provider. Properties: Optional / Mutable
--------	-------	---

3318 The following describes the serialization of the entity in both JSON and XML:

3319 **JSON media type:** application/CIMI-MeterCollection+json

3320 **JSON serialization:**

```

3321 { "self": string,
3322   "name": string, ?
3323   "description": string, ?
3324   "created": string, ?
3325   "properties": { "name": string, + }, ?
3326   "meters": [
3327     { "href": string }, +
3328   ], ?
3329   "operations": [
3330     { "rel": "add", "href": string }, ?
3331     { "rel": "edit", "href": string } ?
3332   ] ?
3333   ...
3334 }
```

3335 **XML media type:** application/CIMI-MeterCollection+xml

3336 **XML serialization:**

```

3337 <MeterCollection xmlns="http://www.dmtf.org/cimi">
3338   <self> xs:anyURI </self>
3339   <name> xs:string </name> ?
3340   <description> xs:string </description> ?
3341   <created> xs:string </created>
3342   <property name="xs:string"> xs:string </property> *
3343   <meter href="xs:anyURI"/> *
3344   <operation rel="add" href="xs:anyURI"/> ?
3345   <operation rel="edit" href="xs:anyURI"/> ?
3346   <xs:any>*
3347 </MeterCollection>
```

3348 5.13.6.1 Operations

3349 This entity supports the Read and Update operations.

3350 The following custom operations are also defined:

3351 **Creating a New Meter**

3352 **/link@rel:** add

3353 This operation will create a new Meter.

3354 Input parameters: Either a reference to a Meter Template or a Meter Template itself.

3355 Output parameters: A reference to a new Meter and optionally the representation of the Meter.

3356 • **HTTP/REST Protocol**

3357 To create a new Meter a POST is sent to the "add" URI of the MeterCollection where the HTTP request
3358 body SHALL be as described below. Note this structure allows for certain properties to be passed in "by
3359 value" or by "reference". The definition of each property can be found in section 5.13.3.

3360 **JSON media type:** application/CIMI-MeterCreate+json

3361 **JSON serialization:**

```

3362 { "name": string,
3363     "description": string, ?
3364     "properties": { "name": string, + }, ?
3365     "meterTemplate": { "href": string, ?
3366         "properties": { "name": string, + }, ?
3367         "units": string, ?
3368         "sampleInterval": integer, ?
3369         "associatedTo": [
3370             { "href": string }, +
3371         ], ?
3372         "timeScope": string, ?
3373         "intervalDuration": string, ?
3374         "isContinuous": boolean, ?
3375     }
3376     ...
3377 }
```

3378 **XML media type:** application/CIMI-MeterCreate+xml

3379 **XML serialization**

```

3380 <MeterCreate>
3381   <name> xs:string </name>
3382   <description> xs:string </description> ?
3383   <property name="xs:string"> xs:string </property> *
3384   <meterTemplate href="xs:anyURI"? >
3385     <property name="xs:string"> xs:string </property> *
3386     <units> xs:string </units> ?
3387     <sampleInterval> xs:integer </sampleInterval> ?
3388     <associatedTo href="xs:anyURI"/> *
3389     <timeScope> xs:string </timeScope> ?
3390     <intervalDuration> xs:string </intervalDuration> ?
3391     <isContinuous> xs:boolean </isContinuous> ?
3392   </meterTemplate>
3393   <xs:any>*
3394 </MeterCreate>
```

3395 The serialization of some reference properties are specified such that a request MAY either include a
 3396 reference ("href") to an existing entity or to include the entity "inline" as a set of additional properties.
 3397 Requests SHALL NOT include both a reference and the inlined set of properties.

3398 Upon successful processing of the request, the HTTP response body MAY either be empty or contain a
 3399 serialization of the Meter entity.

3400 **5.13.7 Event Log**

3401 An entity that represents a registry of Events.

Name	EventLog	
Type URI	http://www.dmtf.org/cimi/EventLog	
Attribute	Type	Description
targetEntity	ref	A reference to the entity to which the Events are related.
events	ref[]	A list of references to occurred Events.

		Properties: Optional / Mutable															
persistence	string	A value that indicates the persistence of the Events within the EventLog. For instance, daily, weekly, monthly or yearly. Properties: Mandatory / Mutable															
summary	structure	A summary of all the events present in the EventLog when the read operation is performed, grouped per severity. Each summary attribute has the following sub-attributes:															
		<table border="1"> <thead> <tr> <th>Attribute</th> <th>Type</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>low</td> <td>integer</td> <td>Number of occurred Events with a low severity. Properties: Mandatory / Mutable</td> </tr> <tr> <td>medium</td> <td>integer</td> <td>Number of occurred Events with a medium severity. Properties: Mandatory / Mutable</td> </tr> <tr> <td>high</td> <td>integer</td> <td>Number of occurred Events with a high severity. Properties: Mandatory / Mutable</td> </tr> <tr> <td>critical</td> <td>integer</td> <td>Number of occurred Events with a critical severity. Properties: Mandatory / Mutable</td> </tr> </tbody> </table>	Attribute	Type	Description	low	integer	Number of occurred Events with a low severity. Properties: Mandatory / Mutable	medium	integer	Number of occurred Events with a medium severity. Properties: Mandatory / Mutable	high	integer	Number of occurred Events with a high severity. Properties: Mandatory / Mutable	critical	integer	Number of occurred Events with a critical severity. Properties: Mandatory / Mutable
		Attribute	Type	Description													
		low	integer	Number of occurred Events with a low severity. Properties: Mandatory / Mutable													
		medium	integer	Number of occurred Events with a medium severity. Properties: Mandatory / Mutable													
		high	integer	Number of occurred Events with a high severity. Properties: Mandatory / Mutable													
critical	integer	Number of occurred Events with a critical severity. Properties: Mandatory / Mutable															
Properties: Mandatory / Mutable																	

3402 The following describes the serialization of the entity in both JSON and XML:

3403 **JSON media type:** application/CIMI-EventLog+json

3404 **JSON serialization:**

```

3405 { "self": string,
3406   "name": string, ?
3407   "description": string, ?
3408   "created": string, ?
3409   "properties": { "name": string, + }, ?
3410   "targetEntity": { "href": string },
3411   "events": [
3412     { "href": string }, +
3413   ], ?
3414   "persistence", string,
3415   "summary", {
3416     "low": number,
3417     "medium": number,
3418     "high": number,
3419     "critical": number
3420   }, ?
3421   "operations": [
3422     { "rel": "edit", "href": string }, ?
3423     { "rel": "delete", "href": string } ?
3424   ] ?
3425   ...

```

3426 }

3427 **XML media type:** application/CIMI-EventLog+xml

3428 **XML serialization:**

```

3429 <EventLog xmlns="http://www.dmtf.org/cimi">
3430   <self> xs:anyURI </self>
3431   <name> xs:string </name> ?
3432   <description> xs:string </description> ?
3433   <created> xs:string </created>
3434   <property name="xs:string"> xs:string </property> *
3435   <targetEntity href="xs:anyURI"/>
3436   <event href="xs:anyURI"/> *
3437   <persistence> xs:string </persistence>
3438   <summary>
3439     <low> xs:integer </low>
3440     <medium> xs:integer </medium>
3441     <high> xs:integer </high>
3442     <critical> xs:integer </critical>
3443   </summary>
3444   <operation rel="edit" href="xs:anyURI"/> ?
3445   <operation rel="delete" href="xs:anyURI"/> ?
3446   <xs:any>*
3447 </EventLog>
    
```

3448 **5.13.7.1 Operations**

3449 This entity supports the Read, Update and Delete operations.

3450 **5.13.8 Event Log Collection**

3451 A Event Log Collection entity represents the collection of Event Logs within a Provider. This resource can
 3452 be used to locate EventLogs.

Name	EventLogCollection	
Type URI	http://www.dmtf.org/cimi/EventLogCollection	
Attribute	Type	Description
eventLogs	ref[]	An array of references to the set of Event Logs in the Provider. Properties: Optional / Mutable

3453 The following describes the serialization of the entity in both JSON and XML:

3454 **JSON media type:** application/CIMI-EventLogCollection+json

3455 **JSON serialization:**

```

3456 { "self": string,
3457   "name": string, ?
3458   "description": string, ?
3459   "created": string, ?
3460   "properties": { "name": string, + }, ?
3461   "eventLogs": [
3462     { "href": string }, +
3463   ], ?
3464   "operations": [
3465     { "rel": "edit", "href": string } ?
3466   ] ?
    
```



```
3467     ...
3468 }
```

3469 **XML media type:** application/CIMI-EventLogCollection+xml

3470 **XML serialization:**

```
3471 <EventLogCollection xmlns="http://www.dmtf.org/cimi">
3472   <self> xs:anyURI </self>
3473   <name> xs:string </name> ?
3474   <description> xs:string </description> ?
3475   <created> xs:string </created>
3476   <property name="xs:string"> xs:string </property> *
3477   <eventLog href="xs:anyURI"/> *
3478   <operation rel="edit" href="xs:anyURI"/> ?
3479   <xs:any>*
3480 </EventLogCollection>
```

3481 5.13.8.1 Operations

3482 This entity supports the Read and Update operations.

3483 5.13.9 Event

3484 An entity that represents the notification of an event within the managed infrastructure. Some examples of
3485 Events may be:

- 3486 • Machine X has been rebooted by guest OS
- 3487 • Machine X is not responding to platform services
- 3488 • A new vCPU has been added to machine X following defined elasticity rules

3489 The scope of the Event concept is any kind of information that the Provider is able to track within its
3490 infrastructure and that can constitute useful information for the consumer. Possible examples, but not
3491 limited to, are errors and inconveniences that occur in the (virtual) resources assigned to consumers,
3492 some provider initiated actions such as maintenance tasks, etc.

Name	Event	
Type URI	http://www.dmtf.org/cimi/Event	
Attribute	Type	Description
eventTime	dateTimeUTC	The time and date of creation of the Event. Properties: Mandatory / Immutable
type	string	A value that indicates the kind of Event (informational, error, alarm, etc.). This attribute is read-only . Properties: Mandatory / Mutable
severity	string	A value indicating the Event severity. Possible values are: critical, high, medium or low. This attribute is read-only . Properties: Mandatory / Mutable
contact	string	An optional identifier that references a contact point to solve the problem (helpdesk, technical staff, etc.). This attribute is read-only .

		Properties: Optional / Mutable
correlatedEvents	ref[]	A list of Event references whose notifications are correlated with (related to) this one. Properties: Optional / Mutable

3493 The following describes the serialization of the entity in both JSON and XML:

3494 **JSON media type:** application/CIMI-Event+json

3495 **JSON serialization:**

```

3496 { "self": string,
3497   "name": string, ?
3498   "description": string, ?
3499   "created": string, ?
3500   "properties": { "name": string, + }, ?
3501   "eventTime": string,
3502   "type": string,
3503   "severity": string,
3504   "contact": string, ?
3505   "correlatedEvents": [
3506     { "href": string }, +
3507   ], ?
3508   "operations": [
3509     { "rel": "edit", "href": string }, ?
3510     { "rel": "delete", "href": string } ?
3511   ] ?
3512   ...
3513 }
```

3514 **XML media type:** application/CIMI-Event+xml

3515 **XML serialization:**

```

3516 <Event xmlns="http://www.dmtf.org/cimi">
3517   <self> xs:anyURI </self>
3518   <name> xs:string </name> ?
3519   <description> xs:string </description> ?
3520   <created> xs:string </created>
3521   <property name="xs:string"> xs:string </property> *
3522   <eventTime> xs:dateTime </eventTime>
3523   <type> xs:string </type>
3524   <severity> xs:string </severity>
3525   <contact> xs:string </contact> ?
3526   <correlatedEvent href="xs:anyURI"/> *
3527   <operation rel="edit" href="xs:anyURI"/> ?
3528   <operation rel="delete" href="xs:anyURI"/> ?
3529   <xs:any>*
3530 </Event>
```

3531 5.13.9.1 Operations

3532 This entity supports the Read, Update and Delete operations.

3533 5.13.10 Event Collection

3534 An Event Collection entity represents the collection of Events within a Provider. This entity can be used to
3535 locate Events.

Name	EventCollection	
Type URI	http://www.dmtf.org/cimi/EventCollection	
Attribute	Type	Description
events	ref[]	An array of references to the set of Events in the Provider. Properties: Optional / Mutable

3536 The following describes the serialization of the entity in both JSON and XML:

3537 **JSON media type:** application/CIMI-EventCollection+json

3538 **JSON serialization:**

```

3539 { "self": string,
3540   "name": string, ?
3541   "description": string, ?
3542   "created": string, ?
3543   "properties": { "name": string, + }, ?
3544   "events": [
3545     { "href": string }, +
3546   ], ?
3547   "operations": [
3548     { "rel": "edit", "href": string } ?
3549   ] ?
3550   ...
3551 }
```

3552 **XML media type:** application/CIMI-EventCollection+xml

3553 **XML serialization:**

```

3554 <EventCollection xmlns="http://www.dmtf.org/cimi">
3555   <self> xs:anyURI </self>
3556   <name> xs:string </name> ?
3557   <description> xs:string </description> ?
3558   <created> xs:string </created>
3559   <property name="xs:string"> xs:string </property> *
3560   <event href="xs:anyURI"/> *
3561   <operation rel="edit" href="xs:anyURI"/> ?
3562   <xs:any>*
3563 </EventCollection>
```

3564 5.13.10.1 Operations

3565 This entity supports the Read and Update operations.

3566 6 Scenarios

3567 6.1 Initial Scenario

3568 The following scenarios serves as an initial starting point:

3569 A cloud consumer has a machine (e.g. a machine image plus some configuration information) that
3570 she wants to run on virtual resources provided by a cloud. The image for this machine already
3571 exists within the cloud infrastructure. Using that image, she requests the cloud to create a
3572 Machine instance. The request may specify configurable options, such as the number of virtual

3573 CPUs, memory size, disk size, etc. The provider will provision the required resources and deploy
3574 the Machine. The consumer may start and stop the Machine.

3575 This scenario is composed of the following use cases from the “Scoping Framework for Cloud
3576 Management Models and Protocol Requirements” document [ref needed]. Note that all statements to the
3577 effect of “he/she does foo” should be construed as expanding to “he/she causes some piece of
3578 consumer-side software to do foo”.

3579 **6.1.1 Create and deploy a Machine using a Provider created Machine Template**

3580 Steps to create a machine using an existing Machine Template:

- 3581 1. Consumer makes a request to list all machine templates from the Provider.
- 3582 2. The Provider gives the Consumer a list of machine templates that the Consumer has access to.
- 3583 3. The Consumer browses the Machine Templates and gets details of each template. The details of
3584 the template contains the following information:
 - 3585 a. name of the template
 - 3586 b. a textual description of the template – this could include information such as operating
3587 system of the machine image that the template references
 - 3588 c. a unique reference to the template
 - 3589 d. the hardware profile or configuration referenced by the template such as # of CPUs,
3590 assigned disk capacity, and networks that the machine is configured to connect to
 - 3591 e. the name of the image that the template references - this could be the guest operating
3592 system and could include names of applications that may be built into the image by the
3593 Provider
- 3594 4. Consumer selects the template of interest and makes a request to the Provider to create the
3595 machine by providing the following information
 - 3596 a. desired name of the machine
 - 3597 b. some description text for the machine
 - 3598 c. reference to the selected Machine Template
- 3599 5. If the template indicates that the Consumer must provide specialization information such as
3600 administrator password, licensing keys etc, the Consumer supplies the information
- 3601 6. The Provider accepts the request and provides a Job id to the Consumer. The Provider creates
3602 the Machine and deploys it to the Consumer’s cloud.
- 3603 7. The Consumer can use the Job id to track progress of the machine creation.
- 3604 8. Once the Job status shows Job as being completed successfully, the Consumer can start using
3605 the Machine. If the Consumer issues a Machine list query for his/her cloud, the created Machine
3606 is returned in the list.

3607 **6.1.2 Create a Machine by passing a Machine Template by value**

3608 Instead of creating and persisting a MachineTemplate, the Consumer may pass a MachineTemplate by
3609 value. The MachineTemplate may contain, for example:

- 3610 a. reference to a Machine Configuration

- 3611 b. reference to a Machine Image
- 3612 c. reference to a Machine Admin entity
- 3613 d. reference to a Volume Template / Volume Configuration + Volume Image
- 3614 e. reference to a Network Template / Network Configuration

3615 **6.1.3 Create a Machine using a Consumer created Machine Template**

3616 Steps to create a Machine from a Machine Template that has been created by the Consumer:

- 3617 1. Create and save a Machine Template. See the scenarios for creating a Machine Template.
- 3618 2. Create a Machine by passing a reference to a Machine Template.

3619 **6.1.4 Create a Machine Template by specifying individual components**

3620 Consumer can create their own machine templates by browsing available resources. The steps are:

- 3621 1. Consumer chooses the components for the Machine Template, such as:
 - 3622 a. a Machine Configuration
 - 3623 b. a Machine Image
 - 3624 c. a Machine Admin
 - 3625 d. a Volume Template / Volume Configuration + Volume Image
 - 3626 e. a Network Template / Network Configuration
- 3627 2. Consumer makes a request to create a Machine Template by building a MachineTemplate that
3628 contains references to above chosen components. If a Consumer does not specify some
3629 components, the Provider can return an error or use default values.
- 3630 3. The Provider validates that required components are present and creates the Machine Template.
3631 The Provider informs the Consumer of the location of the Machine Template. If the Consumer
3632 makes a request to browse Consumer created Machine Templates, this new template appears in
3633 the list.

3634 **6.1.5 Create a Machine Template from a template file**

- 3635 1. Consumer starts the create template process. He passes a reference to a file that contains
3636 metadata for creating a Machine Template.
- 3637 2. Provider accepts the request and provides a Job id to the Consumer.
- 3638 3. The Provider parses the template file, validates it and creates the template. If required
3639 components are not present, the Provider can return an error or use default values.
- 3640 4. The Provider informs the Consumer of the location of the template. If the Consumer makes a
3641 request to browse Consumer created Machine Templates, this new template appears in the list.

3642 **6.1.6 Control Machine State (CMWG065)**

3643 Our consumer controls the state of her Machine by updating the `status` attribute of the entity
3644 corresponding to that Machine. For example, updating the value to STOPPED would stop the Machine.

3645 **6.2 Machine Image Scenarios**

3646 **6.2.1 Create new Machine Image from an image file**

- 3647 1. Consumer starts the create Machine Image process. She passes a reference to the image file
3648 (e.g. an OVF file).
- 3649 2. Provider accepts the request, validates the request, creates the Machine Image and saves it to
3650 the Consumer's cloud.
- 3651 3. If the Consumer makes a request to browse Consumer created Machine Images, this new image
3652 appears in the list.

3653 **6.2.2 Create new Machine Image from Machine instance**

- 3654 1. Consumer selects a Machine instance from his Machine list
- 3655 2. Consumer requests to make an image of the selected Machine instance by providing a name and
3656 a location to save the image to.
- 3657 3. Provider accepts the request and creates the machine image and saves it to the location provided
3658 by the Consumer
- 3659 4. If the Consumer makes a request to browse Consumer created machine images, this new image
3660 appears in the list

3661 **6.3 System Scenario**

3662 The following scenario serves to illustrate the creation of a System as a composite of Machines, Volumes,
3663 and Networks:

3664 A cloud consumer has a system template (e.g., machine, machine image, network, storage) that
3665 it wants to run on virtual resources provided by a cloud. It will upload the template and request
3666 the cloud to deploy it. The request may specify configurable options, such as the number of
3667 virtual system targets. The provider will provision the required resources and deploy the template.
3668 The provisioned resources will be monitored and data made available to the cloud consumer. The
3669 consumer may start, modify, and stop the system. The consumer will terminate its use when it is
3670 complete.

3671 **6.3.1 List System Templates (CMWG010)**

3672 Since our cloud consumer is smart/lazy, the first thing she does is to verify that the System she wants to
3673 deploy does not already have an existing template in her target Site. By performing a `Read` operation on
3674 her target Site, our consumer obtains a list of references to the System Templates. Subsequent `Read`
3675 operations on the referenced System Templates allows our consumer to browse the available System
3676 Templates, and, via their `name` and `description` attributes, determine if her template is among them.

3677 **6.3.2 Create System Template**

3678 Having ascertained that a template of her desired System does not already exist within her target site, our
3679 consumer creates a System Template by posting a System Template to the System Template Collection.

3680 **6.3.3 Create and Deploy a System to a Site Using a System Template** 3681 **(CMWG017/CMWG035)**

3682 Once her template is created, our consumer creates her System via that entity's `Create` operation using a
3683 reference to the recently created System Template. This action recursively creates Machines, Volumes,

3684 and Networks (along with their interrelationships) for every Machine Template, Volume Template, and
3685 Network Templates contained by the System Template.

3686 Our consumer monitors the progress of the `Create` operation by using the `Read` operation to poll `state`
3687 attribute of the newly created System and/or, optionally, using the `Read` operation to poll an associated
3688 `Job` entity.

3689 **6.3.4 Get Monitoring Information (CMWG066)**

3690 Our consumer monitors her System and its constituent Machines, Volumes, and Networks by `Reading` the
3691 appropriate entity.

3692 **6.3.5 Control System State (CMWG065)**

3693 Our consumer controls the state of her System by `Updating` the `status` attribute of the entity
3694 corresponding to that System. For example, updating the value to `STOPPED` would stop the System.
3695 This action recursively updates the `status` attributes of all Machines contained within her system, with
3696 corresponding effects on the operational state of those machines. Individual Machines can be controlled
3697 in a similar fashion.

3698 **6.3.6 Remove System from a Site (CMWG051)**

3699 When our consumer is finished with her System, she removes it by invoking the `Delete` operation on the
3700 entity corresponding to that System. This action recursively removes the Machines, Volumes, and
3701 Networks contained within her system. The removal of Volumes and Networks would obviously be
3702 contingent on whether or not these entities were shared by other, active Machines.

3703 **7 Security**

3704 This specification considers two separate but related security domains. The first domain, API-level
3705 security, concerns the protection of the entities modeled by this specification. For example, insuring that
3706 unauthorized users are not allowed to alter a Machine instance. The second domain, resource-level
3707 security, deals with the protection of the underlying resources represented by these entities. For example,
3708 insuring that unauthorized users cannot login to the Linux instance corresponding to that Machine.

3709 **7.1 API Level Security**

3710 **7.1.1 Authentication**

3711 Except in cases where the access control policy allows for anonymous requests, the Provider SHALL
3712 authenticate all request messages and determine the identity of the Consumer. The techniques used to
3713 authenticate messages are outside the scope of this specification.

3714 Protocol bindings of the CIMI Model specification are encouraged to include requirements for the most
3715 common authentication mechanisms applicable to that protocol (e.g. the use of `BasicAuth` for protocols
3716 using `HTTP`).

3717 **7.1.2 Message Integrity**

3718 Messages exchanged between the Consumer and the Provider SHOULD have message integrity
3719 protections applied. The mechanisms used to provide message integrity are outside the scope of this
3720 specification.

3721 Protocol bindings of the CIMI Model specification are encouraged to include requirements for the most
3722 common integrity mechanisms applicable to that protocol (e.g. the use of `TLS` for protocols using `HTTP`).

3723 **7.1.3 Message Confidentiality**

3724 Messages exchanged between the Consumer and the Provider MAY have message confidentiality
3725 protections applied. The mechanisms used to provide message confidentiality are outside the scope of
3726 this specification.

3727 **7.1.4 Authorization**

3728 The Provider SHOULD process messages only if authorized by access control policy, which may
3729 reference the Consumer's identity, the message type and content, and other contextual information when
3730 making this decision. The language in which this access control policy is expressed as well as the
3731 process by which these authorization decisions are made are outside the scope of this specification.

3732 **7.1.5 Multi-Tenancy**

3733 In cases where a Provider uses multi-tenancy to support a set of Consumers, the operations in this
3734 specification are modeled under the assumption that each Consumer's view of the system (i.e. which
3735 entities are visible, discoverable, and accessible) is scoped to those entities provisioned for or created by
3736 that Consumer. To the Consumer it appears that the Provider is implementing a sole-use instance of the
3737 CIMI API (albeit one who's non-functional characteristics may be influenced the actions of invisible co-
3738 Consumers).

3739 **7.2 Resource Level Credentials**

3740 This specification intentionally avoids constraining the type, nature, or operation of the resources
3741 represented by the entities that it defines. It is therefore outside the scope of this specification to define
3742 the mechanism(s) used to access the resource represented by the Machine entity. There is, however, an
3743 integration point between this specification and such mechanisms, namely the management of the
3744 credentials (user names, passwords, keys, etc.) used to provision such access. This information is
3745 encapsulated by the Machine Admin entity (described in Section 0).

3746

Bibliography

3747

3748 **DMTF DSP-ISO102**, Distributed Management Task Force, Inc., *Architecture for Managing Clouds White*
3749 *Paper 1.0*, http://dmf.org/sites/default/files/standards/documents/DSP-ISO102_1.0.0.pdf

3750 **DMTF DSP-ISO103**, Distributed Management Task Force, Inc., *Use Cases and Interactions for Managing*
3751 *Clouds 1.0.0*, http://www.dmtf.org/sites/default/files/standards/documents/DSP-ISO103_1.0.0.pdf

3752 **DMTF DSP-ISXXXX**, Distributed Management Task Force, Inc., *Scoping Framework for Cloud*
3753 *Management Models and Protocol Requirements 0.1.5*,
3754 [http://members.dmtf.org/apps/org/workgroup/cmwg/download.php/56339/Cloud%20Management%20Fra](http://members.dmtf.org/apps/org/workgroup/cmwg/download.php/56339/Cloud%20Management%20Framework_v015.doc)
3755 [mework_v015.doc](http://members.dmtf.org/apps/org/workgroup/cmwg/download.php/56339/Cloud%20Management%20Framework_v015.doc)