1

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

2
3

An Interface for Managing Cloud Infrastructure

4

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: 2012-03-17

Provide any comments through the DMTF Feedback Portal:
0Hhttp://www.dmtf.org/standards/feedback

Version: 0.0.35
Status: Work In Progress - not a DMTF Standard
Publication Date: 2011-09-14
Document Number: DSP0263

5
6
7
8
9

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

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26   withdrawn or modified after publication, and shall be indemnified and held harmless by any party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

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

32   **Abstract**

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

37   **Acknowledgments**

38   TBD

CONTENTS

155

FIGURES

156

162

163   1   Scope

164   Quoting the "Architecture for Managing Clouds White Paper":

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

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

170   1.1   Document Structure

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

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

174   2   References

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

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

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

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

185   **IETF RFC 2246**, T. Dierks and C. Allen, *The TLS Protocol Version 1.0*, Janury 1999,
186   http://www.ietf.org/rfc/rfc2246.txt

187   **IETF RFC 3986**, T.Berners-Lee et al, *Uniform Resource Identifiers (URI): Generic Syntax*, August 1998,
188   http://www.ietf.org/rfc/rfc3986.txt

189   **IETF RFC 4288**, N. Freed and J. Klensin, *Media Type Specifications and Registration Procedures*,
190   December 2005, http://www.ietf.org/rfc/rfc4288.txt

191   **IETF RFC 4346**, T. Dierks and E. Rescorla, *The Transport Layer Security (TLS) Protocol Version 1.1*,
192   April 2006, http://www.ietf.org/rfc/rfc4346.txt

193   **IETF RFC 4627**, D. Crockford, *The application/json Media Type for JavaScript Object Notation (JSON)*,
194   July 2006, http://www.ietf.org/rfc/rfc4627.txt

195   **IETF RFC 5246**, T. Dierks and E. Rescorla, *The Transport Layer Security (TLS) Protocol Version 1.1*,
196   http://www.ietf.org/rfc/rfc5246.txt

197   **ISO 8601:20044**, International Organization for Standardization, Geneva, Switzerland, *Data elements and*
198   *interchange formats -- Information interchange -- Representation of dates and times*, March 2008,
199   http://www.iso.org/iso/iso_catalogue/ catalogue_tc/catalogue_detail.htm?csnumber=40874

200   **ITU-T X.509**, Telecommunication Standardization Sector of ITU, *Information technology - Open Systems*
201   *Interconnection - The Directory: Public- key and attribute certificate frameworks*, November 2008,
202   http://www.itu.int/rec/T-REC-X.509-200811-I

203   **NIST Special Publication 800-57**, Elaine Barker et al, *Recommendation for Key Management – Part 1:*
204   *General (Revised)*, March 2007, http://csrc.nist.gov/publications/nistpubs/800-57/sp800-57-Part1-
205   revised2_Mar08-2007.pdf

206   **NIST Special Publication 800-131A,** Elaine Barker and Allen Roginsky, *Transitions: Recommendation*
207   *for Transitioning the Use of Cryptographic Algorithms and Key Lengths*, January 2011,
208   http://csrc.nist.gov/publications/nistpubs/800-131A/sp800-131A.pdf

209   **Representational State Transfer**, Roy Fielding, Doctoral dissertation, University of California,
210   *Architectural Styles and the Design of Network-based Software Architectures (Chapter 5)*, 2000,
211   http://www.ics.uci.edu/~fielding/pubs/dissertation/rest_arch_style.htm

212   **XMLSchema - Part 1**, World Wide Web Consortium (W3C) Recommendation, H. Thompson, et al.,
213   Editors, *XML Schema Part 1: Structures Second Edition*, 28 October 2004,
214   http://www.w3.org/TR/xmlschema-1/

215   **XMLSchema - Part 2**, World Wide Web Consortium (W3C) Recommendation, P. Biron, A. Malhotra,
216   Editors, *XML Schema Part 2: Datatypes (Second Edition)* , 28 October 2004,
217   http://www.w3.org/TR/xmlschema-2/

218   **DMTF DSP-0243**, Distributed Management Task Force, Inc., *Open Virtualization Format Specification*
219   *1.1.0*, http://www.dmtf.org/sites/default/files/standards/documents/DSP0243_1.1.0.pdf

220   **DMTF DSP-0259**, Distributed Management Task Force, Inc., *Cloud Infrastructure Management Interface -*
221   *CIM Model (CIMI-CIM) 0.0.1*, http://members.dmtf.org/apps/org/workgroup/cmwg/download.php/yyyy

222   **DMTF DSP-XXXX**, Distributed Management Task Force, Inc., *Cloud Infrastructure Management Interface*
223   *- RelaxNG Model (CIMI-RNG) 0.0.1,*
224   *http://members.dmtf.org/apps/org/workgroup/cmwg/download.php/zzz*

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

3   Terms and Definitions

3.1   Authentication

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

3.2   Authorization

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

3.3   Cloud Service Consumer

235   A category of actors that includes the Consumer Business Manager (who approves business and
236   financial expenditures for consumed services, accounts for used service instances, establishes business
237   relationships; sets up accounts, budget, and terms; etc.), the Consumer Service Administrator (who
238   requests service instances and changes to service instances, purchase services within the business
239   relationship; create Service Users (including policies), allocate resources, such as compute and storage,
240   generate reports (usage), etc.), and Service Users (who uses service instances provided by a Cloud

241 Service Provider). The term "**Consumer**" is used when the indicated action or activity could involve one or
242 more of the above actors. In cases where the distinction between the actors in this category is relevant,
243 the more detailed term will be used.

3.4   Cloud Service Provider

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

3.5   Configuration

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

3.6   Message Confidentiality

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

3.7   Message Integrity

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

3.8   Template

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

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

273 1. By reference - allow Consumers to reference a Template (that exists as an entity in the Provider) as
274   part of the instantiation operation.

275 2. By value - allow Consumers to dynamically provide the Template information as part of the
276   instantiation operation.

277 3. Reference with overrides - allow Consumers to reference a Template (that exists as an entity in the
278   Provider) and provide additional values that override the attributes of that Template as part of the
279   instantiation operation.

280   **4   REST/HTTP Protocol**

281   **4.1   Protocol Definition**

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

287   The entities in the system are identified by URIs. Dereferencing (via an HTTP GET) the URI of an entity
288   will yield a representation of the entity containing attributes and links to associated entities. To begin
289   operations, a client must know the URI to the main entry point of a Cloud Provider - also known as the
290   "Cloud Entry Point" entity. All other entities within the environment shall then be discoverable via the
291   iterative following of links to associated resource within each resource retrieved.

292   **4.1.1   Protocol Security**

293   Cloud Providers SHALL support secure HTTP connections using TLS. Cloud Providers MAY support non-
294   secure HTTP connections. TLS 1.0, which shall be implemented, is specified in [RFC2246], and the TLS
295   1.1 and TLS 1.2 should be implemented as specified in [RFC4346] and [RFC5246], respectively.

296   To ensure a minimum level of security and interoperability between implementations, all CIMI clients and
297   servers shall support the TLS_DHE_DSS_WITH_3DES_EDE_CBC_SHA cipher suite (hexadecimal value
298   {0x0013}), which is also the mandatory cipher suite for TLS 1.0 (see [RFC2246] Section 9, Mandatory
299   Cipher Suites), as well as the TLS_RSA_WITH_AES_128_CBC_SHA cipher suite (hexadecimal value
300   {0x002F}) shall be implemented, which is the mandatory cipher suite for both TLS 1.1 and TLS 1.2. In
301   addition, the TLS_RSA_WITH_NULL_SHA cipher suite (hexadecimal value {0x0002}) shall be supported
302   by both CIMI clients and servers to implement authenticated, non-encrypted communications. Finally, the
303   TLS_RSA_WITH_AES_128_CBC_SHA256 cipher suite (hexadecimal value {0x003C}) should be
304   included with all recommended TLS 1.2 implementations to meet the transition to a security strength of
305   112 bits (guidance is provided in NIST Special Publication 800-57 [NIST 800-57]and NIST Special
306   Publication 800-131A [NIST 800-131A]). Implementers are free to include additional cipher suites, but
307   must prefer the mandatory ones in negotiation.

308   **4.1.2   XML Namespaces**

309   The following table lists the XML namespaces that are used in this specification. The choice of any
310   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**

311   **4.1.3   URI Space**

312   URIs returned by providers are to be treated as opaque by client implementations. Clients MUST NOT
313   make assumptions about the layout of the URIs or the structures of the URIs of the resources.

314   Editor's Note: This may not be true if we defined well-known query parameters so we need to revisit this

315 **4.1.4 Media Types**

316 In this specification, resource representations and request bodies are encoded in either JSON, as
317 specified in [RFC4627] or in XML.

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

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

326 **4.1.5 Request Headers**

327 This specification uses general-header, request-header, and entity-header headers as defined in HTTP
328 1.1 [RFC2616] in request messages to provide metadata about the message. Applications using
329 messages defined in this specification shall use headers consistent with HTTP 1.1.

330 In addition to headers defined in HTTP 1.1, request messages may include a header defined by this
331 specification to indicate the version of the CIMI API that the client used to send the message.

```
332    X-CIMI-Specification-Version = "X-CIMI-Specification-Version" ":" api-version
333    api-version                 = "1.0"
```

334 **4.1.6 Request Parameters**

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

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

338   **Table: Request Parameters**

Format	Description	Example
?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	MyMachine?name,description,status Would return only "name", "description", "status" attributes of the Assembly132.
?[attr_regex]	Attribute regular expression. If none of the resource attributes match the pattern, then the resource would be returned in its complete form. <resource_uri> is equivalence to <resource_uri>?[.*]	MyMachine?[.*contain.*] Would return contained_in and container_type 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

?<attri1>:[collapse]	This would return only attri1, and if attri1 is a Collection, it would be collapsed. If attri1 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 collections	For example, /assembly123?[verbose] would return the expanded list of all the components, including all the attributes

339   The client must URL encode the request parameters.

4.1.7   Response Headers

341   This specification uses general-header, response-header, and entity-header headers as defined in HTTP
342   1.1 [RFC2616] in response messages to provide metadata about the message. Applications using
343   messages defined in this specification shall use headers consistent with HTTP 1.1.

344   In addition to headers defined in HTTP 1.1, response messages shall include a header defined by this
345   specification to indicate the URI for the job created to process the associated request message.
346   X-CIMI-Job-URI = "X-CIMI-Job-URI" ":" *string*

347   If a job was not created to process the request message this header shall not be included in the response
348   message.

4.1.8   HTTP Status Codes

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

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

354   **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:

	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.
202 Accepted	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.
	A service implementing this specification SHOULD return a Job entity as well as a 'X-CIMI-Job-URI' HTTP header indicating the URI of the Job entity itself.
	Per the HTTP/1.1 specification:
	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
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

355 4.1.9 Serialization of References

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

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

360 and this in XML:

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

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

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

366 and this in XML:

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

368 however, for brevity they are excluded.

369 4.1.10 Serialization of Arrays

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

```
374          "volumes" : [
375            { "volume": { "href": string },
376              "attachmentPoint": string,
377              "protocol": string } +
378          ], ?
```

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

382 4.2   Protocol Resource Operations

383 This section defines the set of common REST/HTTP operations that a Cloud Provider might expose. At its
384 core there are four basic CRUD (Create, Read, Update and Delete) operations. The manner in which
385 these are used is consistent across all resources within the model; therefore, their use is defined once
386 and is to be applied consistently. Some resources support specialized operations that do not fit well into
387 a CRUD style of operation and those will all follow a similar high-level pattern but each operation is
388 allowed to have slight variations to accommodate its specific needs. The specifics of these special
389 operations are detailed within the section that defines the resource.

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

397 4.2.1 Common CRUD (Create Read Update and Delete) Operations

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

401 4.2.1.1 Creating a new Resource

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

405 The request will be of the following form:

```
406     POST <addURI> HTTP/1.1
407     Host: ...
408     Accept: application/CIMI-...
409     Content-Type: application/CIMI-...
410     X-CIMI-Specification-Version: 1.0
411
412     <serialization of request to create a new resource>
```

413 The following provides additional constraints on the request message:
414   **X-CIMI-Specification-Version**
415   This REQUIRED HTTP header specifies the version of this specification that is being used.

416 The response will be of the following form:

```
417     HTTP/1.1 201 Created
418     Location: ...
419     Content-Type: application/CIMI-...
420     X-CIMI-Specification-Version: 1.0
421
422     <serialization of new resource>
```

423 The following provides additional constraints on the response message:
424   **X-CIMI-Specification-Version**
425   This REQUIRED HTTP header specifies the version of this specification that is being used.

426 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.

427   4.2.1.2 Reading a Resource

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

429   The request will be of the following form:

```
430        GET <ResourceURI> HTTP/1.1
431        Host: ...
432        Accept: application/CIMI-...
433        X-CIMI-Specification-Version: 1.0
```

434   The following provides additional constraints on the request message:
435   **X-CIMI-Specification-Version**
436   This REQUIRED HTTP header specifies the version of this specification that is being used.

437   The response will be of the following form:

```
438        HTTP/1.1 200 OK
439        Content-Type: application/CIMI-...
440        X-CIMI-Specification-Version: 1.0
441
442        <serialization of resource>
```

443   The following provides additional constraints on the response message:
444   **X-CIMI-Specification-Version**
445   This REQUIRED HTTP header specifies the version of this specification that is being used.

446   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.

447   4.2.1.3 Updating a Resource

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

452   The request will be of the following form:

```
453        PUT <editURI> HTTP/1.1
454        Host: ...
455        Accept: application/CIMI-...
456        Content-Type: application/CIMI-...
457        X-CIMI-Specification-Version: 1.0
458
459        <serialization of request to update a resource>
```

460   The following provides additional constraints on the request message:

461   **X-CIMI-Specification-Version**
462   This REQUIRED HTTP header specifies the version of this specification that is being used.

463   The response will be of the following form:

```
464      HTTP/1.1 200 OK
465      Content-Type: application/CIMI-...
466      X-CIMI-Specification-Version: 1.0
467
468      <serialization of updated resource>
```

469   The following provides additional constraints on the response message:

470   **X-CIMI-Specification-Version**
471   This REQUIRED HTTP header specifies the version of this specification that is being used.

472   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.

473   **4.2.1.3.1 Partial Updates to a Resource**

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

478   `http://example.com/resource?attribute1,attribute2,...`

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

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

484   For clarity, the resource referenced by this modified URI is a distinct resource from the original one; it just
485   happens to be a subset of the original resource - one that contains just the attributes listed in the query
486   parameters. The semantics of a normal HTTP PUT are still adhered to - it is a complete replacement
487   update of the specified resource.

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

```
489      PUT /machines/myMachine?name,description HTTP/1.1
490      Host: ...
491      Accept: application/CIMI-Machine
492      Content-Type: application/CIMI-Machine
493      X-CIMI-Specification-Version: 1.0
494
495      <Machine>
496        <name>My New Machine</name>
497      </Machine>
```

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

499   **4.2.1.4 Deleting a Resource**

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

504   The request will be of the following form:

```
505       DELETE <deleteURI> HTTP/1.1
506       Host: ...
507       X-CIMI-Specification-Version: 1.0
```

508   The following provides additional constraints on the request message:

509   **X-CIMI-Specification-Version**
510   This REQUIRED HTTP header specifies the version of this specification that is being used.

511   The response will be of the following form:

```
512       HTTP/1.1 200 OK
513       X-CIMI-Specification-Version: 1.0
```

514   The following provides additional constraints on the response message:

515   **X-CIMI-Specification-Version**
516   This REQUIRED HTTP header specifies the version of this specification that is being used.

517   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.

518   **4.2.1.5 Other Operations**

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

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

525   The request will be of the following form:

```
526       POST <operationLinkURI> 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 perform some action>
```

533   The following provides additional constraints on the request message:

534   **X-CIMI-Specification-Version**
535   This REQUIRED HTTP header specifies the version of this specification that is being used.

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

537   Note that the definition of the "Create" operation (see section 4.2.1.1) follows this same pattern - it is just
538   called out for ease of reference.

4.2.1.6 Synchronous Operations

540   If a Provider supports the Job entity then each incoming PUT, DELETE, POST request SHALL result in a
541   Job entity being created and a reference to that Job entity SHALL be returned back to the client via the X-
542   CIMI-Job-URI HTTP Header in the HTTP response message:
543   `X-CIMI-Job-URI: <uri-to-Job>`

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

4.2.1.7 Asynchronous Operations

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

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

5   Model

561   This model assumes that a business relationship has already been established between the Cloud
562   Consumer and the Cloud Provider. This relationship may include financial terms, creating separately
563   administered clouds that the consuming organization is paying for, and the establishment of
564   authentication credentials to access the administrative entry point for each cloud. This scope of this
565   model is one separately administered cloud.

5.1   Alternative Model Formats

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

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

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

572   5.2   Entity Metadata

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

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

Data	Type	Description
uri	URI	The unique identifier of this entity; assigned upon entity creation. This attribute value is **immutable**, and should be **unique** in the provider's cloud.
typeURI	URI	A unique URI associated with, and denoting, this entity type.
name	string	The name of the entity type.
attributeMetadata		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:

Data	Type	Description
name	string	The name of the attribute.
namespace	URI	The namespace in which this attribute is defined. It is recommended that a deference 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.
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.
required	boolean	Indicates whether this entity requires this attribute to be present. When absent the implied value is "false".
constraints		Type specific data that describes the constraints of this attribute. When absent the there are no constraints.

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

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

580   **JSON serialization:**
581   { "uri": *string*,
582   "typeURI": *URI*,
583   "name": *string*,
584   "attributes" : [
585   { "name": "*string*",
586   "namespace": "*string*", ?
587   "type": "*string*", ?

```
588       "required": boolean, ?
589       ...constraints...? } +
590    ] ?
591    ...
592  }
```

XML media type: application/CIMI-EntityMetadata+xml

XML serialization:
```
<EntityMetadata xmlns="http://www.dmtf.org/cimi">
  <uri> xs:anyURI </uri>
  <name> xs:string </name>
  <typeURI> xs:anyURI </typeURI>
  <attribute name="xs:string" namespace="xs:anyURI"? type="xs:string"
             required="xs:boolean"? >
    ...constraints...?
  </attribute> *
  <xs:any>*
</EntityMetadata>
```

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

5.2.1   Attribute Types

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

type="string"

The JSON SHALL be of the form:
```
"values": [ "string", + ] ?
```

The XML SHALL be of the form:
```
<value> xs:string </value> *
```

type="integer"

The JSON SHALL be of the form:
```
"values": [ integer, + ], ?
"ranges": [ { "low": integer, "high": integer }, + ] ?
```

The XML SHALL be of the form:
```
<value> xs:integer </value> *
<range low="xs:integer" high="xs:integer"/> *
```

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

type="boolean"

The JSON SHALL be of the form:
```
"value": boolean ?
```

The XML SHALL be of the form:
```
<value> xs:boolean </value> ?
```

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

5.2.2 Examples

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

```
<EntityMetadata xmlns="http://www.dmtf.org/cimi">
  <uri> http://example.org/types/VC </uri>
  <typeURI> http://www.dmtf.org/cimi/VolumeConfiguration </typeURI>
  <name> VolumeConfiguration </name>
  <attribute name="Location" namespace="http://example.org/" type="string"/>
</EntityMetadata>
```

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

```
<EntityMetadata xmlns="http://www.dmtf.org/cimi">
  <uri> http://example.org/types/VC </uri>
  <typeURI> http://www.dmtf.org/cimi/VolumeConfiguration </typeURI>
  <name> VolumeConfiguration </name>
  <attribute name="Location" namespace="http://example.org/" type="string"
             required="true">
    <value> NYC </value>
    <value> LAX </value>
  </attribute>
</EntityMetadata>
```

The following shows the same VolumeConfiguration serialized in JSON:

```
{ "uri": "http://example.org/types/VC",
  "typeURI": "http://www.dmtf.org/cimi/VolumeConfiguration",
  "name": "VolumeConfiguration",
  "attributes": [
    { "name": "Location",
      "namespace": "http://example.org",
      "type": "string",
      "required": true,
      "values": [ "NYC", "LAX" ]
  }
}
```

5.3 Entities

The following sections detail the attributes of the entities depicted in the CIMI model.

Note that the format and syntax of the attributes of type "URI" is defined by RFC 3986 [RFC3986] with the following, additional constraints: Relative URIs MUST start with a "/", otherwise the URI is assumed to be absolute and no URI processing (to determine the full path) will be performed. Relative URIs are interpreted as being relative to the root URI of the CloudEntryPoint.

Attributes are either "immutable" (their values are fixed for the lifetime of the entity), or "mutable" (their values may change over the lifetime of the entity). Unless otherwise noted, all attributes are mutable. Mutable attributes are either "writeable" (their value may be changed by the Consumer) or "read-only" (their value may only be changed by the Provider). Unless otherwise noted, all mutable attributes are writeable.

671   ### 5.3.1   Identifiers

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

674   • Identifier names shall be treated as case sensisitive

675   • Identifier names shall only use the following set of characters:

676   o   Upper case ASCII (U+0041 through U+005A)

677   o   Lower case ASCII (U+061 through U+007A)

678   o   Digits (U+0030 through U+0039)

679   o   Underscore (U+005F)

680   ### 5.3.2   Common Attributes

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

Attribute	Type	Description	Optionality
uri	URI	The unique identifier of this entity; assigned upon entity creation. This attribute value is **immutable**, and should be **unique** in the provider's cloud.	mandatory
name	string	The human readable name of this entity; assigned by the creator as a part of the entity creation input.	optional
description	string	The human readable description of this entity; assigned by the creator as a part of the entity creation input.	optional
created	DateTimeUTC	The timestamp when this entity was created. The format should be unambiguous, and the value is **immutable**.	optional
properties		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. 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.	optional

682   #### 5.3.2.1 Units

683   Some of the entities defined by this specification have attributes that describe an amount of something
684   that belongs to, or is associated with that entity. For example, the `Machine` entity has a `memory` attribute
685   which describes "the size of the memory allocated to this machine". This specification adopts the
686   convention of representing such attributes via a duple consisting of a `quantity` (represented as an
687   integer) and `units` (represented as a string). The allowable values for `units` are listed in the following
688   table. Their meaning is defined in IEC 80000-13:2008 [IEC 80000-13:2008]. Their numerical equivalents
689   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
yottabye	10^24	yobibyte	2^80

690   5.4   Cloud Entry Point

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

694   **Type URI:** http://www.dmtf.org/cimi/CloudEntryPoint

Attribute	Type	Description	Optionality
systemTemplates	URI	A reference to the System Template Collection of this CloudEntry Point.	optional
systems	URI	A reference to the System Collection of this Cloud Entry Point.	optional
machineTemplates	URI	A reference to the Machine Template Collection of this Cloud Entry Point.	optional
machineConfigs	URI	A reference to the Machine Configuration Collection of this Cloud Entry Point.	optional
machineImages	URI	A reference to the Machine Image Collection of this Cloud Entry Point.	optional
machineAdmins	URI	A reference to the Machine Admin Collection of this Cloud Entry Point.	optional
machines	URI	A reference to the Machine Collection of this Cloud Entry Point.	optional
volumeTemplates	URI	A reference to the Volume Template Collection of this Cloud Entry Point.	optional
volumeConfigs	URI	A reference to the Volume Configuration Collection of this Cloud Entry Point.	
volumeImages	URI	A reference to the Volume Image Collection of this Cloud Entry Point.	

volumes	URI	A reference to the Volume Collection of this Cloud Entry Point.	optional
networkTemplates	URI	A reference to the Network Template Collection of this Cloud Entry Point.	optional
networkConfigs	URI	A reference to the Network Configuration Collection of this Cloud Entry Point.	optional
networks	URI	A reference to the Network Collection of this Cloud Entry Point.	optional
vspTemplates	URI	A reference to the VSP Template Collection of this Cloud Entry Point.	optional
vspConfigs	URI	A reference to the VSP Configuration Collection of this Cloud Entry Point.	optional
vsps	URI	A reference to the VSP Collection of this Cloud Entry Point.	optional
meterTemplates	URI	A reference to the Meter Template Collection of this Cloud Entry Point.	optional
meters	URI	A reference to the Meter Collection of this Cloud Entry Point.	optional
eventLogs	URI	A reference to the Event Log Collection of this Cloud Entry Point.	optional
events	URI	A reference to the Event Collection of this Cloud Entry Point	optional
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.	optional
entityMetadata	map	A map of references to the metadata for each entity supported by the Provider. The "name" portion of the map is the "Type URI" of the entity, and the "value" portion of the map is a URI to the metadata resource of the entity. If an entity does not have any metadata then it will not appear in this 'map' - e.g. it has no constraints beyond what the CIMI specification defines nor does it have any extension attributes.	optional

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

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

697   **JSON serialization:**

```
698        { "uri": string,
699          "name": string, ?
700          "description": string, ?
701          "created": string,  ?
702          "properties": [ "name": string, + ], ?
703          "systemTemplates": { "href": string }, ?
704          "systems": { "href": string }, ?
705          "machineTemplates": { "href": string }, ?
706          "machineConfigs": { "href": string }, ?
707          "machineImages": { "href": string }, ?
708          "machineAdmins": { "href" string }, ?
```

```
709                "machines": { "href": string }, ?
710                "volumeTemplates": { "href": string }, ?
711                "volumeConfigs": { "href": string }, ?
712                "volumeImages": { "href": string }, ?
713                "volumes": { "href": string }, ?
714                "networkTemplates": { "href": string }, ?
715                "networkConfigs": { "href": string }, ?
716                "networks": { "href": string }, ?
717                "vspTemplates": { "href": string }, ?
718                "vspConfigs": { "href": string }, ?
719                "vsps": { "href": string }, ?
720                "meterTemplates": { "href": string }, ?
721                "meters": { "href": string }, ?
722                "eventLogs": { "href": string }, ?
723                "events": { "href": string }, ?
724                "job_time": number, ?
725                "entityMetadata": [
726                  { "rel": "entity-type-uri", "href": string }, +
727                ], ?
728                "operations": [
729                  { "rel": "edit", "href": string }, ?
730                ] ?
731                ...
732              }
```

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

734   **XML serialization:**

```
735              <CloudEntryPoint xmlns="http://www.dmtf.org/cimi">
736                <uri> xs:anyURI </uri>
737                <name> xs:string </name> ?
738                <description> xs:string </description> ?
739                <created> xs:dateTime </created>
740                <property name="xs:string"> xs:string </property> *
741                <systemTemplates href="xs:anyURI"/> ?
742                <systems href="xs:anyURI"/> ?
743                <machineTemplates href="xs:anyURI"/> ?
744                <machineConfigs href="xs:anyURI"/> ?
745                <machineImages href="xs:anyURI"/> ?
746                <machineAdmins href="xs:anyURI"/> ?
747                <machines href="xs:anyURI"/> ?
748                <volumeTemplates href="xs:anyURI"/> ?
749                <volumeConfigs href="xs:anyURI"/> ?
750                <volumeImages href="xs:anyURI"/> ?
751                <volumes href="xs:anyURI"/> ?
752                <networkTemplates href="xs:anyURI"/> ?
753                <networkConfigs href="xs:anyURI"/> ?
754                <networks href="xs:anyURI"/> ?
755                <vspTemplates href="xs:anyURI"/> ?
756                <vspConfigs href="xs:anyURI"/> ?
757                <vsps href="xs:anyURI"/> ?
758                <meterTemplates href="xs:anyURI"/> ?
759                <meters href="xs:anyURI"/> ?
760                <eventLogs href="xs:anyURI"/> ?
761                <events href="xs:anyURI"/> ?
762                <job_time> xs:integer </job_time>
763                <entityMetadata typeURI="entity-type-uri " href="xs:anyURI"/> *
764                <operation rel="edit" href="xs:anyURI"/> ?
765                <xs:any>*
766              </CloudEntryPoint>
```

767 **5.4.1   Operations**

768 This entity supports the Read and Update operations.

769 **5.5   System Entities and Relationships**

770 The following diagram illustrates the entities involved in constructing a System and their relationships.
771 Although this drawing is in the style of an Entity Relationship diagram, the use of UML is neither rigorous
772 nor normative.
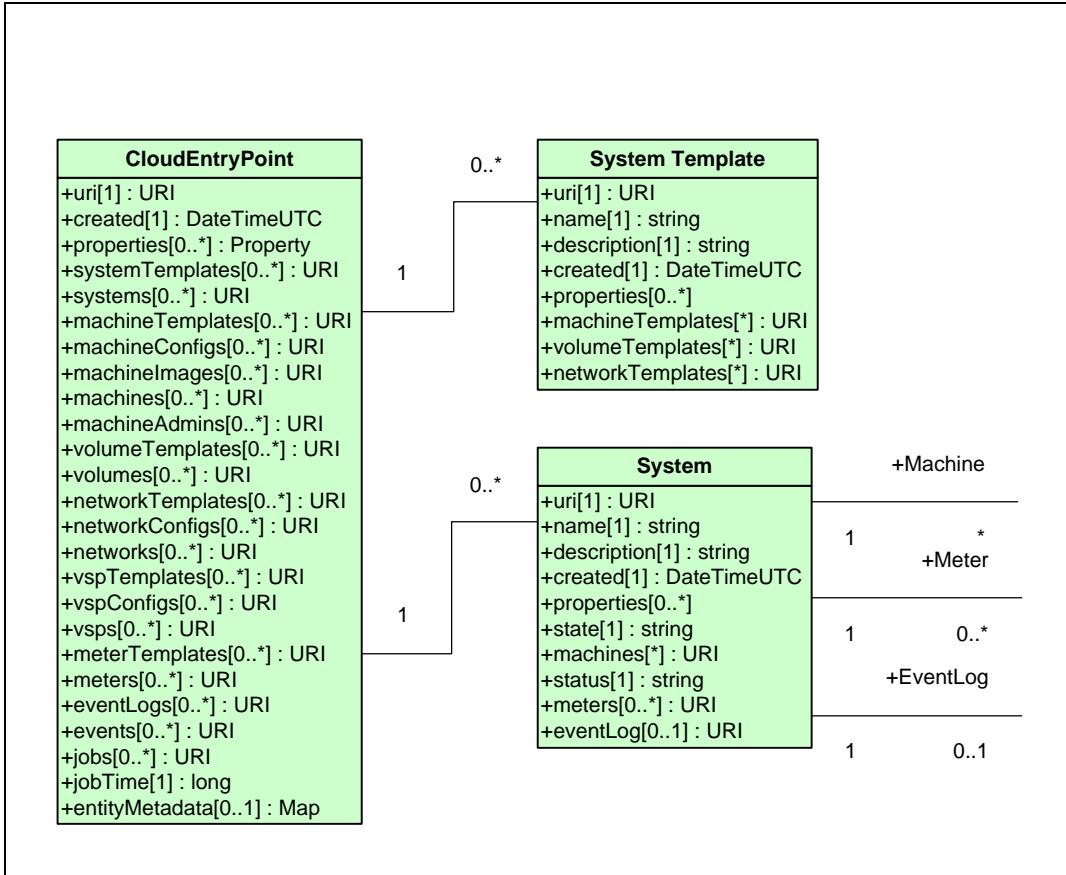
773

774 **Figure 1 - System Entities**

775 **5.5.1   System Template**

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

778 **Type URI:** http://www.dmtf.org/cimi/SystemTemplate

Attribute	Type	Description	Optionality
volumeTemplates	URI[]	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.	optional
machineTemplates	URI[]	List of Machine Templates referenced in this System Template. Items in this list are used to create Machines as	optional

		part of the System creation process.	
networkTemplates	URI[]	List of NetworkTemplates referenced in this System Template. Items in this list are used to create Networks as part of the System creation process.	optional

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

780   **JSON serialization:**

781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
```
{ "uri": string,
  "name": string, ?
  "description": string, ?
  "created": string,  ?
  "properties": [ "name": string, + ], ?
  "volumeTemplates": [
    { "href": string }, +
  ], ?
  "machineTemplates": [
    { "href": string }, +
  ], ?
  "networkTemplates": [
    { "href": string }, +
  ], ?
  "operations": [
    { "rel": "edit", "href": string }, ?
    { "rel": "delete", "href": string } ?
  ] ?
  ...
}
```

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

802   **XML serialization:**

803
804
805
806
807
808
809
810
811
812
813
814
815
```
<SystemTemplate xmlns="http://www.dmtf.org/cimi">
  <uri> xs:anyURI </uri>
  <name> xs:string </name> ?
  <description> xs:string </description> ?
  <created> xs:string </created>
  <property name="xs:string"> xs:string </property> *
  <volumeTemplate href="xs:anyURI"/> *
  <machineTemplate href="xs:anyURI"/> *
  <networkTemplate href="xs:anyURI"/> *
  <operation rel="edit" href="xs:anyURI"/> ?
  <operation rel="delete" href="xs:anyURI"/> ?
  <xs:any>*
</SystemTemplate>
```

816   **5.5.1.1 Operations**

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

819   **5.5.2   System Template Collection**

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

822   **Type URI:** http://www.dmtf.org/cimi/SystemTemplateCollection

Attribute	Type	Description	Optionality
systemTemplates	URI[]	An array of URIs referencing the set of System Templates in the Provider.	optional

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

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

825   **JSON serialization:**

```
826          { "uri": string,
827            "name": string, ?
828            "description": string, ?
829            "created": string,  ?
830            "properties": [ "name": string, + ], ?
831            "systemTemplates": [
832              { "href": string }, +
833            ], ?
834            "operations": [
835              { "rel": "add", "href": string }, ?
836              { "rel": "edit", "href": string } ?
837            ] ?
838            ...
839          }
```

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

841   **XML serialization:**

```
842          <SystemTemplateCollection xmlns="http://www.dmtf.org/cimi">
843            <uri> xs:anyURI </uri>
844            <name> xs:string </name> ?
845            <description> xs:string </description> ?
846            <created> xs:string </created>
847            <property name="xs:string"> xs:string </property> *
848            <systemTemplate href="xs:anyURI"/> *
849            <operation rel="add" href="xs:anyURI"/> ?
850            <operation rel="edit" href="xs:anyURI"/> ?
851            <xs:any>*
852          </SystemTemplateCollection>
```

853   **5.5.2.1 Operations**

854   This entity supports the Read and Update operations. Creation of new System Template entities is
855   supported via a POST to the "addLink" URI as described in section 4.2.1.1.

856   **5.5.3   System**

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

863   **Type URI:** http://www.dmtf.org/cimi/System

Attribute	Type	Description	Optionality

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.	mandatory
machines	URI[]	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 to this System.	optional
status	string	Indicates the operational status of the entity. For example, STARTED, STOPPED, SUSPENDED are operational status. Updating this value amounts to controlling the status of the System. For example, updating the value to STOPPED would indicate that theSystem should be stopped from whichever status it is currently at. Changing the attribute value may result in a long running operation.	mandatory
meters	URI[]	A list of references to Meters monitored for this System.	optional
eventLog	URI	A reference to the EventLog of this System.	optional

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

865   **JSON serialization:**

```
866        { "uri": string,
867          "name": string, ?
868          "description": string, ?
869          "created": string,   ?
870          "properties": [ "name": string, + ], ?
871          "state": string,
872          "machines": [
873            { "href": string }, +
874          ], ?
875          "status": string,
876          "meters": [
877            { "href": string }, +
878          ], ?
879          "eventLog": { "href": string }, ?
880          "operations": [
881            { "rel": "edit", "href": string }, ?
882            { "rel": "delete", "href": string } ?
883          ] ?
884          ...
885        }
```

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

887   **XML serialization:**

```
888        <System xmlns="http://www.dmtf.org/cimi">
889          <uri> xs:anyURI </uri>
890          <name> xs:string </name> ?
891          <description> xs:string </description> ?
892          <created> xs:string </created>
893          <property name="xs:string"> xs:string </property> *
```

```
894        <state> xs:string </state>
895        <machine href="xs:anyURI"/> *
896        <status> xs:string </status>
897        <meter href="xs:anyURI"/> *
898        <eventLog href="xs:anyURI"/> ?
899        <operation rel="edit" href="xs:anyURI"/> ?
900        <operation rel="delete" href="xs:anyURI"/> ?
901        <xs:any>*
902      </System>
```

5.5.3.1 Operations

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

5.5.4   System Collection

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

Type URI: http://www.dmtf.org/cimi/SystemCollection

Attribute	Type	Description	Optionality
systems	URI[]	An array of URIs referencing the set of Systems in the Provider.	optional

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

JSON media type: application/CIMI-SystemCollection+json

JSON serialization:

```
913      { "uri": string,
914        "name": string, ?
915        "description": string, ?
916        "created": string,  ?
917        "properties": [ "name": string, + ], ?
918        "systems": [
919          { "href": string }, +
920        ], ?
921        "operations": [
922          { "rel": "add", "href": string }, ?
923          { "rel": "edit", "href": string } ?
924        ] ?
925        ...
926      }
```

XML media type: application/CIMI-SystemCollection+xml

XML serialization:

```
929      <SystemCollection xmlns="http://www.dmtf.org/cimi">
930        <uri> xs:anyURI </uri>
931        <name> xs:string </name> ?
932        <description> xs:string </description> ?
933        <created> xs:string </created>
934        <property name="xs:string"> xs:string </property> *
935        <system href="xs:anyURI"/> *
936        <operation rel="add" href="xs:anyURI"/> ?
937        <operation rel="edit" href="xs:anyURI"/> ?
938        <xs:any>*
939      </SystemCollection>
```

940   **5.5.4.1 Operations**

941   This entity supports the Read and Update operations.

942   The following custom operations are also defined:

943   • **Creating a New System**

944   **/link@rel:** add

945   This operation will create a new System.

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

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

948   • **HTTP/REST Protocol**

949   To create a new System a POST is sent to the "add" URI of the SystemCollection where the HTTP
950   request body SHALL be as described below. Note this structure allows for certain properties to be
951   passed in "by value" or by "reference". The definition of each property can be found in section 5.4.1.

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

953   **JSON serialization:**

954
```
{ "name": string,
  "description": string, ?
  "properties": [ "name": string, + ], ?
  "systemTemplate" : { "href": string , ?
    "properties": [ "name": string, + ], ?
    "volumeTemplates": [
      { "href": string, ?
        "properties": [ "name": string, + ], ?
        "volumeConfig": { "href": string, ?
          "properties": [ "name": string, + ], ?
          "format": string, ?
          "capacity": { "quantity": number, "units": string }, ?
          "supportsSnapshots": boolean, ?
          "guestInterface": string, ?
        },
        "volumeImage": { "href": string,
          "imageLocation": { "href": string }, ?
          "imageData": string, ?
          "bootable": boolean ?
        } ?
      }, +
    ], ?
    "machineTemplates": [
      { "href": string, ?
        "properties": [ "name": string, + ], ?
        "machineConfig": { "href": string, ?
          "cpu": string, ?
          "memory": { "quantity": integer, "units": string }, ?
          "disks" : [
            { "capacity": { "quantity": integer, "units": string },
              "guestInterface": string }, +
          ] ?
        },
        "machineImage": { "href": string,
          "imageLocation": { "href": string }, ?
          "imageData": string, ?
        }, ?
```

```
991              "machineAdmin": { "href": string, ?
992                <provider specific data> ?
993              }, ?
994              "volumes": [
995                { "href": string, "attachmentPoint": string, "protocol": string}, +
996              ], ?
997              "volumeTemplates": [
998                { "href": string, ?
999                  "attachmentPoint": string, "protocol": string,
1000                 "volumeConfig": { "href": string, ?
1001                   "properties": [ "name": string, + ], ?
1002                   "format": string, ?
1003                   "capacity": { "quantity": number, "units": string }, ?
1004                   "supportsSnapshots": boolean, ?
1005                   "guestInterface": string ?
1006                 }, ?
1007                 "volumeImage": { "href": string,
1008                   "properties": [ "name": string, + ], ?
1009                   "imageLocation": { "href": string }, ?
1010                   "imageData": string, ?
1011                   "bootable": boolean ?
1012                 } ?
1013               }, +
1014             ] ?
1015           }
1016         ], ?
1017         "networkTemplates": [
1018           { "networkTemplate": { "href": string, ?
1019               "properties": [ "name": string, + ] ?
1020             <add network properties when defined>
1021           }
1022         ], ?
1023       }
1024       ...
1025     }
```

1026 **XML media type:** application/CIMI-SystemCreate+xml

1027 **XML serialization**

```
1028         <SystemCreate>
1029           <name> xs:string </name>
1030           <description> xs:string </description> ?
1031           <property name="xs:string"> xs:string </property> *
1032           <systemTemplate href="xs:anyURI"?>
1033             <property name="xs:string"> xs:string </property> *
1034
1035           <volumeTemplate href="xs:anyURI"? >
1036             <property name="xs:string"> xs:string </property> *
1037             <volumeConfig href="xs:anyURI"? >
1038               <property name="xs:string"> xs:string </property> *
1039               <format> xs:string </format> ?
1040               <capacity quantity="xs:integer" units="xs:string"/> ?
1041               <supportsSnapshots> xs:boolean </supportsSnapshots> ?
1042               <guestInterface> xs:string </guestInterface> ?
1043             </volumeConfig>
1044             <volumeImage href="xs:anyURI">
1045               <property name="xs:string"> xs:string </property> *
1046               <imageLocation href="xs:anyURI"/> ?
1047               <imageData> xs:any* </imageData> ?
1048               <bootable> xs:boolean </bootable> ?
1049             </volumeImage> ?
1050           </volumeTemplate> *
1051
```

```
1052          <machineTemplate href="xs:anyURI"? >
1053            <property name="xs:string"> xs:string </property> *
1054            <machineConfig href="xs:anyURI"? >
1055              <property name="xs:string"> xs:string </property> *
1056              <cpu> xs:string </cpu> ?
1057              <memory quantity="xs:integer" units="xs:string"/>
1058              <disk>
1059                <capacity quantity="xs:integer" units="xs:string">
1060                <guestInterface> xs:string </guestInterface>
1061              </disk> *
1062            </machineConfig>
1063            <machineImage href="xs:anyURI">
1064              <property name="xs:string"> xs:string </property> *
1065              <imageLocation href="xs:anyURI"/> ?
1066              <imageData> xs:string </imageData> ?
1067            </machineImage>
1068            <machineAdmin href="xs:anyURI"? >
1069              xs:any*  <!-- provider specific data -->
1070            </machineAdmin> ?
1071            <volume href="xs:anyURI"
1072                    attachmentPoint="xs:string" protocol="xs:string" /> *
1073            <volumeTemplate href="xs:anyURI"?
1074                        attachmentPoint="xs:string" protocol="xs:string" >
1075              <property name="xs:string"> xs:string </property> *
1076              <volumeConfig href="xs:anyURI">
1077                <property name="xs:string"> xs:string </property> *
1078                <format> xs:string </format> ?
1079                <capacity quantity="xs:integer" units="xs:string"/> ?
1080                <supportsSnapshots> xs:boolean </supportsSnapshots> ?
1081                <guestInterface> xs:string </guestInterface> ?
1082              </volumeConfig>
1083              <volumeImage href="xs:anyURI">
1084                <property name="xs:string"> xs:string </property> *
1085                <imageLocation href="xs:anyURI"/> ?
1086                <imageData> xs:any* </imageData> ?
1087                <bootable> xs:boolean </bootable> ?
1088              </volumeImage> ?
1089            </volumeTemplate> *
1090          </machineTemplate> *
1091
1092          <networkTemplate href="xs:anyURI"? >
1093            <property name="xs:string"> xs:string </property> *
1094            <add network properties when defined>
1095          </networkTemplate> *
1096
1097        </systemTemplate>
1098        <xs:any>*
1099      </SystemCreate>
```

1100　The serialization of some reference properties are specified such that a request MAY either include a
1101　reference ("href") to an existing entity or to include the entity "inline" as a set of additional properties.
1102　Requests SHALL NOT include both a reference and the inlined set of properties.

1103　Upon successful processing of the request, the HTTP response body MAY either be empty or contain a
1104　serialization of the System entity.

1105　**5.6　Machine Entities and Relationships**

1106　The following diagram illustrates the entities involved in constructing a Machine and their relationships.
1107　Although this drawing is in the style of an Entity Relationship diagram, the use of UML is neither rigorous
1108　nor normative.
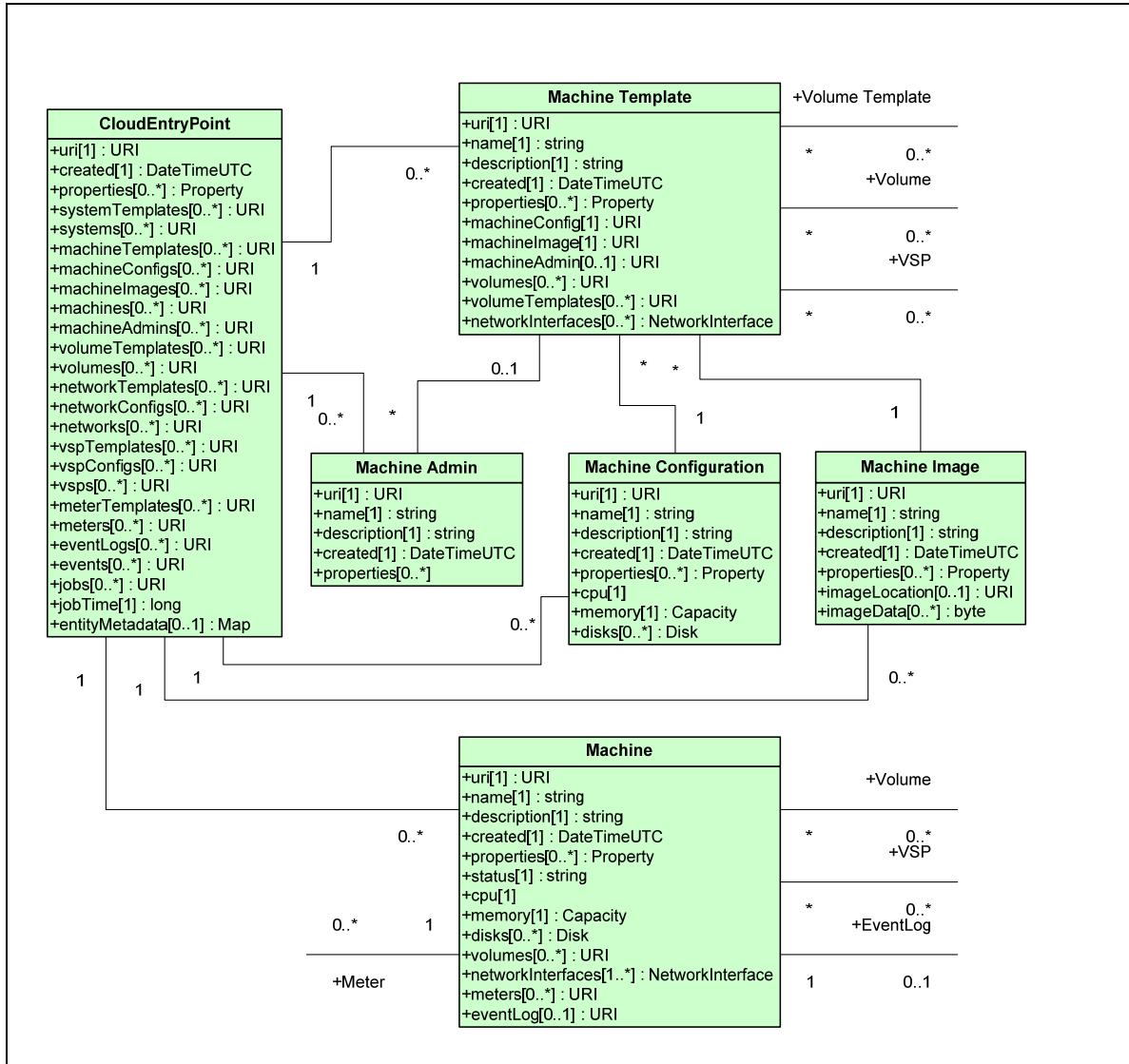
1109

1110   **Figure 2 - Machine Entities**

1111   ## 5.6.1   Machine Template

1112   A Machine Template represents the set of metadata and instructions used in the creation of a Machine.

1113   **Type URI:** http://www.dmtf.org/cimi/MachineTemplate

Attribute	Type	Description	Optionality
machineConfig	URI	A reference to the Machine Configuration that will be used to create a Machine from this Machine Template.	mandatory
machineImage	URI	A reference to the Machine Image that will be used to create a Machine from this Machine Template.	mandatory
machineAdmin	URI	A reference to the Machine Admin that will be used to create the initial login credential for the new Machine.	optional

volumes	A list of references to existing Volumes that will be attached to the Machine during its creation.			optional
	Each volume attribute has the following sub-attributes which describe aspects of the way in which the Machine will be attached to the Volume:			

Attribute	Type	Description
attachmentPoint	String	File system path where the Volume will be attached
protocol	String	Protocol that will be used to access this Volume (e.g. NFS, iSCSI)
volume	URI	Reference to the Volume that will be attached.

volumeTemplates	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.			optional
	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 `volumeTemplates` attribute of the relevant System Template. If the same Volume Template URI is listed in both the `volumeTemplates` attribute of a System Template and in the `volumeTemplates` 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.			
	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:			

Attribute	Type	Description
attachmentPoint	String	File system path where the Volume will be attached
protocol	String	Protocol that will be used to access this Volume (e.g. NFS, iSCSI)
volumeTemplate	URI	Reference to the Volume Template that will be used to create a new Volume.

networkInterfaces	A list of sub-entities that define the network interfaces that will be created on Machines instantiated from this template.			optional

Attribute	Type	Description
vsp	URI	A reference to the VSP (Virtual Switch Port) for this network interface.
hostname	string	DNS resolvable name associated with this network interface.

		macAddress	string	Address assigned by the hypervisor when a machine is created or a unique address can be manually assigned.	
		state	string	The state of an interface configurable to be "Active" or "Standby".	
		protocol	string	Selected network protocol such as - IPv4 or IPv6.	
		allocation	string	The option for "Dynamic Host Allocation Protocol" or static.	
		address	string	The IP address assigned to a virtual interface.	
		defaultGateway	string	An IP address to a firewall or router that serves other networks.	
		dns	string	The IP address of the Domain Name Service from host name to IP resolution.	
		maxTransmissionUnit	integer	To set the largest supported packet size.	

1114   The following describes the serialization of the entity in both JSON and XML:

1115   **JSON media type:** application/CIMI-MachineTemplate+json

1116   **JSON serialization:**

```
1117        { "uri": string,
1118          "name": string,
1119          "description": string, ?
1120          "created": string,  ?
1121          "properties": [ "name": string, + ], ?
1122          "machineConfig": { "href": string },
1123          "machineImage": { "href": string },
1124          "machineAdmin": { "href": string }, ?
1125          "volumes": [
1126            { "href": string, "attachmentPoint": string, "protocol": string }, +
1127          ], ?
1128          "volumeTemplates": [
1129            { "href": string, "attachmentPoint": string, "protocol": string }, +
1130          ], ?
1131          "networkInterfaces": [
1132            { "vsp": {"href": string}, "hostname": string, "macAddress": string,
1133              "state": string, "protocol": string, "allocation": string,
1134              "address": string, "defaultGateway": string, "dns": string,
1135              "maxTransmissionUnit": integer }, +
1136          ], ?
1137          "operations": [
1138            { "rel": "edit", "href": string }, ?
1139            { "rel": "delete", "href": string } ?
1140          ] ?
1141          ...
1142        }
```

1143   **XML media type:** application/CIMI-MachineTemplate+xml

1144   **XML serialization:**

```
1145            <MachineTemplate xmlns="http://www.dmtf.org/cimi">
1146              <uri> xs:anyURI </uri>
1147              <name> xs:string </name>
1148              <description> xs:string </description> ?
1149              <created> xs:string </created>
1150              <property name="xs:string"> xs:string </property> *
1151              <machineConfig href="xs:anyURI"/>
1152              <machineImage href="xs:anyURI"/>
1153              <machineAdmin href="xs:anyURI"/> ?
1154              <volume href="xs:anyURI"
1155                      attachmentPoint="xs:string" protocol="xs:string" /> *
1156              <volumeTemplate href="xs:anyURI"
1157                          attachmentPoint="xs:string" protocol="xs:string" /> *
1158              <networkInterface>
1159                <vsp href="xs:anyURI"/>
1160                <hostname> xs:string </hostname>
1161                <macAddress> xs:string </macAddress>
1162                <state> xs:string </state>
1163                <protocol> xs:string </protocol>
1164                <allocation> xs:string </alloction>
1165                <address> xs:string </address>
1166                <defaultGateway> xs:string </defaultGateway>
1167                <dns> xs:string </dns>
1168                <maxTransmissionUnit> xs:integer </maxTransmissionUnit>
1169              </networkInterface> *
1170              <operation rel="edit" href="xs:anyURI"/> ?
1171              <operation rel="delete" href="xs:anyURI"/> ?
1172              <xs:any>*
1173            </MachineTemplate>
```

1174   **5.6.1.1 Operations**

1175   This entity supports the Read, Update and Delete operations. Create is supported via the Machine
1176   Template Collection entity.

1177   **5.6.2   Machine Template Collection**

1178   A Machine Template Collection entity represents the collection of Machine Template entities within a
1179   Provider. This entity can be used to locate and create Machine Templates.

1180   **Type URI:** http://www.dmtf.org/cimi/MachineTemplateCollection

Attribute	Type	Description	Optionality
machineTemplates	URI[]	An array of URIs referencing the set of Machine Templates in the Provider.	optional

1181   The following describes the serialization of the entity in both JSON and XML:

1182   **JSON media type:** application/CIMI-MachineTemplateCollection+json

1183   **JSON serialization:**

```
1184            { "uri": string,
1185              "name": string, ?
1186              "description": string, ?
1187              "created": string,  ?
```

```
1188            "properties": [ "name": string, + ], ?
1189            "machineTemplates": [
1190              { "href": string }, +
1191            ], ?
1192            "operations": [
1193              { "rel": "add", "href": string }, ?
1194              { "rel": "edit", "href": string } ?
1195            ] ?
1196            ...
1197          }
```

1198   **XML media type:** application/CIMI-MachineTemplateCollection+xml

1199   **XML serialization:**

```
1200          <MachineTemplateCollection xmlns="http://www.dmtf.org/cimi">
1201            <uri> xs:anyURI </uri>
1202            <name> xs:string </name> ?
1203            <description> xs:string </description> ?
1204            <created> xs:string </created>
1205            <property name="xs:string"> xs:string </property> *
1206            <machineTemplate href="xs:anyURI"/> *
1207            <operation rel="add" href="xs:anyURI"/> ?
1208            <operation rel="edit" href="xs:anyURI"/> ?
1209            <xs:any>*
1210          </MachineTemplateCollection>
```

1211   5.6.2.1 Operations

1212   This entity supports the Read and Update operations. Creation of new Machine Template entities is
1213   supported via a POST to the "addLink" URI as described in section 4.2.1.1.

1214   5.6.3   Machine Configuration

1215   The Machine Configuration entity represents the set of configuration values that define the (virtual)
1216   hardware resources of a to-be-realized Machine Instance. Machine Configurations are created by
1217   Providers and MAY, at the Providers discretion, be created by Consumers.

1218   **Type URI:** http://www.dmtf.org/cimi/MachineConfiguration

Attribute	Type	Description	Optionality
cpu	*TBD*	Indicate the amount of CPU (based on standard CPU measurement) that a Machine realized from this configuration would have, by default.	mandatory
memory	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><tr><td>Attribute</td><td>Type</td><td>Description</td></tr><tr><td>quantity</td><td>integer</td><td>A numerical quantity expressed as an integer.</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**.</td></tr></table>	mandatory

disks	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.	mandatory
	Each disks attribute has the following sub-attributes:	

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.
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**.

Attribute	Type	Description
guestInterface	string	This property indicates the interface offered to a Machine instance to gain access to the storage contents.

1219　**JSON media type:** application/CIMI-MachineConfiguration+json

1220　**JSON serialization:**

```
1221        { "uri": string,
1222          "name": string,
1223          "description": string, ?
1224          "created": string,  ?
1225          "properties": [ "name": string, + ], ?
1226          "cpu": string,
1227          "memory": { "quantity": integer, "units": string },
1228          "disks" : [
1229            { "capacity": { "quantity": integer, "units": string },
1230              "guestInterface": string }, +
1231          ], ?
1232          "operations": [
1233            { "rel": "edit", "href": string }, ?
1234            { "rel": "delete", "href": string } ?
1235          ] ?
1236          ...
1237        }
```

1238　**XML media type:** application/CIMI-MachineConfiguration+xml

1239　**XML serialization:**

```
1240        <MachineConfiguration xmlns="http://www.dmtf.org/cimi">
1241          <uri> xs:anyURI </uri>
1242          <name> xs:string </name>
1243          <description> xs:string </description> ?
1244          <created> xs:string </created>
1245          <property name="xs:string"> xs:string </property> *
1246          <cpu> xs:string </cpu>
1247          <memory quantity="xs:integer" units="xs:string"/>
```

```
1248            <disk>
1249              <capacity quantity="xs:integer" units="xs:string">
1250              <guestInterface> xs:string </guestInterface>
1251            </disk> *
1252            <operation rel="edit" href="xs:anyURI"/> ?
1253            <operation rel="delete" href="xs:anyURI"/> ?
1254            <xs:any>*
1255          </MachineConfiguration>
```

5.6.3.1 Operations

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

5.6.4 Machine Configuration Collection

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

Type URI: http://www.dmtf.org/cimi/MachineConfigurationCollection

Attribute	Type	Description	Optionality
machineConfigurations	URI[]	An array of URIs referencing the set of Machine Configurations in the Provider.	optional

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

JSON media type: application/CIMI-MachineConfigurationCollection+json

JSON serialization:

```
{ "uri": string,
  "name": string, ?
  "description": string, ?
  "created": string,  ?
  "properties": [ "name": string, + ], ?
  "machineConfigurations": [
    { "href": string }, +
  ], ?
  "operations": [
    { "rel": "add", "href": string }, ?
    { "rel": "edit", "href": string } ?
  ] ?
  ...
}
```

XML media type: application/CIMI-MachineConfigurationCollection+xml

XML serialization:

```
<MachineConfigurationCollection xmlns="http://www.dmtf.org/cimi">
  <uri> xs:anyURI </uri>
  <name> xs:string </name> ?
  <description> xs:string </description> ?
  <created> xs:string </created>
  <property name="xs:string"> xs:string </property> *
  <machineConfiguration href="xs:anyURI"/> *
  <operation rel="add" href="xs:anyURI"/> ?
  <operation rel="edit" href="xs:anyURI"/> ?
  <xs:any>*
```

```
1292          </MachineConfigurationCollection>
```

5.6.4.1 Operations

1293

1294   This entity supports the Read and Update operations. Creation of new Machine Configuration entities is
1295   supported via a POST to the "addLink" URI as described in section 4.2.1.1.

5.6.5   Machine Image

1296

1297   This entity represents the information (e.g. an Open Virtualization Format (OVF) package) necessary for
1298   hardware virtualized resources to create a Machine Instance; it contains configuration data such as
1299   startup instructions, including possible combinations of the following:

1300   •   the software image (i.e. a copy of an installed Machine) which is to be instantiated on the disk
1301   and other virtual resources

1302   •   installation software, which, when executed on the hardware (virtual) resources, builds the
1303   machine instance

1304   •   both a disk image and a set of software and parameters in order to install new components not
1305   included in original disk image

1306   **Type URI:** http://www.dmtf.org/cimi/MachineImage

Attribute	Type	Description	Optionality
imageLocation	URI	A URI that references the location of the binary data that makes up this image. Either this attribute must be present or the imageData attribute must be present.* This value is **imutable.**	optional
imageData	byte[]	The binary data that makes up this image. Either this attribute must be present or the imageLocation attribute must be present.* This value is **imutable**.	optional

1307   * Either imageLocation or imageData SHALL be present, however both values shall not be present
1308   simultaneously.

1309   **JSON media type:** application/CIMI-MachineImage+json

1310   **JSON serialization:**

```
1311          { "uri": string,
1312            "name": string,
1313            "description": string, ?
1314            "created": string,  ?
1315            "properties": [ "name": string, + ], ?
1316            "imageLocation": { "href": string }, ?
1317            "imageData": string, ?
1318            "operations": [
1319              { "rel": "edit", "href": string }, ?
1320              { "rel": "delete", "href": string } ?
1321            ] ?
1322            ...
1323          }
```

1324   **XML media type:** application/CIMI-MachineImage+xml

1325   **XML serialization:**

```
1326        <MachineImage xmlns="http://www.dmtf.org/cimi">
1327          <uri> xs:anyURI </uri>
1328          <name> xs:string </name>
1329          <description> xs:string </description> ?
1330          <created> xs:string </created>
1331          <property name="xs:string"> xs:string </property> *
1332          <imageLocation href="xs:anyURI"/> ?
1333          <imageData> xs:string </imageData> ?
1334          <operation rel="edit" href="xs:anyURI"/> ?
1335          <operation rel="delete" href="xs:anyURI"/> ?
1336          <xs:any>*
1337        </MachineImage>
```

1338   ### 5.6.5.1 Operations

1339   This entity supports the Read, Update and Delete operations. Create is supported via the Machine Image
1340   Collection entity.

1341   ## 5.6.6   Machine Image Collection

1342   A Machine Image Collection entity represents the collection of Machine Image entities within a Provider.
1343   This entity can be used to locate and create Machine Images.

1344   **Type URI:** http://www.dmtf.org/cimi/MachineImageCollection

Attribute	Type	Description	Optionality
machineImages	URI[]	An array of URIs referencing the set of Machine Images in the Provider.	optional

1345   The following describes the serialization of the entity in both JSON and XML:

1346   **JSON media type:** application/CIMI-MachineImageCollection+json

1347   **JSON serialization:**

```
1348        { "uri": string,
1349          "name": string, ?
1350          "description": string, ?
1351          "created": string,  ?
1352          "properties": [ "name": string, + ], ?
1353          "machineImages": [
1354            { "href": string }, +
1355          ], ?
1356          "operations": [
1357            { "rel": "add", "href": string }, ?
1358            { "rel": "edit", "href": string } ?
1359          ] ?
1360          ...
1361        }
```

1362   **XML media type:** application/CIMI-MachineImageCollection+xml

1363   **XML serialization:**

```
1364        <MachineImageCollection xmlns="http://www.dmtf.org/cimi">
1365          <uri> xs:anyURI </uri>
1366          <name> xs:string </name> ?
```

```
1367        <description> xs:string </description> ?
1368        <created> xs:string </created>
1369        <property name="xs:string"> xs:string </property> *
1370        <machineImage href="xs:anyURI"/> *
1371        <operation rel="add" href="xs:anyURI"/> ?
1372        <operation rel="edit" href="xs:anyURI"/> ?
1373        <xs:any>*
1374      </MachineImageCollection>
```

1375 **5.6.6.1 Operations**

1376 This entity supports the Read and Update operations. Creation of new Machine Image entities is
1377 supported via a POST to the "addLink" URI as described in section 4.2.1.1.

1378 **5.6.7   Machine**

1379 An instantiated compute resource that encapsulates both CPU and Memory.

1380 **Type URI:** http://www.dmtf.org/cimi/Machine

Attribute	Type	Description	Optionality
status	string	Indicates the operational status of the entity. For example, STARTED, STOPPED, SUSPENDED are operational status. Updating the value amounts to controlling the status of the machine. For example, updating value to STOPPED would indicate that the machine should be stopped from whichever status it is currently at. Changing the attribute value may result in a long running operation.	mandatory
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.	mandatory
memory		The size of the memory (RAM) allocated to this Machine. When this value is increased, it implies that the Machine is allocated more RAM, and vice versa when the value is decreased. This attribute has the following sub-attributes which serve to describe it: <table><tr><td>Attribute</td><td>Type</td><td>Description</td></tr><tr><td>quantity</td><td>integer</td><td>A numerical quantity expressed as an integer.</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**.</td></tr></table>	mandatory
disks		The list of disks (local storages) that are part of the Machine. Adding an element to this list creates a disk.	optional

disk:

Attribute	Type	Description
capacity		The capacity of the disk described by this attribute. Indicates the capacity of the disk described by this attribute. This attribute has the following, sub-attributes. <table><tr><th>Attribute</th><th>Type</th><th>Description</th></tr><tr><td>quantity</td><td>integer</td><td>A numerical quantity expressed as an integer.</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**.</td></tr></table>
format	string	The format/type of this disk (e.g. ext4, NTFS)
attachmentPoint	string	File system path where this disk is attached.
guestInterface	string	This property indicates the interface offered to a Machine instance to gain access to the storage contents.

volumes	The list of networked volumes that are attached to this Machine. 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. Each volume attribute has the following sub-attributes which describe aspects of the way in which the Machine is attached to the Volume: <table><tr><th>Attribute</th><th>Type</th><th>Description</th></tr><tr><td>attachmentPoint</td><td>string</td><td>File system path where this volume is attached</td></tr><tr><td>protocol</td><td>string</td><td>Protocol used to access this volume (e.g. NFS, iSCSI)</td></tr><tr><td>volume</td><td>URI</td><td>Reference to the attached Volume.</td></tr></table>	optional
networkInterfaces	A list of sub-entities that define the network interfaces on this Machine. <table><tr><th>Attribute</th><th>Type</th><th>Description</th></tr><tr><td>vsp</td><td>URI</td><td>A reference to the VSP (Virtual Switch Port) for this network interface.</td></tr><tr><td>hostname</td><td>string</td><td>DNS resolvable name associated with this network interface.</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.</td></tr></table>	mandatory

		state	string	The state of an interface configurable to be "Active" or "Standby".	
		protocol	string	Selected network protocol such as - IPv4 or IPv6.	
		allocation	string	The option for "Dynamic Host Allocation Protocol" or static.	
		address	string	The IP address assigned to a virtual interface.	
		deafultGateway	string	An IP address to a firewall or router that serves other networks.	
		dns	string	The IP address of the Domain Name Service from host name to IP resolution.	
		maxTransmissionUnit	integer	To set the largest supported packet size.	
meters	URI[]		A list of references to Meters monitored for this Machine.		optional
eventLog	URI		A reference to the EventLog of this Machine.		optional

1381 The following describes the serialization of the entity in both JSON and XML:

1382 **JSON media type:** application/CIMI-Machine+json

1383 **JSON serialization:**

```
1384    { "uri": string,
1385      "name": string,
1386      "description": string, ?
1387      "created": string,  ?
1388      "properties": [ "name": string, + ], ?
1389      "status": string,
1390      "networkInterfaces": [
1391        { "network": { "href": string }, "address": string, "hostname", string }, +
1392      ], ?
1393      "cpu": string,
1394      "memory": { "quantity": integer, "units": string },
1395      "disks" : [
1396        { "capacity": { "quantity": integer, "units": string },
1397          "format": string, "attachmentPoint": string, "guestInterface": string } +
1398      ], ?
1399      "volumes" : [
1400        { "volume": { "href": string },
1401          "attachmentPoint": string,
1402          "protocol": string } +
1403      ], ?
1404      "networkInterfaces": [
1405        { "vsp": {"href": string}, "hostname": string, "macAddress": string,
1406          "state": string, "protocol": string, "allocation": string,
1407          "address": string, "defaultGateway": string, "dns": string,
1408          "maxTransmissionUnit": integer }, +
1409      ], ?
1410      "meters": [
1411        { "href": string }, +
1412      ], ?
1413      "eventLog": { "href": string }, ?
```

```
1414            "operations": [
1415              { "rel": "edit", "href": string }, ?
1416              { "rel": "delete", "href": string }, ?
1417              { "rel": "http://www.dmtf.org/cimi/stop", "href": string } ?
1418            ]
1419            ...
1420          }
```

1421   **XML media type:** application/CIMI-Machine+xml

1422   **XML serialization:**

```
1423          <Machine xmlns="http://www.dmtf.org/cimi">
1424            <uri> xs:anyURI </uri>
1425            <name> xs:string </name>
1426            <description> xs:string </description> ?
1427            <created> xs:string </created>
1428            <property name="xs:string"> xs:string </property> *
1429            <status> xs:string </status>
1430            <networkInterface network="xs:anyURI" address="xs:string"
1431                            hostname="xs:string"> *
1432            <cpu> xs:string </cpu>
1433            <memory quantity="xs:integer" units="xs:string"/>
1434            <disk>
1435              <capacity quantity="xs:integer" units="xs:string"/>
1436              <format> xs:string </format>
1437              <attachmentPoint> xs:string </attachmentPoint>
1438              <guestInterface> xs:string </guestInterface>
1439            </disk> *
1440            <volume href="xs:anyURI" attachmentPoint="xs:string" protocol="xs:string"/> *
1441            <networkInterface>
1442              <vsp href="xs:anyURI"/>
1443              <hostname> xs:string </hostname>
1444              <macAddress> xs:string </macAddress>
1445              <state> xs:string </state>
1446              <protocol> xs:string </protocol>
1447              <allocation> xs:string </alloction>
1448              <address> xs:string </address>
1449              <defaultGateway> xs:string </defaultGateway>
1450              <dns> xs:string </dns>
1451              <maxTransmissionUnit> xs:integer </maxTransmissionUnit>
1452            </networkInterface> *
1453            <meter href="xs:anyURI"/> *
1454            <eventLog href="xs:anyURI"/> ?
1455            <operation rel="edit" href="xs:anyURI"/> ?
1456            <operation rel="delete" href="xs:anyURI"/> ?
1457            <operation rel="http://www.dmtf.org/cimi/stop" href="xs:anyURI"/> ?
1458            <xs:any>*
1459          </Machine>
```

1460   **5.6.7.1 Operations**

1461   This entity supports the Read, Update and Delete operations. Create is supported via the Machine
1462   Collection entity.

1463   The following custom operations are also defined:

1464   • **Stopping a Machine**

1465   **/link@rel:** http://www.dmtf.org/cimi/stop

1466   This operation will stop a Machine.

1467   Input parameters: None.

1468   Output parameters: None.

1469   Upong successful completion of this operation the Machine will be in the "STOPPED" state.

1470   • **HTTP/REST Protocol**

1471   To stop a Machine a POST is sent to the "http://www.dmtf.org/cimi/stop" URI of the Machine where the
1472   HTTP request body SHALL be as described below.

1473   **JSON media type:** application/CIMI-Action+json

1474   **JSON serialization:**

```
1475        { "action": "http://www.dmtf.org/cimi/action/stop" ,
1476          "properties": [ "name": string, + ] ?
1477          ...
1478        }
```

1479   **XML media type:** application/CIMI-Action+xml

1480   **XML serialization**

```
1481        <Action xmlns="http://www.dmtf.org/cimi">
1482          <action> http://www.dmtf.org/cimi/action/stop </action>
1483          <property name="xs:string"> xs:string </property> *
1484          <xs:any>*
1485        </Action>
```

1486   Upon successful processing of the request, the HTTP response body will be empty.

1487   ### 5.6.8   Machine Collection

1488   A Machine Collection resource represents the collection of Machine entities within a Provider. This
1489   resource can be used to locate and create Machines.

1490   **Type URI:** http://www.dmtf.org/cimi/MachineCollection

Attribute	Type	Description	Optionality
machines	URI[]	An array of URIs referencing the set of Machines in the Provider.	optional

1491   The following describes the serialization of the entity in both JSON and XML:

1492   **JSON media type:** application/CIMI-MachineCollection+json

1493   **JSON serialization:**

```
1494        { "uri": string,
1495          "name": string, ?
1496          "description": string, ?
1497          "created": string,  ?
1498          "properties": [ "name": string, + ], ?
1499          "machines": [
1500            { "href": string }, +
1501          ], ?
1502          "operations": [
1503            { "rel": "add", "href": string }, ?
1504            { "rel": "edit", "href": string } ?
1505          ] ?
1506          ...
```

1507
```
          }
```

1508   **XML media type:** application/CIMI-MachineCollection+xml

1509   **XML serialization:**

1510
1511
1512
1513
1514
1515
1516
1517
1518
1519
1520
```
          <MachineCollection xmlns="http://www.dmtf.org/cimi">
            <uri> xs:anyURI </uri>
            <name> xs:string </name> ?
            <description> xs:string </description> ?
            <created> xs:string </created>
            <property name="xs:string"> xs:string </property> *
            <machine href="xs:anyURI"/> *
            <operation rel="add" href="xs:anyURI"/> ?
            <operation rel="edit" href="xs:anyURI"/> ?
            <xs:any>*
          </MachineCollection>
```

1521   **5.6.8.1 Operations**

1522   This entity supports the Read and Update operations.

1523   The following custom operations are also defined:

1524   • **Creating a New Machine**

1525   **/link@rel:** add

1526   This operation will create a new Machine.

1527   Input parameters: Either a reference to a Machine Template or a Machine Template itself.

1528   Output parameters: A reference to a new Machine and optionally the representation of the Machine.

1529   • **HTTP/REST Protocol**

1530   To create a new Machine a POST is sent to the "add" URI of the MachineCollection where the HTTP
1531   request body SHALL be as described below. Note this structure allows for certain properties to be
1532   passed in "by value" or by "reference". The definition of each property can be found in section 5.6.

1533   **JSON media type:** application/CIMI-MachineCreate+json

1534   **JSON serialization:**

1535
1536
1537
1538
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549
1550
1551
1552
1553
```
          { "name": string,
            "description": string, ?
            "properties": [ "name": string, + ], ?
            "machineTemplate": { "href": string, ?
              "properties": [ "name": string, + ], ?
              "machineConfig": { "href": string, ?
                "cpu": string, ?
                "memory": { "quantity": integer, "units": string }, ?
                "disks" : [
                   { "capacity": { "quantity": integer, "units": string },
                     "guestInterface": string }, +
                ] ?
              },
              "machineImage": { "href": string,
                "imageLocation": { "href": string }, ?
                "imageData": string, ?
              }, ?
              "machineAdmin": { "href": string, ?
                <provider specific data> ?
```

```
1554                    }, ?
1555                    "volumes": [
1556                      { "href": string, "attachmentPoint": string, "protocol": string}, +
1557                    ], ?
1558                    "volumeTemplates": [
1559                      { "href": string, ?
1560                        "attachmentPoint": string, "protocol": string,
1561                        "volumeConfig": { "href": string, ?
1562                          "properties": [ "name": string, + ], ?
1563                          "format": string, ?
1564                          "capacity": { "quantity": number, "units": string }, ?
1565                          "supportsSnapshots": boolean, ?
1566                          "guestInterface": string ?
1567                        }, ?
1568                        "volumeImage": { "href": string,
1569                          "properties": [ "name": string, + ], ?
1570                          "imageLocation": { "href": string }, ?
1571                          "imageData": string, ?
1572                          "bootable": boolean ?
1573                        } ?
1574                      }, +
1575                    ]
1576                  }
1577                  ...
1578                }
```

1579   **XML media type:** application/CIMI-MachineCreate+xml

1580   **XML serialization**

```
1581              <MachineCreate>
1582                <name> xs:string </name>
1583                <description> xs:string </description> ?
1584                <property name="xs:string"> xs:string </property> *
1585
1586                <machineTemplate href="xs:anyURI"? >
1587                  <property name="xs:string"> xs:string </property> *
1588                  <machineConfig href="xs:anyURI"? >
1589                    <property name="xs:string"> xs:string </property> *
1590                    <cpu> xs:string </cpu> ?
1591                    <memory quantity="xs:integer" units="xs:string"/>
1592                    <disk>
1593                      <capacity quantity="xs:integer" units="xs:string">
1594                      <guestInterface> xs:string </guestInterface>
1595                    </disk> *
1596                  </machineConfig>
1597                  <machineImage href="xs:anyURI">
1598                    <property name="xs:string"> xs:string </property> *
1599                    <imageLocation href="xs:anyURI"/> ?
1600                    <imageData> xs:string </imageData> ?
1601                  </machineImage>
1602                  <machineAdmin href="xs:anyURI"? >
1603                    xs:any*   <!-- provider specific data -->
1604                  </machineAdmin> ?
1605                  <volume href="xs:anyURI"
1606                          attachmentPoint="xs:string" protocol="xs:string" /> *
1607                  <volumeTemplate href="xs:anyURI"?
1608                                  attachmentPoint="xs:string" protocol="xs:string" >
1609                    <property name="xs:string"> xs:string </property> *
1610                    <volumeConfig href="xs:anyURI">
1611                      <property name="xs:string"> xs:string </property> *
1612                      <format> xs:string </format> ?
1613                      <capacity quantity="xs:integer" units="xs:string"/> ?
1614                      <supportsSnapshots> xs:boolean </supportsSnapshots> ?
```

```
1615              <guestInterface> xs:string </guestInterface> ?
1616          </volumeConfig>
1617          <volumeImage href="xs:anyURI">
1618              <property name="xs:string"> xs:string </property> *
1619              <imageLocation href="xs:anyURI"/> ?
1620              <imageData> xs:any* </imageData> ?
1621              <bootable> xs:boolean </bootable> ?
1622          </volumeImage> ?
1623        </volumeTemplate> *
1624      </machineTemplate>

1626      <xs:any>*

1628    </MachineCreate>
```

1629 The serialization of some reference properties are specified such that a request MAY either include a
1630 reference ("href") to an existing entity or to include the entity "inline" as a set of additional properties.
1631 Requests SHALL NOT include both a reference and the inlined set of properties.

1632 Upon successful processing of the request, the HTTP response body MAY either be empty or contain a
1633 serialization of the Machine entity.

1634 ### 5.6.9   Machine Admin

1635 A Machine Admin entity contains the information required to create the initial administrative super- user of
1636 a newly created Machine. Due to the variation between operating systems and Providers, this
1637 specification does not manadate one particular set of attributes that all implementations need to support.
1638 However, Providers are expected to extend this entity with additional attributes to meet their
1639 requirements.

1640 For example, a Provider might extend this entity with username and password attributes, which would
1641 then be the login information for new Machines. These extension attributes would appear as siblings to
1642 the common attributes like 'name' and 'description'.

1643 **Type URI:** http://www.dmtf.org/cimi/MachineAdmin

1644 Some common extensions that Providers might use include:

1645 **UserName/Password:**

Attribute	Type	Description	Optionality
userName	string	The initial superuser's user name.	mandatory
password	string	Initial superuser's password.	mandatory

1646 **Public Key:**

Attribute	Type	Description	Optionality
key	byte[]	The digit of the public key for the initial superuser.	mandatory

1647 **JSON media type:** application/CIMI-MachineAdmin+json

1648 **JSON serialization:**

```
1649    { "uri": string,
1650      "name": string, ?
1651      "description": string, ?
```

```
1652            "created": string,  ?
1653            "properties": [ "name": string, + ], ?
1654            "operations": [
1655              { "rel": "edit", "href": string } ?
1656              { "rel": "delete", "href": string } ?
1657            ] ?
1658            ...
1659          }
```

1660   **XML media type:** application/CIMI-MachineAdmin+xml

1661   **XML serialization:**

```
1662          <MachineAdmin xmlns="http://www.dmtf.org/cimi">
1663            <uri> xs:anyURI </uri>
1664            <name> xs:string </name> ?
1665            <description> xs:string </description> ?
1666            <created> xs:string </created>
1667            <property name="xs:string"> xs:string </property> *
1668            <operation rel="edit" href="xs:anyURI"/> ?
1669            <operation rel="delete" href="xs:anyURI"/> ?
1670            <xs:any>*
1671          </MachineAdmin>
```

1672   **5.6.9.1 Operations**

1673   This entity supports the Read, Update and Delete operations. Create is supported via the Machine Admin
1674   Collection entity.

1675   **5.6.10 Machine Admin Collection**

1676   A Machine Admin Collection entity represents the collection of Machine Admin entities within a Provider.
1677   This entity can be used to locate and create MachineAdmins.

1678   **Type URI:** http://www.dmtf.org/cimi/MachineAdminCollection

Attribute	Type	Description	Optionality
machineAdmins	URI[]	An array of URIs referencing the set of Machine Admins in the provider.	optional

1679   The following describes the serialization of the entity in both JSON and XML:

1680   **JSON media type:** application/CIMI-MachineAdminCollection+json

1681   **JSON serialization:**

```
1682          { "uri": string,
1683            "name": string, ?
1684            "description": string, ?
1685            "created": string,  ?
1686            "properties": [ "name": string, + ], ?
1687            "machineAdmins": [
1688              { "href": stirng }, +
1689            ], ?
1690            "operations": [
1691              { "rel": "add", "href": string }, ?
1692              { "rel": "edit", "href": string } ?
1693            ] ?
1694            ...
1695          }
```

1696   **XML media type:** application/CIMI-MachineAdminCollection+xml

1697   **XML serialization:**

```
1698        <MachineAdminCollection xmlns="http://www.dmtf.org/cimi">
1699          <uri> xs:anyURI </uri>
1700          <name> xs:string </name> ?
1701          <description> xs:string </description> ?
1702          <created> xs:string </created>
1703          <property name="xs:string"> xs:string </property> *
1704          <machineAdmin href="xs:anyURI"/> *
1705          <operation rel="add" href="xs:anyURI"/> ?
1706          <operation rel="edit" href="xs:anyURI"/> ?
1707          <xs:any>*
1708        </MachineAdminCollection>
```

1709   **5.6.10.1   Operations**

1710   This entity supports the Read and Update operations. Creation of new Machine Admin entities is
1711   supported via a POST to the "addLink" URI as described in section 4.2.1.1.

1712   ## 5.7   Volume Entities and Relationships

1713   The following diagram illustrates the entities involved in constructing a Volume and their relationships.
1714   Although this drawing is in the style of an Entity Relationship diagram, the use of UML is neither rigorous
1715   nor normative.
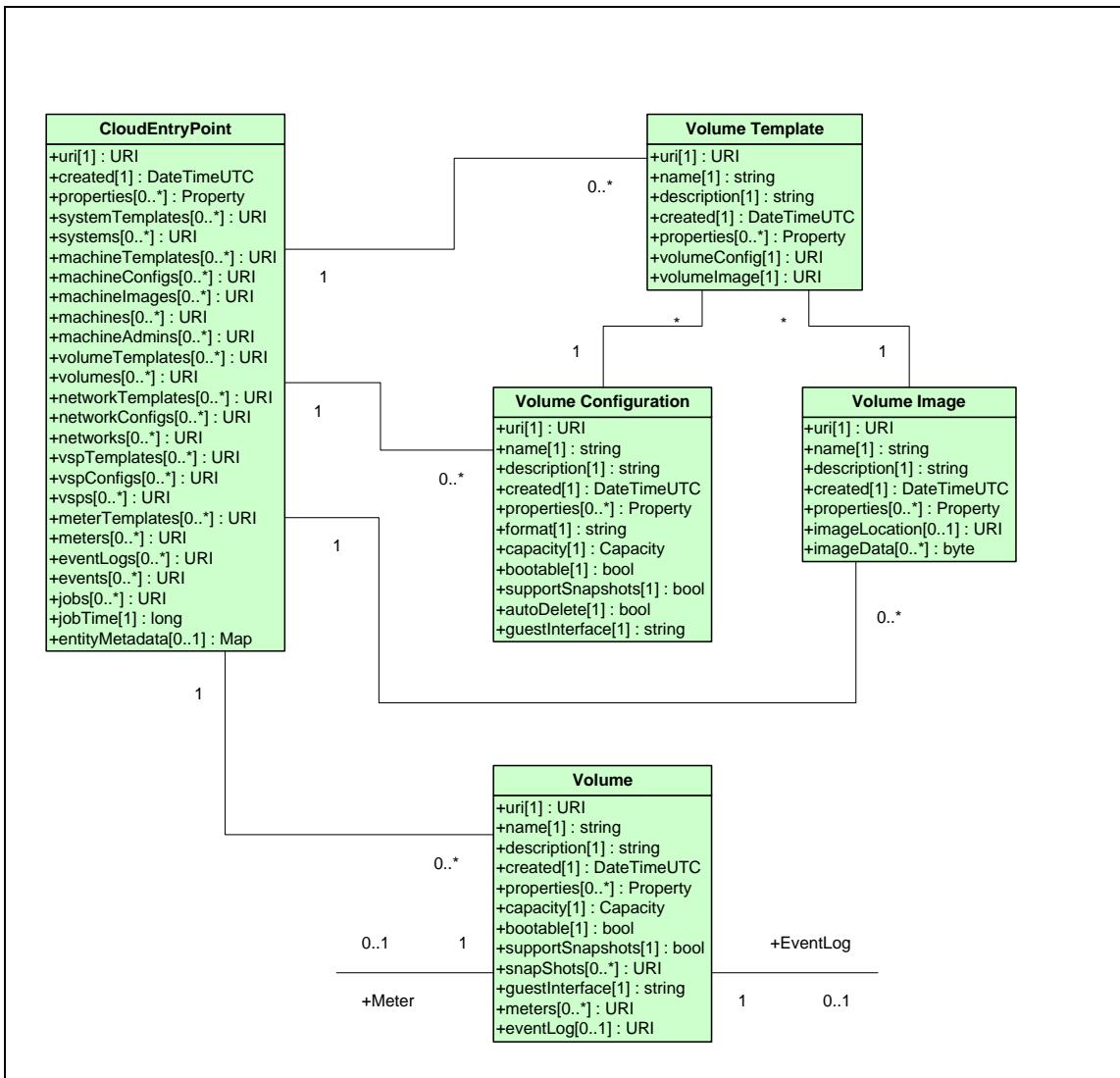
1716

1717   **Figure 3 - Volume Entities**

1718   **5.7.1   Volume Template**

1719   This entity captures the configuration values for realizing a Volume. A Volume Template may be used to
1720   create multiple Volumes.

1721   **Type URI:** http://www.dmtf.org/cimi/VolumeTemplate

Attribute	Type	Description	Optionality
volumeConfig	URI	A reference to the Volume Configuration that will be used to create a Volume from this Volume Template.	mandatory
volumeImage	URI	A reference to the Volume Image that will be used to create a Volume from this Volume Template.	mandatory

1722   The following describes the serialization of the entity in both JSON and XML:

1723 **JSON media type:** application/CIMI-VolumeTemplate+json

1724 **JSON serialization:**

```
1725        { "uri": string,
1726          "name": string, ?
1727          "description": string, ?
1728          "created": string,  ?
1729          "properties": [ "name": string, + ], ?
1730          "volumeConfig": { "href": string },
1731          "volumeImage": { "href": string },
1732          "operations": [
1733            { "rel": "edit", "href": string }, ?
1734            { "rel": "delete", "href": string } ?
1735          ] ?
1736          ...
1737        }
```

1738 **XML media type:** application/CIMI-VolumeTemplate+xml

1739 **XML serialization:**

```
1740        <VolumeTemplate xmlns="http://www.dmtf.org/cimi">
1741          <uri> xs:anyURI </uri>
1742          <name> xs:string </name> ?
1743          <description> xs:string </description> ?
1744          <created> xs:string </created>
1745          <property name="xs:string"> xs:string </property> *
1746          <volumeConfig href="xs:anyURI"/>
1747          <volumeImage href="xs:anyURI"/>
1748          <operation rel="edit" href="xs:anyURI"/> ?
1749          <operation rel="delete" href="xs:anyURI"/> ?
1750          <xs:any>*
1751        </VolumeTemplate>
```

1752 **5.7.1.1 Operations**

1753 This entity supports the Read, Update and Delete operations. Create is supported via the Volume
1754 Template Collection entity.

1755 **5.7.2 Volume Template Collection**

1756 A Volume Template Collection entity represents the collection of VolumeTemplate entities within a
1757 Provider. This entity can be used to locate and create Volume Templates.

1758 **Type URI:** http://www.dmtf.org/cimi/VolumeTemplateCollection

Attribute	Type	Description	Optionality
volumeTemplates	URI[]	An array of URIs referencing the set of Volume Templates in the Provider.	optional

1759 The following describes the serialization of the entity in both JSON and XML:

1760 **JSON media type:** application/CIMI-VolumeTemplateCollection+json

1761 **JSON serialization:**

```
1762        { "uri": string,
1763          "name": string, ?
1764          "description": string, ?
1765          "created": string,  ?
```

```
1766            "properties": [ "name": string, + ], ?
1767            "volumeTemplates": [
1768              { "href": string }, +
1769            ], ?
1770            "operations": [
1771              { "rel": "add", "href": string }, ?
1772              { "rel": "edit", "href": string } ?
1773            ] ?
1774            ...
1775          }
```

1776 **XML media type:** application/CIMI-VolumeTemplateCollection+xml

1777 **XML serialization:**

```
1778          <VolumeTemplateCollection xmlns="http://www.dmtf.org/cimi">
1779            <uri> xs:anyURI </uri>
1780            <name> xs:string </name> ?
1781            <description> xs:string </description> ?
1782            <created> xs:string </created>
1783            <property name="xs:string"> xs:string </property> *
1784            <volumeTemplate href="xs:anyURI"/> *
1785            <operation rel="add" href="xs:anyURI"/> ?
1786            <operation rel="edit" href="xs:anyURI"/> ?
1787            <xs:any>*
1788          </VolumeTemplateCollection>
```

1789 **5.7.2.1 Operations**

1790 This entity supports the Read and Update operations. Creation of new Volume Template entities is
1791 supported via a POST to the "addLink" URI as described in section 4.2.1.1.

1792 **5.7.3   Volume Configuration**

1793 The Volume Configuration entity represents the set of configuration values needed to create a Volume
1794 with certain characteristics. Volume Configurations are created by Providers and MAY, at the Providers
1795 discretion, be created by Consumers.

1796 **Type URI:** http://www.dmtf.org/cimi/VolumeConfiguration

Attribute	Type	Description	Optionality
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".	mandatory
capacity	The default size, when limited, of the Volume created from this Volume Configuration. This attribute has the following, sub-attributes. | Attribute	Type	Description
---	---	---
quantity	integer	A numerical quantity expressed as an integer.		mandatory

	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**.	
supportsSnapshots	boolean		This property indicates whether Volumes created from this Volume Configuration will support the ability to take snapshots.	mandatory
guestInterface	String		This property indicates the interface that will be offered to a Machine instances by Volumes created from this Volume Configuration.	mandatory

1797   The following describes the serialization of the entity in both JSON and XML:

1798   **JSON media type:** application/CIMI-VolumeConfiguration+json

1799   **JSON serialization:**

```
1800          { "uri": string,
1801            "name": string, ?
1802            "description": string, ?
1803            "created": string,   ?
1804            "properties": [ "name": string, + ], ?
1805            "format": string,
1806            "capacity": { "quantity": number, "units": string },
1807            "supportsSnapshots": boolean,
1808            "guestInterface": string,
1809            "operations": [
1810              { "rel": "edit", "href": string }, ?
1811              { "rel": "delete", "href": string } ?
1812            ] ?
1813            ...
1814          }
```

1815   **XML media type:** application/CIMI-VolumeConfiguration+xml

1816   **XML serialization:**

```
1817          <VolumeConfiguration xmlns="http://www.dmtf.org/cimi">
1818            <uri> xs:anyURI </uri>
1819            <name> xs:string </name> ?
1820            <description> xs:string </description> ?
1821            <created> xs:string </created>
1822            <property name="xs:string"> xs:string </property> *
1823            <format> xs:string </format>
1824            <capacity quantity="xs:integer" units="xs:string"/>
1825            <supportsSnapshots> xs:boolean </supportsSnapshots>
1826            <guestInterface> xs:string </guestInterface>
1827            <operation rel="edit" href="xs:anyURI"/> ?
1828            <operation rel="delete" href="xs:anyURI"/> ?
1829            <xs:any>*
1830          </VolumeConfiguration>
```

1831   **5.7.3.1 Operations**

1832   This entity supports the Read, Update and Delete operations. Create is supported via the Volume
1833   Configuration Collection entity.

1834   **5.7.4   Volume Configuration Collection**

1835   A Volume Configuration Collection entity represents the collection of Volume Configuration entities within
1836   a Provider. This entity can be used to locate and create Volume Configurations.

1837   **Type URI:** http://www.dmtf.org/cimi/VolumeConfigurationCollection

Attribute	Type	Description	Optionality
volumeConfigurations	URI[]	An array of URIs referencing the set of Volume Configurations in the Provider.	optional

1838   The following describes the serialization of the entity in both JSON and XML:

1839   **JSON media type:** application/CIMI-VolumeConfigurationCollection+json

1840   **JSON serialization:**

```
1841        { "uri": string,
1842          "name": string, ?
1843          "description": string, ?
1844          "created": string,  ?
1845          "properties": [ "name": string, + ], ?
1846          "volumeConfigurations": [
1847            { "href": string }, +
1848          ], ?
1849          "operations": [
1850            { "rel": "add", "href": string }, ?
1851            { "rel": "edit", "href": string } ?
1852          ] ?
1853          ...
1854        }
```

1855   **XML media type:** application/CIMI-VolumeConfigurationCollection+xml

1856   **XML serialization:**

```
1857        <VolumeConfigurationCollection xmlns="http://www.dmtf.org/cimi">
1858          <uri> xs:anyURI </uri>
1859          <name> xs:string </name> ?
1860          <description> xs:string </description> ?
1861          <created> xs:string </created>
1862          <property name="xs:string"> xs:string </property> *
1863          <volumeConfiguration href="xs:anyURI"/> *
1864          <operation rel="add" href="xs:anyURI"/> ?
1865          <operation rel="edit" href="xs:anyURI"/> ?
1866          <xs:any>*
1867        </VolumeConfigurationCollection>
```

1868   **5.7.4.1 Operations**

1869   This entity supports the Read and Update operations. Creation of new Volume Image entities is
1870   supported via a POST to the "addLink" URI as described in section 4.2.1.1.

1871   **5.7.5   Volume Image**

1872   This entity represents an image that could be place on a pre-loaded volume.

1873   **Type URI:** http://www.dmtf.org/cimi/VolumeImage

Attribute	Type	Description	Optionality
imageLocation	URI	A URI that references the location of the binary data that makes up this image. Either this attribute must be present or the imageData attribute must be present.* This value is **imutable.**	optional
imageData	byte[]	The binary data that makes up this image. Either this attribute must be present or the imageLocation attribute must be present.* This value is **imutable**.	optional
bootable	boolean	This property indicates whether Volumes created from this Volume Configuration will be bootable.	mandatory

1874   * Either imageLocation or imageData SHALL be present, however both values shall not be present
1875   simultaneously.

1876   The following describes the serialization of the entity in both JSON and XML:

1877   **JSON media type:** application/CIMI-VolumeImage+json

1878   **JSON serialization:**

```
1879        { "uri": string,
1880          "name": string, ?
1881          "description": string, ?
1882          "created": string,  ?
1883          "properties": [ "name": string, + ], ?
1884          "imageLocation": { "href": string }, ?
1885          "imageData": string, ?
1886          "bootable": boolean,
1887          "operations": [
1888            { "rel": "edit", "href": string }, ?
1889            { "rel": "delete", "href": string } ?
1890          ] ?
1891          ...
1892        }
```

1893   **XML media type:** application/CIMI-VolumeImage+xml

1894   **XML serialization:**

```
1895        <VolumeImage xmlns="http://www.dmtf.org/cimi">
1896          <uri> xs:anyURI </uri>
1897          <name> xs:string </name> ?
1898          <description> xs:string </description> ?
1899          <created> xs:string </created>
1900          <property name="xs:string"> xs:string </property> *
1901          <imageLocation href="xs:anyURI"/> ?
1902          <imageData> xs:any* </imageData> ?
1903          <bootable> xs:boolean </bootable>
1904          <operation rel="edit" href="xs:anyURI"/> ?
1905          <operation rel="delete" href="xs:anyURI"/> ?
1906          <xs:any>*
1907        </VolumeImage>
```

5.7.5.1 Operations

1908

1909 This entity supports the Read, Update and Delete operations. Create is supported via the Volume Image
1910 Collection entity.

5.7.6   Volume Image Collection

1911

1912 A Volume Image Collection entity represents the collection of VolumeImage entities within a Provider.
1913 This entity can be used to locate and create Volume Images.

1914 **Type URI:** http://www.dmtf.org/cimi/VolumeImageCollection

Attribute	Type	Description	Optionality
volumeImages	URI[]	An array of URIs referencing the set of Volume Images in the Provider.	optional

1915 The following describes the serialization of the entity in both JSON and XML:

1916 **JSON media type:** application/CIMI-VolumeImageCollection+json

1917 **JSON serialization:**

```
1918        { "uri": string,
1919          "name": string, ?
1920          "description": string, ?
1921          "created": string,  ?
1922          "properties": [ "name": string, + ], ?
1923          "volumeImages": [
1924            { "href": string }, +
1925          ], ?
1926          "operations": [
1927            { "rel": "add", "href": string }, ?
1928            { "rel": "edit", "href": string } ?
1929          ] ?
1930          ...
1931        }
```

1932 **XML media type:** application/CIMI-VolumeImageCollection+xml

1933 **XML serialization:**

```
1934        <VolumeImageCollection xmlns="http://www.dmtf.org/cimi">
1935          <uri> xs:anyURI </uri>
1936          <name> xs:string </name> ?
1937          <description> xs:string </description> ?
1938          <created> xs:string </created>
1939          <property name="xs:string"> xs:string </property> *
1940          <volumeImage href="xs:anyURI"/> *
1941          <operation rel="add" href="xs:anyURI"/> ?
1942          <operation rel="edit" href="xs:anyURI"/> ?
1943          <xs:any>*
1944        </VolumeImageCollection>
```

5.7.6.1 Operations

1945

1946 This entity supports the Read and Update operations. Creation of new Volume Image entities is
1947 supported via a POST to the "addLink" URI as described in section 4.2.1.1.

1948 **5.7.7   Volume**

1949 A Volume represents storage at either the block or file-system level. Volumes can be attached to
1950 Machines. Once attached, Volumes can be accessed by processes on that Machine.

1951 **Type URI:** http://www.dmtf.org/cimi/Volume

Attribute	Type	Description	Optionality
capacity		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. <table><tr><td>Attribute</td><td>Type</td><td>Description</td></tr><tr><td>quantity</td><td>integer</td><td>A numerical quantity expressed as an integer.</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**.</td></tr></table>	mandatory
bootable	boolean	This property indicates whether this Volume is bootable.	mandatory
supportsSnapshots	boolean	This property indicates whether the Volume supports the ability to take snapshots.	mandatory
snapShots	URI[]	A list of references to Volume Images that represent snapshots taken from the Volume.	optional
guestInterface	String	This property indicates the interface offered to a Machine instance to gain access to the storage contents.	mandatory
meters	URI[]	A list of references to Meters monitored for this Volume.	optional
eventLog	URI	A reference to the EventLog of this Volume.	optional

1952 The following describes the serialization of the entity in both JSON and XML:

1953 **JSON media type:** application/CIMI-Volume+json

1954 **JSON serialization:**

```
1955    { "uri": string,
1956      "name": string, ?
1957      "description": string, ?
1958      "created": string,  ?
1959      "properties": [ "name": string, + ], ?
1960      "capacity": { "quantity": number, "units": string },
1961      "bootable": boolean,
1962      "supportsSnapshots": boolean,
1963      "snapshots": [
1964        { "href": string }, +
1965      ], ?
1966      "guestInterface": string,
```

```
1967          "meters": [
1968            { "href": string }, +
1969          ], ?
1970          "eventLog": { "href": string }, ?
1971          "operations": [
1972            { "rel": "edit", "href": string }, ?
1973            { "rel": "delete", "href": string } ?
1974          ] ?
1975          ...
1976        }
```

1977 **XML media type:** application/CIMI-Volume+xml

1978 **XML serialization:**

```
1979        <Volume xmlns="http://www.dmtf.org/cimi">
1980          <uri> xs:anyURI </uri>
1981          <name> xs:string </name> ?
1982          <description> xs:string </description> ?
1983          <created> xs:string </created>
1984          <property name="xs:string"> xs:string </property> *
1985          <capacity quantity="xs:integer" units="xs:string"/>
1986          <bootable> xs:boolean </bootable>
1987          <supportsSnapshots> xs:boolean </supportsSnapshots>
1988          <shapshot href="xs:anyURI"/> *
1989          <guestInterface> xs:string </guestInterface>
1990          <meter href="xs:anyURI"/> *
1991          <eventLog href="xs:anyURI"/> ?
1992          <operation rel="edit" href="xs:anyURI"/> ?
1993          <operation rel="delete" href="xs:anyURI"/> ?
1994          <xs:any>*
1995        </Volume>
```

1996 **5.7.7.1 Operations**

1997 This entity supports the Read, Update and Delete operations. Create is supported via the Volume
1998 Collection entity.

1999 **5.7.8 Volume Collection**

2000 A Volume Collection entity represents the collection of Volumes within a Provider. This entity can be used
2001 to locate and create Volumes.

2002 **Type URI:** http://www.dmtf.org/cimi/VolumeCollection

Attribute	Type	Description	Optionality
volumes	URI[]	An array of URIs referencing the set of Volumes in the provider.	optional

2003 The following describes the serialization of the entity in both JSON and XML:

2004 **JSON media type:** application/CIMI-VolumeCollection+json

2005 **JSON serialization:**

```
2006        { "uri": string,
2007          "name": string, ?
2008          "description": string, ?
2009          "created": string,  ?
2010          "properties": [ "name": string, + ], ?
2011          "volumes": [
2012            { "href": string }, +
```

```
2013            ], ?
2014            "operations": [
2015              { "rel": "add", "href": string }, ?
2016              { "rel": "edit", "href": string } ?
2017            ] ?
2018            ...
2019          }
```

2020   **XML media type:** application/CIMI-VolumeCollection+xml

2021   **XML serialization:**

```
2022            <VolumeCollection xmlns="http://www.dmtf.org/cimi">
2023              <uri> xs:anyURI </uri>
2024              <name> xs:string </name> ?
2025              <description> xs:string </description> ?
2026              <created> xs:string </created>
2027              <property name="xs:string"> xs:string </property> *
2028              <volume href="xs:anyURI"/> *
2029              <operation rel="add" href="xs:anyURI"/> ?
2030              <operation rel="edit" href="xs:anyURI"/> ?
2031              <xs:any>*
2032            </VolumeCollection>
```

2033   **5.7.8.1 Operations**

2034   This entity supports the Read and Update operations.

2035   The following custom operations are also defined:

2036   •   **Creating a New Volume**

2037   **/link@rel:** add

2038   This operation will create a new Volume.

2039   Input parameters: Either a reference to a Volume Template or a Volume Template itself.

2040   Output parameters: A reference to a new Volume and optionally the representation of the Volume.

2041   •   **HTTP/REST Protocol**

2042   To create a new Volume a POST is sent to the "add" URI of the VolumeCollection where the HTTP
2043   request body SHALL be as described below. Note this structure allows for certain properties to be
2044   passed in "by value" or by "reference". The definition of each property can be found in section 5.6.9.

2045   **JSON media type:** application/CIMI-VolumeCreate+json

2046   **JSON serialization:**

```
2047            { "name": string,
2048              "description": string, ?
2049              "properties": [ "name": string, + ], ?
2050              "volumeTemplate": { "href": string, ?
2051                "properties": [ "name": string, + ], ?
2052                "volumeConfig": { "href": string, ?
2053                  "properties": [ "name": string, + ], ?
2054                  "format": string, ?
2055                  "capacity": { "quantity": number, "units": string }, ?
2056                  "supportsSnapshots": boolean, ?
2057                  "guestInterface": string, ?
2058                },
2059                "volumeImage": { "href": string,
```

```
2060              "imageLocation": { "href": string }, ?
2061              "imageData": string, ?
2062              "bootable": boolean ?
2063            } ?
2064          }
2065          ...
2066        }
```

2067   **XML media type:** application/CIMI-VolumeCreate+xml

2068   **XML serialization**

```
2069        <VolumeCreate>
2070          <name> xs:string </name>
2071          <description> xs:string </description> ?
2072          <property name="xs:string"> xs:string </property> *
2073          <volumeTemplate href="xs:anyURI"? >
2074            <property name="xs:string"> xs:string </property> *
2075            <volumeConfig href="xs:anyURI"? >
2076              <property name="xs:string"> xs:string </property> *
2077               <format> xs:string </format> ?
2078              <capacity quantity="xs:integer" units="xs:string"/> ?
2079              <supportsSnapshots> xs:boolean </supportsSnapshots> ?
2080              <guestInterface> xs:string </guestInterface> ?
2081            </volumeConfig>
2082            <volumeImage href="xs:anyURI">
2083              <property name="xs:string"> xs:string </property> *
2084              <imageLocation href="xs:anyURI"/> ?
2085              <imageData> xs:any* </imageData> ?
2086              <bootable> xs:boolean </bootable> ?
2087            </volumeImage> ?
2088          </volumeTemplate>
2089          <xs:any>*
2090        </VolumeCreate>
```

2091   The serialization of some reference properties are specified such that a request MAY either include a
2092   reference ("href") to an existing entity or to include the entity "inline" as a set of additional properties.
2093   Requests SHALL NOT include both a reference and the inlined set of properties.

2094   Upon successful processing of the request, the HTTP response body MAY either be empty or contain a
2095   serialization of the Volume entity.

5.8   Network Entities and Relationships

2097   The following diagram illustrates the entities involved in constructing Networks and their Virtual Switch
2098   Ports (VSPs) and their relationships. Although this drawing is in the style of an Entity Relationship
2099   diagram, the use of UML is neither rigorous nor normative.
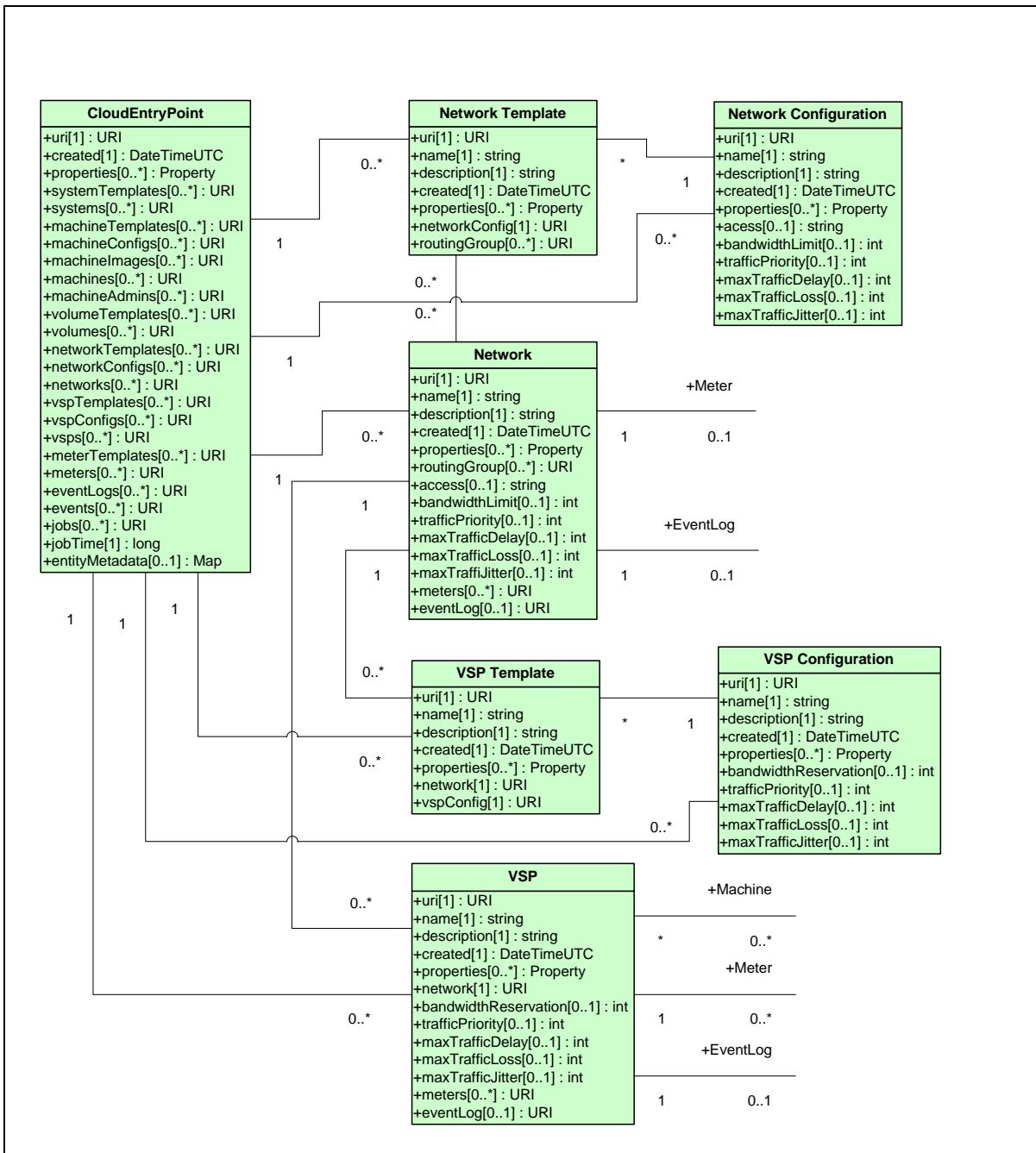
2100

Figure 4 - Network Entities

5.8.1   Network Template

2103   The Network Template is a set of configuration values for realizing a Network. An instance of Network
2104   Template may be used to create multiple Networks.

2105   **Type URI:** http://www.dmtf.org/cimi/NetworkTemplate

Attribute	Type	Description	Optionality

networkConfig	URI	A reference to the Network Configuration that will be used to create a Network from this Network Template.	mandatory
routingGroup	URI[]	Array of references to Networks that will route to each other.	mandatory

2106 The following describes the serialization of the entity in both JSON and XML:

2107 **JSON media type:** application/CIMI-NetworkTemplate+json

2108 **JSON serialization:**

```
2109          { "uri": string,
2110            "name": string, ?
2111            "description": string, ?
2112            "created": string,  ?
2113            "properties": [ "name": string, + ], ?
2114            "networkConfig": { "href": string },
2115            "routingGroup": [
2116              { "href": string }, +
2117            ], ?
2118            "operations": [
2119              { "rel": "edit", "href": string }, ?
2120              { "rel": "delete", "href": string } ?
2121            ] ?
2122            ...
2123          }
```

2124 **XML media type:** application/CIMI-NetworkTemplate+xml

2125 **XML serialization:**

```
2126          <NetworkTemplate xmlns="http://www.dmtf.org/cimi">
2127            <uri> xs:anyURI </uri>
2128            <name> xs:string </name> ?
2129            <description> xs:string </description> ?
2130            <created> xs:string </created>
2131            <property name="xs:string"> xs:string </property> *
2132            <networkConfig href="xs:anyURI"/>
2133            <routingGroup href="xs:anyURI"/> *
2134            <operation rel="edit" href="xs:anyURI"/> ?
2135            <operation rel="delete" href="xs:anyURI"/> ?
2136            <xs:any>*
2137          </NetworkTemplate>
```

2138 5.8.1.1 Operations

2139 This entity supports the Read, Update and Delete operations. Create is supported via the Network
2140 Template Collection entity.

2141 5.8.2 Network Template Collection

2142 A Network Template Collection entity represents the collection of NetworkTemplateas within a Provider.
2143 This resource can be used to locate and create NetworkTemplates.

2144 **Type URI:** http://www.dmtf.org/cimi/NetworkTemplateCollection

Attribute	Type	Description	Optionality
networkTemplates	URI[]	An array of URIs referencing the set of Network Templates in the Provider.	optional

2145 The following describes the serialization of the entity in both JSON and XML:

2146 **JSON media type:** application/CIMI-NetworkTemplateCollection+json

2147 **JSON serialization:**

```
2148   { "uri": string,
2149     "name": string, ?
2150     "description": string, ?
2151     "created": string,  ?
2152     "properties": [ "name": string, + ], ?
2153     "networkTemplates": [
2154       { "href": string }, +
2155     ], ?
2156     "operations": [
2157       { "rel": "add", "href": string }, ?
2158       { "rel": "edit", "href": string } ?
2159     ] ?
2160     ...
2161   }
```

2162 **XML media type:** application/CIMI-NetworkTemplateCollection+xml

2163 **XML serialization:**

```
2164   <NetworkTemplateCollection xmlns="http://www.dmtf.org/cimi">
2165     <uri> xs:anyURI </uri>
2166     <name> xs:string </name> ?
2167     <description> xs:string </description> ?
2168     <created> xs:string </created>
2169     <property name="xs:string"> xs:string </property> *
2170     <networkTemplate href="xs:anyURI"/> *
2171     <operation rel="add" href="xs:anyURI"/> ?
2172     <operation rel="edit" href="xs:anyURI"/> ?
2173     <xs:any>*
2174   </NetworkTemplateCollection>
```

2175 5.8.2.1 Operations

2176 This entity supports the Read and Update operations. Creation of new Network Template entities is
2177 supported via a POST to the "addLink" URI as described in section 4.2.1.1.

2178 5.8.3 Network Configuration

2179 The set of configuration values representing the information needed to create a Network with certain
2180 characteristics.

2181 **Type URI:** http://www.dmtf.org/cimi/NetworkConfiguration

Attribute	Type	Description	Optionality
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.	optional
bandwidthLimit	integer	Maximum allowable bandwidth.	mandatory
trafficPriority	integer	Indicates priority of traffic on this network.	mandatory
maxTrafficDelay	integer	The requested maximum delay for end to end transmission	mandatory

		specified in nanoseconds with uint64 (i.e. latency).	
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.	optional
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.	optional

2182   The following describes the serialization of the entity in both JSON and XML:

2183   **JSON media type:** application/CIMI-NetworkConfiguration+json

2184   **JSON serialization:**

```
2185        { "uri": string,
2186          "name": string, ?
2187          "description": string, ?
2188          "created": string,  ?
2189          "properties": [ "name": string, + ], ?
2190          "access": string, ?
2191          "bandwidthLimit": number,
2192          "trafficPriority": number,
2193          "maxTrafficDelay": number,
2194          "maxTrafficLoss": number, ?
2195          "maxTrafficJitter": number, ?
2196          "operations": [
2197            { "rel": "edit", "href": string }, ?
2198            { "rel": "delete", "href": string } ?
2199          ] ?
2200          ...
2201        }
```

2202   **XML media type:** application/CIMI-NetworkConfiguration+xml

2203   **XML serialization:**

```
2204        <NetworkConfiguration xmlns="http://www.dmtf.org/cimi">
2205          <uri> xs:anyURI </uri>
2206          <name> xs:string </name> ?
2207          <description> xs:string </description> ?
2208          <created> xs:string </created>
2209          <property name="xs:string"> xs:string </property> *
2210          <access> xs:string </access> ?
2211          <bandwidthLimit> xs:string <bandwidthLimit> 
2212          <trafficPriority> xs:integer </trafficPriority>
2213          <maxTrafficDelay> xs:integer </maxTrafficDelay>
2214          <maxTrafficLoss> xs:integer </maxTrafficLoss> ?
2215          <maxTrafficJitter> xs:integer </maxTrafficJitter> ?
2216          <operation rel="edit" href="xs:anyURI"/> ?
2217          <operation rel="delete" href="xs:anyURI"/> ?
2218          <xs:any>*
2219        </NetworkConfiguration>
```

2220   **5.8.3.1 Operations**

2221   This entity supports the Read, Update and Delete operations. Create is supported via the Network
2222   Configuration Collection entity.

2223 **5.8.4 Network Configuration Collection**

2224 A Network Configuration Collection entity represents the collection of Network Configurations within a
2225 Provider. This entity can be used to locate and create Network Configurations.

2226 **Type URI:** http://www.dmtf.org/cimi/NetworkConfigurationCollection

Attribute	Type	Description	Optionality
networkConfigurations	URI[]	An array of URIs referencing the set of Network Configurations in the provider.	optional

2227 The following describes the serialization of the entity in both JSON and XML:

2228 **JSON media type:** application/CIMI-NetworkConfigurationCollection+json

2229 **JSON serialization:**

```
2230        { "uri": string,
2231          "name": string, ?
2232          "description": string, ?
2233          "created": string,  ?
2234          "properties": [ "name": string, + ], ?
2235          "networkConfigurations": [
2236            { "href": string }, +
2237          ], ?
2238          "operations": [
2239            { "rel": "add", "href": string }, ?
2240            { "rel": "edit", "href": string } ?
2241          ] ?
2242          ...
2243        }
```

2244 **XML media type:** application/CIMI-NetworkConfigurationCollection+xml

2245 **XML serialization:**

```
2246        <NetworkConfigurationCollection xmlns="http://www.dmtf.org/cimi">
2247          <uri> xs:anyURI </uri>
2248          <name> xs:string </name> ?
2249          <description> xs:string </description> ?
2250          <created> xs:string </created>
2251          <property name="xs:string"> xs:string </property> *
2252          <networkConfiguration href="xs:anyURI"/> *
2253          <operation rel="add" href="xs:anyURI"/> ?
2254          <operation rel="edit" href="xs:anyURI"/> ?
2255          <xs:any>*
2256        </NetworkCollectionCollection>
```

2257 **5.8.4.1 Operations**

2258 This entity supports the Read and Update operations. Creation of new Network Collection entities is
2259 supported via a POST to the "addLink" URI as described in section 4.2.1.1.

2260 **5.8.5   Network**

2261 A Network is a realized entity that represents an abstraction of a layer 2 broadcast domain.

2262 **Type URI:** http://www.dmtf.org/cimi/Network

Attribute	Type	Description	Optionality
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.	optional
bandwidthLimit	integer	Maximum allowable bandwidth.	mandatory
trafficPriority	integer	Indicates priority of traffic on this network.	mandatory
maxTrafficDelay	integer	The requested maximum delay for end to end transmission specified in nanoseconds with uint64 (i.e. latency).	mandatory
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.	optional
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.	mandatory
routingGroup	URI[]	Array of references to Networks that will route to each other.	mandatory
meters	URI[]	A list of references to Meters monitored for this Network.	optional
eventLog	URI	A reference to the EventLog of this Network.	optional

2263   The following describes the serialization of the entity in both JSON and XML:

2264   **JSON media type:** application/CIMI-Network+json

2265   **JSON serialization:**

```
2266            { "uri": string,
2267              "name": string, ?
2268              "description": string, ?
2269              "created": string,  ?
2270              "properties": [ "name": string, + ], ?
2271              "access": string, ?
2272              "bandwidthLimit": number,
2273              "trafficPriority": number,
2274              "maxTrafficDelay": number,
2275              "maxTfafficLoss": number, ?
2276              "maxTrafficJitter": number,
2277              "routingGroup": [
2278                { "href": string }, +
2279              ], ?
2280              "meters": [
2281                { "href": string }, +
2282              ], ?
2283              "eventLog": { "href": string }, ?
2284              "operations": [
2285                { "rel": "edit", "href": string }, ?
2286                { "rel": "delete", "href": string } ?
2287              ] ?
2288              ...
```

2289   }

2290 **XML media type:** application/CIMI-Network+xml

2291 **XML serialization:**

```
2292        <Network xmlns="http://www.dmtf.org/cimi">
2293          <uri> xs:anyURI </uri>
2294          <name> xs:string </name> ?
2295          <description> xs:string </description> ?
2296          <created> xs:string </created>
2297          <property name="xs:string"> xs:string </property> *
2298          <access> xs:string </access> ?
2299          <bandwidthLimit> xs:integer </bandwidthLimit>
2300          <trafficPriority> xs:integer </trafficPriority>
2301          <maxTrafficDelay> xs:integer </maxTrafficDelay>
2302          <maxTrafficLoss> xs:integer </maxTrafficLoss> ?
2303          <maxTrafficJitter> xs:integer </maxTrafficJiffer>
2304          <routingGroup href="xs:anyURI"/> *
2305          <meter href="xs:anyURI"/> *
2306          <eventLog" href="xs:anyURI"/> ?
2307          <operation rel="edit" href="xs:anyURI"/> ?
2308          <operation rel="delete" href="xs:anyURI"/> ?
2309          <xs:any>*
2310        </Network>
```

2311 **5.8.5.1 Operations**

2312 This entity supports the Read, Update and Delete operations. Create is supported via the Network
2313 Collection entity.

2314 **5.8.6   Network Collection**

2315 A Network Collection entity represents the collection of Networks within a Provider. This entity can be
2316 used to locate and create Networks.

2317 **Type URI:** http://www.dmtf.org/cimi/NetworkCollection

Attribute	Type	Description	Optionality
networks	URI[]	An array of URIs referencing the set of Networks in the Provider.	optional

2318 The following describes the serialization of the entity in both JSON and XML:

2319 **JSON media type:** application/CIMI-NetworkCollection+json

2320 **JSON serialization:**

```
2321        { "uri": string,
2322          "name": string, ?
2323          "description": string, ?
2324          "created": string,  ?
2325          "properties": [ "name": string, + ], ?
2326          "networks": [
2327            { "href": string }, +
2328          ], ?
2329          "operations": [
2330            { "rel": "add", "href": string }, ?
2331            { "rel": "edit", "href": string } ?
2332          ] ?
2333          ...
2334        }
```

2335   **XML media type:** application/CIMI-NetworkCollection+xml

2336   **XML serialization:**

```
2337          <NetworkCollection xmlns="http://www.dmtf.org/cimi">
2338            <uri> xs:anyURI </uri>
2339            <name> xs:string </name> ?
2340            <description> xs:string </description> ?
2341            <created> xs:string </created>
2342            <property name="xs:string"> xs:string </property> *
2343            <network href="xs:anyURI"/> *
2344            <operation rel="add" href="xs:anyURI"/> ?
2345            <operation rel="edit" href="xs:anyURI"/> ?
2346            <xs:any>*
2347          </NetworkCollection>
```

2348   **5.8.6.1 Operations**

2349   This entity supports the Read and Update operations.

2350   The following custom operations are also defined:

2351   • **Creating a New Network**

2352   **/link@rel:** add

2353   This operation will create a new Network.

2354   Input parameters: Either a reference to a Network Template or a Network Template itself.

2355   Output parameters: A reference to a new Network and optionally the representation of the Network.

2356   • **HTTP/REST Protocol**

2357   To create a new Network a POST is sent to the "add" URI of the NetworkCollection where the HTTP
2358   request body SHALL be as described below. Note this structure allows for certain properties to be passed
2359   in "by value" or by "reference". The definition of each property can be found in section 5.8.

2360   **JSON media type:** application/CIMI-NetworkCreate+json

2361   **JSON serialization:**

```
2362          { "name": string,
2363            "description": string, ?
2364            "properties": [ "name": string, + ], ?
2365            "networkTemplate": { "href": string, ?
2366              "properties": [ "name": string, + ] ?
2367              "networkConfig": { "href": string, ?
2368                "properties": [ "name": string, + ], ?
2369                "access": string, ?
2370                "bandwidthLimit": number, ?
2371                "trafficPriority": number, ?
2372                "maxTrafficDelay": number, ?
2373                "maxTrafficLoss": number, ?
2374                "maxTrafficJitter": number, ?
2375              },
2376              "routingGroup": [
2377                { "href": string }, +
2378              ], ?
2379            }
2380            ...
2381          }
```

2382 **XML media type:** application/CIMI-NetworkCreate+xml

2383 **XML serialization**

```
2384    <NetworkCreate>
2385      <name> xs:string </name>
2386      <description> xs:string </description> ?
2387      <property name="xs:string"> xs:string </property> *
2388      <networkTemplate href="xs:anyURI"? >
2389        <property name="xs:string"> xs:string </property> *
2390        <networkConfig href="xs:anyURI"? >
2391          <access> xs:string </access> ?
2392          <bandwidthLimit> xs:string <bandwidthLimit> ?
2393          <trafficPriority> xs:integer </trafficPriority> ?
2394          <maxTrafficDelay> xs:integer </maxTrafficDelay> ?
2395          <maxTrafficLoss> xs:integer </maxTrafficLoss> ?
2396          <maxTrafficJitter> xs:integer </maxTrafficJitter> ?
2397        </networkConfig>
2398        <routingGroup href="xs:anyURI"/> *
2399      </networkTemplate>
2400      <xs:any>*
2401    </NetworkCreate>
```

2402 The serialization of some reference properties are specified such that a request MAY either include a
2403 reference ("href") to an existing entity or to include the entity "inline" as a set of additional properties.
2404 Requests SHALL NOT include both a reference and the inlined set of properties.

2405 Upon successful processing of the request, the HTTP response body MAY either be empty or contain a
2406 serialization of the Network entity.

2407 ### 5.8.7 VSP (Virtual Switch Port) Template

2408 The VSP Template is a set of Configuration values for realizing a VSP. A VSP Template may be used to
2409 create multiple VSPs.

2410 **Type URI:** http://www.dmtf.org/cimi/VSPTemplate

Attribute	Type	Description	Optionality
network	URI	A reference to the network to be associated with this VSP.	mandatory
vspConfig	URI	A reference to the VSP Configuration that will be used to create a VSP from this VSP Template.	mandatory

2411 The following describes the serialization of the entity in both JSON and XML:

2412 **JSON media type:** application/CIMI-VSPTemplate+json

2413 **JSON serialization:**

```
2414    { "uri": string,
2415      "name": string, ?
2416      "description": string, ?
2417      "created": string,  ?
2418      "properties": [ "name": string, + ], ?
2419      "network": { "href": string },
2420      "vspConfig": { "href": string },
2421      "operations": [
2422        { "rel": "edit", "href": string }, ?
2423        { "rel": "delete", "href": string } ?
2424      ] ?
```

```
2425          ...
2426        }
```

2427   **XML media type:** application/CIMI-VSPTemplate+xml

2428   **XML serialization:**

```
2429        <VSPTemplate xmlns="http://www.dmtf.org/cimi">
2430          <uri> xs:anyURI </uri>
2431          <name> xs:string </name> ?
2432          <description> xs:string </description> ?
2433          <created> xs:string </created>
2434          <property name="xs:string"> xs:string </property> *
2435          <network href="xs:anyURI"/>
2436          <vspConfig href="xs:anyURI"/>
2437          <operation rel="edit" href="xs:anyURI"/> ?
2438          <operation rel="delete" href="xs:anyURI"/> ?
2439          <xs:any>*
2440        </VSPTemplate>
```

2441   5.8.7.1 Operations

2442   This entity supports the Read, Update and Delete operations. Create is supported via the VSP Template
2443   Collection entity.

2444   5.8.8   VSP (Virtual Switch Port) Template Collection

2445   A VSP Template Collection entity represents the collection of VSP Templates within a Provider. This
2446   entity can be used to locate and create VSP Templates.

2447   **Type URI:** http://www.dmtf.org/cimi/VSPTemplateCollection

Attribute	Type	Description	Optionality
vspTemplates	URI[]	An array of URIs referencing the set of VSP Templates in the Provider.	optional

2448   The following describes the serialization of the entity in both JSON and XML:

2449   **JSON media type:** application/CIMI-VSPTemplateCollection+json

2450   **JSON serialization:**

```
2451        { "uri": string,
2452          "name": string, ?
2453          "description": string, ?
2454          "created": string,  ?
2455          "properties": [ "name": string, + ], ?
2456          "vspTemplates": [
2457            { "href": string }, +
2458          ], ?
2459          "operations": [
2460            { "rel": "add", "href": string }, ?
2461            { "rel": "edit", "href": string } ?
2462          ] ?
2463          ...
2464        }
```

2465   **XML media type:** application/CIMI-VSPTemplateCollection+xml

2466   **XML serialization:**

```
2467        <VSPTemplateCollection xmlns="http://www.dmtf.org/cimi">
2468          <uri> xs:anyURI </uri>
2469          <name> xs:string </name> ?
2470          <description> xs:string </description> ?
2471          <created> xs:string </created>
2472          <property name="xs:string"> xs:string </property> *
2473          <vspTemplate href="xs:anyURI"/> *
2474          <operation rel="add" href="xs:anyURI"/> ?
2475          <operation rel="edit" href="xs:anyURI"/> ?
2476          <xs:any>*
2477        </VSPTemplateCollection>
```

2478   **5.8.8.1 Operations**

2479   This entity supports the Read and Update operations. Creation of new VSP Template entities is
2480   supported via a POST to the "addLink" URI as described in section 4.2.1.1.

2481   **5.8.9   VSP (Virtual Switch Port) Configuration**

2482   The set of configuration values representing the information needed to create a VSP with certain
2483   characteristics.

2484   **Type URI:** http://www.dmtf.org/cimi/VSPConfiguration

Attribute	Type	Description	Optionality
bandwidthReservation	integer	Minimum Bandwidth requirements.	mandatory
trafficPriority	integer	Indicates priority of traffic on this network.	mandatory
maxTrafficDelay	integer	The requested maximum delay for end to end transmission specified in nanoseconds with uint64 (i.e. latency).	mandatory
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.	optional
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.	mandatory

2485   The following describes the serialization of the entity in both JSON and XML:

2486   **JSON media type:** application/CIMI-VSPConfiguration+json

2487   **JSON serialization:**

```
2488        { "uri": string,
2489          "name": string,
2490          "description": string, ?
2491          "created": string,  ?
```

```
2492            "properties": [ "name": string, + ], ?
2493            "bandwidthReservation": number,
2494            "trafficPriority": number,
2495            "maxTrafficDelay": number,
2496            "maxTrafficLoss": number, ?
2497            "maxTrafficJitter": number,
2498            "operations": [
2499              { "rel": "edit", "href": string }, ?
2500              { "rel": "delete", "href": string } ?
2501            ] ?
2502            ...
2503          }
```

2504 **XML media type:** application/CIMI-VSPConfiguration+xml

2505 **XML serialization:**

```
2506          <VSPConfiguration xmlns="http://www.dmtf.org/cimi">
2507            <uri> xs:anyURI </uri>
2508            <name> xs:string </name>
2509            <description> xs:string </description> ?
2510            <created> xs:string </created>
2511            <property name="xs:string"> xs:string </property> *
2512            <bandwidthReservation> xs:integer </bandwidthReservation>
2513            <trafficPriority> xs:integer </trafficPriority>
2514            <maxTrafficDelay> xs:integer </maxTrafficDelay>
2515            <maxTrafficLoss> xs:integer </maxTrafficLoss> ?
2516            <maxTrafficJitter> xs:integer </maxTrafficJitter>
2517            <operation rel="edit" href="xs:anyURI"/> ?
2518            <operation rel="delete" href="xs:anyURI"/> ?
2519            <xs:any>*
2520          </VSPConfiguration>
```

2521 **5.8.9.1 Operations**

2522 This entity supports the Read, Update and Delete operations. Create is supported via the VSP
2523 Configuration Collection entity.

2524 **5.8.10 VSP (Virtual Switch Port) Configuration Collection**

2525 A VSP Configuration Collection entity represents the collection of VSP Configurations within a Provider.
2526 This entity can be used to locate and create VSP Configurations.

2527 **Type URI:** http://www.dmtf.org/cimi/VSPConfigurationCollection

Attribute	Type	Description	Optionality
vspConfigurations	URI[]	An array of URIs referencing the set of VSP Configurations in the Provider.	optional

2528 The following describes the serialization of the entity in both JSON and XML:

2529 **JSON media type:** application/CIMI-VSPConfigurationCollection+json

2530 **JSON serialization:**

```
2531          { "uri": string,
2532            "name": string, ?
2533            "description": string, ?
2534            "created": string,  ?
2535            "properties": [ "name": string, + ], ?
```

```
2536            "vspConfigurations": [
2537              { "href": string }, +
2538            ], ?
2539            "operations": [
2540              { "rel": "add", "href": string }, ?
2541              { "rel": "edit", "href": string } ?
2542            ] ?
2543            ...
2544          }
```

2545 **XML media type:** application/CIMI-VSPConfigurationCollection+xml

2546 **XML serialization:**

```
2547            <VSPConfigurationCollection xmlns="http://www.dmtf.org/cimi">
2548              <uri> xs:anyURI </uri>
2549              <name> xs:string </name> ?
2550              <description> xs:string </description> ?
2551              <created> xs:string </created>
2552              <property name="xs:string"> xs:string </property> *
2553              <vspConfiguration href="xs:anyURI"/> *
2554              <operation rel="add" href="xs:anyURI"/> ?
2555              <operation rel="edit" href="xs:anyURI"/> ?
2556              <xs:any>*
2557            </VSPConfigurationCollection>
```

2558 **5.8.10.1   Operations**

2559 This entity supports the Read and Update operations. Creation of new VSP Configuration entities is
2560 supported via a POST to the "addLink" URI as described in section 4.2.1.1.

2561 **5.8.11 VSP (Virtual Switch Port)**

2562 A VSP represents the connection parameters of a network port.

2563 **Type URI:** http://www.dmtf.org/cimi/VSP

Attribute	Type	Description	Optionality
network	URI	A reference to the network associated with this VSP.	mandatory
state	string	An indicator of whether or not a specified port is on or off [Default = Enabled].	optional
bandwidthReservation	integer	Minimum Bandwidth requirements.	mandatory
trafficPriority	integer	Indicates priority of traffic on this network.	mandatory
maxTrafficDelay	integer	The requested maximum delay for end to end transmission specified in nanoseconds with uint64 (i.e. latency).	mandatory
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.	optional
maxTrafficJitter	integer	The requested maximum jitter for end to end	mandatory

		transmission with uint32 when traffic is packetized. The variation between packets arriving specified in nanoseconds with uint64.	
meters	URI[]	A list of references to Meters monitored for this VSP.	optional
eventLog	URI	A reference to the EventLog of this VSP.	optional

2564 The following describes the serialization of the entity in both JSON and XML:

2565 **JSON media type:** application/CIMI-VSP+json

2566 **JSON serialization:**

```
2567        { "uri": string,
2568          "name": string,
2569          "description": string, ?
2570          "created": string,  ?
2571          "properties": [ "name": string, + ], ?
2572          "network": { "href": string },
2573          "state": string, ?
2574          "bandwidthLimit": number,
2575          "trafficPriority": number,
2576          "maxTrafficDelay": number,
2577          "maxTfafficLoss": number, ?
2578          "maxTrafficJitter": number,
2579          "meters": [
2580            { "href": string }, +
2581          ], ?
2582          "eventLog": { "href": string }, ?
2583          "operations": [
2584            { "rel": "edit", "href": string }, ?
2585            { "rel": "delete", "href": string } ?
2586          ] ?
2587          ...
2588        }
```

2589 **XML media type:** application/CIMI-VSP+xml

2590 **XML serialization:**

```
2591        <VSP xmlns="http://www.dmtf.org/cimi">
2592          <uri> xs:anyURI </uri>
2593          <name> xs:string </name>
2594          <description> xs:string </description> ?
2595          <created> xs:string </created>
2596          <property name="xs:string"> xs:string </property> *
2597          <network href="xs:anyURI">
2598          <state> xs:string </state> ?
2599          <bandwidthLimit> xs:integer </bandwidthLimit>
2600          <trafficPriority> xs:integer </trafficPriority>
2601          <maxTrafficDelay> xs:integer </maxTrafficDelay>
2602          <maxTrafficLoss> xs:integer </maxTrafficLoss> ?
2603          <maxTrafficJitter> xs:integer </maxTrafficJiffer>
2604          <meter href="xs:anyURI"/> *
2605          <eventLog" href="xs:anyURI"/> ?
2606          <operation rel="edit" href="xs:anyURI"/> ?
2607          <operation rel="delete" href="xs:anyURI"/> ?
2608          <xs:any>*
2609        </VSP>
```

2610   **5.8.11.1   Operations**

2611   This entity supports the Read, Update and Delete operations. Create is supported via the VSP Collection
2612   entity.

2613   **5.8.12 VSP (Virtual Switch Port) Collection**

2614   A VSP Collection entity represents the collection of VSPs within a Provider. This entity can be used to
2615   locate and create VSPs.

2616   **Type URI:** http://www.dmtf.org/cimi/VSPCollection

Attribute	Type	Description	Optionality
vsps	URI[]	An array of URIs referencing the set of VSPs in the Provider.	optional

2617   The following describes the serialization of the entity in both JSON and XML:

2618   **JSON media type:** application/CIMI-VSPCollection+json

2619   **JSON serialization:**

```
2620            { "uri": string,
2621              "name": string, ?
2622              "description": string, ?
2623              "created": string,  ?
2624              "properties": [ "name": string, + ], ?
2625              "vsps": [
2626                { "href": string }, +
2627              ], ?
2628              "operations": [
2629                { "rel": "add", "href": string }, ?
2630                { "rel": "edit", "href": string } ?
2631              ] ?
2632              ...
2633            }
```

2634   **XML media type:** application/CIMI-VSPCollection+xml

2635   **XML serialization:**

```
2636            <VSPCollection xmlns="http://www.dmtf.org/cimi">
2637              <uri> xs:anyURI </uri>
2638              <name> xs:string </name> ?
2639              <description> xs:string </description> ?
2640              <created> xs:string </created>
2641              <property name="xs:string"> xs:string </property> *
2642              <vsp href="xs:anyURI"/> *
2643              <operation rel="add" href="xs:anyURI"/> ?
2644              <operation rel="edit" href="xs:anyURI"/> ?
2645              <xs:any>*
2646            </VSPCollection>
```

2647   **5.8.12.1   Operations**

2648   This entity supports the Read and Update operations.

2649   The following custom operations are also defined:

2650   • **Creating a New VSP**

2651   **/link@rel:** add

2652   This operation will create a new VSP.

2653   Input parameters: Either a reference to a VSP Template or a VSP Template itself.

2654   Output parameters: A reference to a new VSP and, optionally, the representation of the VSP.

2655   • **HTTP/REST Protocol**

2656   To create a new VSP a POST is sent to the "add" URI of the VSPCollection where the HTTP request
2657   body SHALL be as described below. Note this structure allows for certain properties to be passed in "by
2658   value" or by "reference". The definition of each property can be found in section 5.8.7.

2659   **JSON media type:** application/CIMI-VSPCreate+json

2660   **JSON serialization:**

```
2661        { "name": string,
2662          "description": string, ?
2663          "properties": [ "name": string, + ], ?
2664          "vspTemplate": { "href": string, ?
2665            "properties": [ "name": string, + ] ?
2666            "network": { "href": string }, ?
2667            "vspConfig": { "href": string, ?
2668              "properties": [ "name": string, + ], ?
2669              "bandwidthReservation": number, ?
2670              "trafficPriority": number, ?
2671              "maxTrafficDelay": number, ?
2672              "maxTrafficLoss": number, ?
2673              "maxTrafficJitter": number ?
2674            } ?
2675          }
2676          ...
2677        }
```

2678   **XML media type:** application/CIMI-VSPCreate+xml

2679   **XML serialization**

```
2680        <NetworkCreate>
2681          <name> xs:string </name>
2682          <description> xs:string </description> ?
2683          <property name="xs:string"> xs:string </property> *
2684          <vspTemplate href="xs:anyURI"? >
2685            <property name="xs:string"> xs:string </property> *
2686            <network href="xs:anyURI"/> ?
2687            <vspConfig href="xs:anyURI">
2688              <property name="xs:string"> xs:string </property> *
2689              <bandwidthReservation> xs:integer </bandwidthReservation> ?
2690              <trafficPriority> xs:integer </trafficPriority> ?
2691              <maxTrafficDelay> xs:integer </maxTrafficDelay> ?
2692              <maxTrafficLoss> xs:integer </maxTrafficLoss> ?
2693              <maxTrafficJitter> xs:integer </maxTrafficJitter> ?
2694            </vspConfig> ?
2695          </networkTemplate>
2696          <xs:any>*
2697        </NetworkCreate>
```

2698   The serialization of some reference properties are specified such that a request MAY either include a
2699   reference ("href") to an existing entity or to include the entity "inline" as a set of additional properties.
2700   Requests SHALL NOT include both a reference and the inlined set of properties.

2701   Upon successful processing of the request, the HTTP response body MAY either be empty or contain a
2702   serialization of the VSP entity.

5.9   Monitoring Entities and Relationships

2704   The following diagram illustrates the entities involved in tracking the progress of operations as well as
2705   metering and monitoring the status of other entities. Although this drawing is in the style of an Entity
2706   Relationship diagram, the use of UML is neither rigorous nor normative.
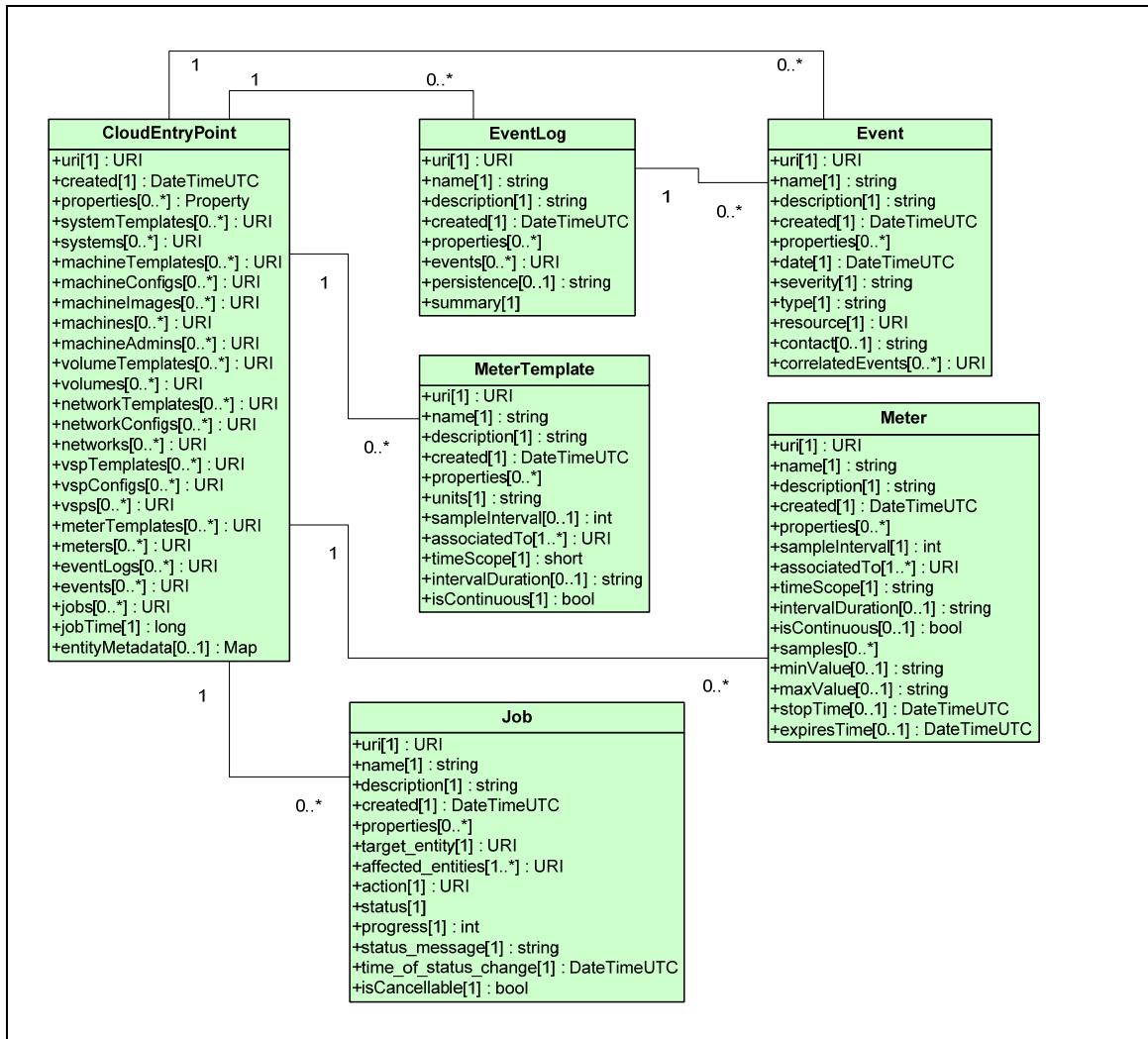
2708   **Figure 5 - Monitoring Entities**

5.9.1   Job

2710   This entity represents a process (i.e. a sequence of one or more operations directed to accomplish a
2711   specific goal) performed by the Provider.

2712   If a Provider supports exposing Job entities to Consumers then each request from a Consumer that would
2713   result in a change to the environment MUST result in a Job entity being created and a reference to that
2714   Job entity MUST be made available to the requestingConsumer. Providers MAY create additional Job
2715   entities for Provider initiated operations.

2716   **Type URI:** http://www.dmtf.org/cimi/Job

Attribute	Type	Description	Optionality
targetEntity	URI	A URI that indicates 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**.	mandatory
affectedEntities	URI[]	An array of URIs that indicates each of the entities upon which the operation is being performed. This array always includes the URI specified in the target_entity attribute. The values in this array are **read-only**.	mandatory
action	URI	A URI that indicates the type of action being performed. This attribute value is **immutable**.	mandatory
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**.	mandatory
progress	integer	An integer value in the range 0 … 100 that indicates the progress of this Job. This attribute value is **read-only**.	mandatory
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**.	mandatory
timeOfStatusChange	DateTimeUTC	A timestamp indicating the last time that the status of the operation changed. This attribute value is **read-only**.	mandatory
isCancellable	boolean	Specifies whether the task being performed by this Job object supports the Cancel operation.	mandatory

2717   The following describes the serialization of the entity in both JSON and XML:

2718   **JSON media type:** application/CIMI-Job+json

2719   **JSON serialization:**

```
2720              { "uri": string,
2721                "name": string,
2722                "description": string, ?
2723                "created": string,  ?
2724                "properties": [ "name": string, + ], ?
2725                "targetEntity": { "href": string },
2726                "affectedEntities": [
2727                  { "href": string }, +
```

```
2728          ], ?
2729          "action": string,
2730          "status": string,
2731          "progress": integer,
2732          "statusMessage": string,
2733          "timeOfStatusChange": date,
2734          "isCancellable": boolean,
2735          "operations": [
2736            { "rel": "edit", "href": string }, ?
2737            { "rel": "delete", "href": string } ?
2738          ] ?
2739          ...
2740        }
```

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

2742   **XML serialization:**

```
2743        <Job xmlns="http://www.dmtf.org/cimi">
2744          <uri> xs:anyURI </uri>
2745          <name> xs:string </name>
2746          <description> xs:string </description> ?
2747          <created> xs:string </created>
2748          <property name="xs:string"> xs:string </property> *
2749          <targetEntity href="xs:anyURI"/>
2750          <affectedEntity> xs:anyURI </affectedEntity> *
2751          <action> xs:anyURI </action>
2752          <status> xs:string </status>
2753          <progess> xs:integer <progress>
2754          <statusMessage> xs:string </statusMessage>
2755          <timeOfStatusChange> xs:dateTime </timeOfStatusChange>
2756          <isCancellable> xs:boolean </isCancellable>
2757          <operation rel="edit" href="xs:anyURI"/> ?
2758          <operation rel="delete" href="xs:anyURI"/> ?
2759          <xs:any>*
2760        </Job>
```

2761   **5.9.1.1 Operations**

2762   This entity supports the Read, Update and Delete operations.

2763   **5.9.2   Job Collection**

2764   A Job Collection entity represents the collection of Jobs within a Provider. This resource can be used to
2765   locate Jobs.

2766   **Type URI:** http://www.dmtf.org/cimi/JobCollection

Attribute	Type	Description	Optionality
jobs	URI[]	An array of URIs referencing the set of Jobs in the Provider.	optional

2767   The following describes the serialization of the entity in both JSON and XML:

2768   **JSON media type:** application/CIMI-JobCollection+json

2769   **JSON serialization:**

```
2770        { "uri": string,
2771          "name": string, ?
2772          "description": string, ?
2773          "created": string,  ?
```

```
2774          "properties": [ "name": string, + ], ?
2775          "jobs": [
2776            { "href": string }, +
2777          ], ?
2778          "operations": [
2779            { "rel": "edit", "href": string } ?
2780          ] ?
2781          ...
2782        }
```

2783   **XML media type:** application/CIMI-JobCollection+xml

2784   **XML serialization:**

```
2785        <JobCollection xmlns="http://www.dmtf.org/cimi">
2786          <uri> xs:anyURI </uri>
2787          <name> xs:string </name> ?
2788          <description> xs:string </description> ?
2789          <created> xs:string </created>
2790          <property name="xs:string"> xs:string </property> *
2791          <job href="xs:anyURI"/> *
2792          <operation rel="edit" href="xs:anyURI"/> ?
2793          <xs:any>*
2794        </JobCollection>
```

2795   **5.9.2.1 Operations**

2796   This entity supports the Read and Update operations.

2797   **5.9.3   Meter Template**

2798   A Meter Template represents the definition of a Meter. A Meter Template can only be created by the
2799   Provider.

2800   **Type URI:** http://www.dmtf.org/cimi/MeterTemplate

Attribute	Type	Description	Optionality
units	string	Name of the used units, e.g. kilobits per second, CPU usage percentage, etc. This attribute is **read-only**.	mandatory
sampleInterval	integer	It indicates the time between consecutive samples in seconds. This attribute is **read-only**.	mandatory
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.uri - a URI that uniquely identifies an entity type. This attribute is **read-only**.	mandatory
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**.	mandatory
intervalDuration	string	It indicates the interval duration when the timeScope is set to "Interval". Possible values: hourly, daily, weekly, monthly or	mandatory

		yearly. This attribute is **read-only**.	
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**.	mandatory

2801   The following describes the serialization of the entity in both JSON and XML:

2802   **JSON media type:** application/CIMI-MeterTemplate+json

2803   **JSON serialization:**

```
2804          { "uri": string,
2805            "name": string, ?
2806            "description": string, ?
2807            "created": string,  ?
2808            "properties": [ "name": string, + ], ?
2809            "units": string,
2810            "sampleInterval": integer,
2811            "associatedTo": [
2812              { "href": string }, +
2813            ], ?
2814            "timeScope": string,
2815            "intervalDuration": string,
2816            "isContinuous": boolean,
2817            "operations": [
2818              { "rel": "edit", "href": string }, ?
2819              { "rel": "delete", "href": string } ?
2820            ] ?
2821            ...
2822          }
```

2823   **XML media type:** application/CIMI-MeterTemplate+xml

2824   **XML serialization:**

```
2825          <MeterTemplate xmlns="http://www.dmtf.org/cimi">
2826            <uri> xs:anyURI </uri>
2827            <name> xs:string </name> ?
2828            <description> xs:string </description> ?
2829            <created> xs:string </created>
2830            <property name="xs:string"> xs:string </property> *
2831            <units> xs:string </units>
2832            <sampleInterval> xs:integer </sampleInterval>
2833            <associatedTo href="xs:anyURI"/> *
2834            <timeScope> xs:string </timeScope>
2835            <intervalDuration> xs:string </intervalDuration>
2836            <isContinuous> xs:boolean </isContinuous>
2837            <operation rel="edit" href="xs:anyURI"/> ?
2838            <operation rel="delete" href="xs:anyURI"/> ?
2839            <xs:any>*
2840          </MeterTemplate>
```

2841   **5.9.3.1 Operations**

2842   This entity supports the Read, Update and Delete operations. Create is supported via the Meter Template
2843   entity.

2844   **5.9.4   Meter Template Collection**

2845   A Meter Template Collection entity represents the collection of Meter Templates within a Provider. This
2846   entity can be used to locate Meter Templates.

2847   **Type URI:** http://www.dmtf.org/cimi/MeterTemplateCollection

Attribute	Type	Description	Optionality
meterTemplates	URI[]	An array of URIs referencing the set of Meter Templates in the Provider.	optional

2848   The following describes the serialization of the entity in both JSON and XML:

2849   **JSON media type:** application/CIMI-MeterTemplateCollection+json

2850   **JSON serialization:**

```
2851        { "uri": string,
2852          "name": string, ?
2853          "description": string, ?
2854          "created": string,  ?
2855          "properties": [ "name": string, + ], ?
2856          "meterTemplates": [
2857            { "href": string }, +
2858          ], ?
2859          "operations": [
2860            { "rel": "add", "href": string } ?
2861            { "rel": "edit", "href": string } ?
2862          ] ?
2863          ...
2864        }
```

2865   **XML media type:** application/CIMI-MeterTemplateCollection+xml

2866   **XML serialization:**

```
2867        <MeterTemplateCollection xmlns="http://www.dmtf.org/cimi">
2868          <uri> xs:anyURI </uri>
2869          <name> xs:string </name> ?
2870          <description> xs:string </description> ?
2871          <created> xs:string </created>
2872          <property name="xs:string"> xs:string </property> *
2873          <meterTemplate href="xs:anyURI"/> *
2874          <operation rel="add" href="xs:anyURI"/> ?
2875          <operation rel="edit" href="xs:anyURI"/> ?
2876          <xs:any>*
2877        </MeterTemplateCollection>
```

2878   **5.9.4.1 Operations**

2879   This entity supports the Read and Update operations. Creation of new Meter Template entities is
2880   supported via a POST to the "addLink" URI as described in section 4.2.1.1.

2881   **5.9.5   Meter**

2882   This entity represents an available Meter of some property associated to a given entity.

2883   **Type URI:** http://www.dmtf.org/cimi/Meter

Attribute	Type	Description	Optionality
units	string	Name of the used units, e.g. kilobits per second, CPU usage percentage, etc. This attribute is **read-only**.	mandatory

sampleInterval	integer	It indicates the time between consecutive samples in seconds. This attribute is **read-only**.	mandatory
associatedTo	URI[]	An array of URIs that indicate the entities to which this Meter can be applied. The value space of these URIs is identical to that of EntityMetadata.uri - a URI that uniquely identifies an entity type.This attribute is **read-only**.	mandatory
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 Metrer 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**.	mandatory
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**.	mandatory
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**.	mandatory
samples	A list of taken samples Each sample attribute has the following sub-attributes:<table><tr><th>Attribute</th><th>Type</th><th>Description</th></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><tr><td>value</td><td>string</td><td>It indicates the sampled value of the measure</td></tr></table>		mandatory
minValue	string	It indicates the expected minimal measure value. This attribute is **read-only**.	optional
maxValue	string	It indicates the expected maximum measure value. This attribute is **read-only**.	optional
stopTime	dateTimeUTC	It indicates a time from which the meter stops tracking samples. This attribute is **writable**.	optional
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**.	

2884   The following describes the serialization of the entity in both JSON and XML:

2885   **JSON media type:** application/CIMI-Meter+json

2886   **JSON serialization:**

```
2887        { "uri": string,
2888          "name": string, ?
2889          "description": string, ?
2890          "created": string,  ?
2891          "properties": [ "name": string, + ], ?
2892          "units": string,
2893          "sampleInterval": integer,
2894          "associatedTo": [
2895            { "href": string }, +
2896          ], ?
2897          "timeScope": string,
2898          "intervalDuration": string,
2899          "isContinuous": boolean,
2900          "samples": [
2901            { "timestamp": string, "value": string }, +
2902          ], ?
2903          "minValue": string, ?
2904          "maxValue": string, ?
2905          "stopTime": string, ?
2906          "expiresTime": string, ?
2907          "operations": [
2908            { "rel": "edit", "href": string }, ?
2909            { "rel": "delete", "href": string } ?
2910          ] ?
2911          ...
2912        }
```

2913   **XML media type:** application/CIMI-Meter+xml

2914   **XML serialization:**

```
2915        <Meter xmlns="http://www.dmtf.org/cimi">
2916          <uri> xs:anyURI </uri>
2917          <name> xs:string </name> ?
2918          <description> xs:string </description> ?
2919          <created> xs:string </created>
2920          <property name="xs:string"> xs:string </property> *
2921          <units> xs:string </units>
2922          <sampleInterval> xs:integer </sampleInterval>
2923          <assocatedTo href="xs:anyURI"/> *
2924          <timeScope> xs:string <timeScope>
2925          <intervalDuration xs:string </intervalDuration>
2926          <isContinuous> xs:boolean </isContinuous>
2927          <sample timestamp="xs:dateTime" value="xs:string"/> *
2928          <minValue> xs:string </minValue> ?
2929          <maxValue> xs:string </maxValue> ?
2930          <stopTime> xs:dateTime </stopTime> ?
2931          <expiresTime> xs:dateTime </expiresTime> ?
2932          <operation rel="edit" href="xs:anyURI"/> ?
2933          <operation rel="delete" href="xs:anyURI"/> ?
2934          <xs:any>*
2935        </Meter>
```

2936   **5.9.5.1 Operations**

2937   This entity supports the Read, Update and Delete operations. Create is supported via the Meter
2938   Collection entity.

2939 **5.9.6 Meter Collection**

2940 A Meter Collection entity represents the collection of Meters within a Provider. This enity can be used to
2941 locate and create Meters.

2942 **Type URI:** http://www.dmtf.org/cimi/MeterCollection

Attribute	Type	Description	Optionality
meters	URI[]	An array of URIs referencing the set of Meters in the Provider.	optional

2943 The following describes the serialization of the entity in both JSON and XML:

2944 **JSON media type:** application/CIMI-MeterCollection+json

2945 **JSON serialization:**

```
2946       { "uri": string,
2947         "name": string, ?
2948         "description": string, ?
2949         "created": string,  ?
2950         "properties": [ "name": string, + ], ?
2951         "meters": [
2952           { "href": string }, +
2953         ], ?
2954         "operations": [
2955           { "rel": "add", "href": string }, ?
2956           { "rel": "edit", "href": string } ?
2957         ] ?
2958         ...
2959       }
```

2960 **XML media type:** application/CIMI-MeterCollection+xml

2961 **XML serialization:**

```
2962       <MeterCollection xmlns="http://www.dmtf.org/cimi">
2963         <uri> xs:anyURI </uri>
2964         <name> xs:string </name> ?
2965         <description> xs:string </description> ?
2966         <created> xs:string </created>
2967         <property name="xs:string"> xs:string </property> *
2968         <meter href="xs:anyURI"/> *
2969         <operation rel="add" href="xs:anyURI"/> ?
2970         <operation rel="edit" href="xs:anyURI"/> ?
2971         <xs:any>*
2972       </MeterCollection>
```

2973 **5.9.6.1 Operations**

2974 This entity supports the Read and Update operations.

2975 The following custom operations are also defined:

2976 • **Creating a New Meter**

2977 **/link@rel:** add

2978 This operation will create a new Meter.

2979 Input parameters: Either a reference to a Meter Template or a Meter Template itself.

2980   Output parameters: A reference to a new Meter and optionally the representation of the Meter.

2981   • **HTTP/REST Protocol**

2982   To create a new Meter a POST is sent to the "add" URI of the MeterCollection where the HTTP request
2983   body SHALL be as described below. Note this structure allows for certain properties to be passed in "by
2984   value" or by "reference". The definition of each property can be found in section 5.9.3.

2985   **JSON media type:** application/CIMI-MeterCreate+json

2986   **JSON serialization:**

```
2987            { "name": string,
2988              "description": string, ?
2989              "properties": [ "name": string, + ], ?
2990              "meterTemplate": { "href": string, ?
2991                "properties": [ "name": string, + ], ?
2992                "units": string, ?
2993                "sampleInterval": integer, ?
2994                "associatedTo": [
2995                   { "href": string }, +
2996                ], ?
2997                "timeScope": string, ?
2998                "intervalDuration": string, ?
2999                "isContinuous": boolean, ?
3000              }
3001              ...
3002            }
```

3003   **XML media type:** application/CIMI-MeterCreate+xml

3004   **XML serialization**

```
3005            <MeterCreate>
3006              <name> xs:string </name>
3007              <description> xs:string </description> ?
3008              <property name="xs:string"> xs:string </property> *
3009              <meterTemplate href="xs:anyURI"? >
3010                <property name="xs:string"> xs:string </property> *
3011                <units> xs:string </units> ?
3012                <sampleInterval> xs:integer </sampleInterval> ?
3013                <associatedTo href="xs:anyURI"/> *
3014                <timeScope> xs:string </timeScope> ?
3015                <intervalDuration> xs:string </intervalDuration> ?
3016                <isContinuous> xs:boolean </isContinuous> ?
3017              </meterTemplate>
3018              <xs:any>*
3019            </MeterCreate>
```

3020   The serialization of some reference properties are specified such that a request MAY either include a
3021   reference ("href") to an existing entity or to include the entity "inline" as a set of additional properties.
3022   Requests SHALL NOT include both a reference and the inlined set of properties.

3023   Upon successful processing of the request, the HTTP response body MAY either be empty or contain a
3024   serialization of the Meter entity.

3025   **5.9.7   Event Log**

3026   An entity that represents a registry of Events.

3027   **Type URI:** http://www.dmtf.org/cimi/EventLog

Attribute	Type	Description	Optionality
events	URI[]	A list of references to occurred Events.	mandatory
persistence	string	A value that indicates the persistence of the Events within the EventLog. For instance, daily, weekly, monthly or yearly. This attribute is **writable**.	mandatory
summary		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: | Attribute	Type	Description
---	---	---
low	integer	Number of occurred Events with a low severity.
medium	integer	Number of occurred Events with a medium severity.
high	integer	Number of occurred Events with a high severity
critical	integer	Number of occurred Events with a critical severity.		mandatory

3028 The following describes the serialization of the entity in both JSON and XML:

3029 **JSON media type:** application/CIMI-EventLog+json

3030 **JSON serialization:**

```
3031    { "uri": string,
3032      "name": string, ?
3033      "description": string, ?
3034      "created": string,   ?
3035      "properties": [ "name": string, + ], ?
3036      "events": [
3037        { "href": string }, +
3038      ], ?
3039      "persistence", string,
3040      "summary", {
3041        "low": number,
3042        "medium": number,
3043        "high": number,
3044        "critical": number
3045      }, ?
3046      "operations": [
3047        { "rel": "edit", "href": string }, ?
3048        { "rel": "delete", "href": string } ?
3049      ] ?
3050      ...
3051    }
```

3052 **XML media type:** application/CIMI-EventLog+xml

3053 **XML serialization:**

```
3054    <EventLog xmlns="http://www.dmtf.org/cimi">
3055      <uri> xs:anyURI </uri>
3056      <name> xs:string </name> ?
3057      <description> xs:string </description> ?
3058      <created> xs:string </created>
```

```
3059          <property name="xs:string"> xs:string </property> *
3060          <event href="xs:anyURI"/> *
3061          <persistence> xs:string </persistence>
3062          <summary>
3063            <low> xs:integer </low>
3064            <medium> xs:integer </medium>
3065            <high> xs:integer <high>
3066            <critical> xs:integer </critical>
3067          </summary>
3068          <operation rel="edit" href="xs:anyURI"/> ?
3069          <operation rel="delete" href="xs:anyURI"/> ?
3070          <xs:any>*
3071        </EventLog>
```

3072   **5.9.7.1 Operations**

3073   This entity supports the Read, Update and Delete operations.

3074   **5.9.8   Event Log Collection**

3075   A Event Log Collection entity represents the collection of Event Logs within a Provider. This resource can
3076   be used to locate EventLogs.

3077   **Type URI:** http://www.dmtf.org/cimi/EventLogCollection

Attribute	Type	Description	Optionality
eventLogs	URI[]	An array of URIs referencing the set of Event Logs in the Provider.	optional

3078   The following describes the serialization of the entity in both JSON and XML:

3079   **JSON media type:** application/CIMI-EventLogCollection+json

3080   **JSON serialization:**

```
3081          { "uri": string,
3082            "name": string, ?
3083            "description": string, ?
3084            "created": string,  ?
3085            "properties": [ "name": string, + ], ?
3086            "eventLogs": [
3087              { "href": string }, +
3088            ], ?
3089            "operations": [
3090              { "rel": "edit", "href": string } ?
3091            ] ?
3092            ...
3093          }
```

3094   **XML media type:** application/CIMI-EventLogCollection+xml

3095   **XML serialization:**

```
3096        <EventLogCollection xmlns="http://www.dmtf.org/cimi">
3097          <uri> xs:anyURI </uri>
3098          <name> xs:string </name> ?
3099          <description> xs:string </description> ?
3100          <created> xs:string </created>
3101          <property name="xs:string"> xs:string </property> *
3102          <eventLog href="xs:anyURI"/> *
3103          <operation rel="edit" href="xs:anyURI"/> ?
3104          <xs:any>*
```

3105
```
</EventLogCollection>
```

5.9.8.1 Operations

3107   This entity supports the Read and Update operations.

5.9.9   Event

3109   An entity that represents the notification of an event within the managed infrastructure. Some examples of
3110   Events may be:

3111   • Machine X has been rebooted by guest OS

3112   • Machine X is not responding to platform services

3113   • A new vCPU has been added to machine X following defined elasticity rules

3114   The scope of the Event concept is any kind of information that the Provider is able to track within its
3115   infrastructure and that can constitute useful information for the consumer. Possible examples, but not
3116   limited to, are errors and inconveniences that occur in the (virtual) resources assigned to consumers,
3117   some provider initiated actions such as maintenance tasks, etc.

3118   **Type URI:** http://www.dmtf.org/cimi/Event

Attribute	Type	Description	Optionality
eventTime	dateTimeUTC	The time and date of creation of the Event. This attribute is **immutable**.	mandatory
type	string	A value that indicates the kind of Event (informational, error, alarm, etc.). This attribute is **read-only**.	mandatory
severity	string	A value indicating the Event severity.Possible values are: critical, high, medium or low. This attribute is **read-only**.	mandatory
resource	URI	The identifier of the resource the Event refers to (machine, network, volume, etc.). This attribute is **immutable**.	mandatory
contact	string	An optional identifier that references a contact point to solve the problem (helpdesk, technical staff, etc.). This attribute is **read-only**.	optional
correlatedEvents	URI[]	A list of Event identifiers whose notifications are correlated with (related to) this one.	optional

3119   The following describes the serialization of the entity in both JSON and XML:

3120   **JSON media type:** application/CIMI-Event+json

3121   **JSON serialization:**

3122
3123
3124
3125
3126
```
{ "uri": string,
  "name": string, ?
  "description": string, ?
  "created": string,  ?
  "properties": [ "name": string, + ], ?
```

```
3127            "eventTime": string,
3128            "type": string,
3129            "severity": string,
3130            "resource": { "href": string },
3131            "contact": string, ?
3132            "correlatedEvents": [
3133              { "href": string }, +
3134            ], ?
3135            "operations": [
3136              { "rel": "edit", "href": string }, ?
3137              { "rel": "delete", "href": string } ?
3138            ] ?
3139            ...
3140          }
```

3141   **XML media type:** application/CIMI-Event+xml

3142   **XML serialization:**

```
3143          <Event xmlns="http://www.dmtf.org/cimi">
3144            <uri> xs:anyURI </uri>
3145            <name> xs:string </name> ?
3146            <description> xs:string </description> ?
3147            <created> xs:string </created>
3148            <property name="xs:string"> xs:string </property> *
3149            <eventTime> xs:dateTime </eventTime>
3150            <type> xs:string </type>
3151            <severity> xs:string </severity>
3152            <resource href="xs:anyURI"/>
3153            <contact> xs:string </contact> ?
3154            <correlatedEvent href="xs:anyURI"/> *
3155            <operation rel="edit" href="xs:anyURI"/> ?
3156            <operation rel="delete" href="xs:anyURI"/> ?
3157            <xs:any>*
3158          </Event>
```

3159   **5.9.9.1 Operations**

3160   This entity supports the Read, Update and Delete operations.

3161   **5.9.10 Event Collection**

3162   An Event Collection entity represents the collection of Events within a Provider. This entity can be used to
3163   locate Events.

3164   **Type URI:** http://www.dmtf.org/cimi/EventCollection

Attribute	Type	Description	Optionality
events	URI[]	An array of URIs referencing the set of Events in the Provider.	optional

3165   The following describes the serialization of the entity in both JSON and XML:

3166   **JSON media type:** application/CIMI-EventCollection+json

3167   **JSON serialization:**

```
3168          { "uri": string,
3169            "name": string, ?
3170            "description": string, ?
3171            "created": string,  ?
3172            "properties": [ "name": string, + ], ?
```

```
3173        "events": [
3174          { "href": string }, +
3175        ], ?
3176        "operations": [
3177          { "rel": "edit", "href": string } ?
3178        ] ?
3179        ...
3180      }
```

3181   **XML media type:** application/CIMI-EventCollection+xml

3182   **XML serialization:**

```
3183        <EventCollection xmlns="http://www.dmtf.org/cimi">
3184          <uri> xs:anyURI </uri>
3185          <name> xs:string </name> ?
3186          <description> xs:string </description> ?
3187          <created> xs:string </created>
3188          <property name="xs:string"> xs:string </property> *
3189          <event href="xs:anyURI"/> *
3190          <operation rel="edit" href="xs:anyURI"/> ?
3191          <xs:any>*
3192        </EventCollection>
```

3193   **5.9.10.1   Operations**

3194   This entity supports the Read and Update operations.

3195   6   Scenarios

3196   6.1   Initial Scenario

3197   The following scenarios serves as an initial starting point:

3198   A cloud consumer has a machine (e.g. a machine image plus some configuration information) that
3199   she wants to run on virtual resources provided by a cloud. The image for this machine already
3200   exists within the cloud infrastructure. Using that image, she requests the cloud to create a
3201   Machine instance. The request may specify configurable options, such as the number of virtual
3202   CPUs, memory size, disk size, etc. The provider will provision the required resources and deploy
3203   the Machine. The consumer may start and stop the Machine.

3204   This scenario is composed of the following use cases from the "Scoping Framework for Cloud
3205   Management Models and Protocol Requirements" document [ref needed]. Note that all statements to the
3206   effect of "he/she does foo" should be construed as expanding to "he/she causes some piece of
3207   consumer-side software to do foo".

3208   6.1.1   Create and deploy a Machine using a Provider created Machine Template

3209   Steps to create a machine using an existing Machine Template:

3210   1. Consumer makes a request to list all machine templates from the Provider.

3211   2. The Provider gives the Consumer a list of machine templates that the Consumer has access to.

3212   3. The Consumer browses the Machine Templates and gets details of each template. The details of
3213   the template contains the following information:

3214   a.   name of the template

b. a textual description of the template – this could include information such as operating system of the machine image that the template references

c. a unique reference to the template

d. the hardware profile or configuration referenced by the template such as # of CPUs, assigned disk capacity, and networks that the machine is configured to connect to

e. the name of the image that the template references - this could be the guest operating system and could include names of applications that may be built into the image by the Provider

4. Consumer selects the template of interest and makes a request to the Provider to create the machine by providing the following information

a. desired name of the machine

b. some description text for the machine

c. reference to the selected Machine Template

5. If the template indicates that the Consumer must provide specialization information such as administrator password, licensing keys etc, the Consumer supplies the information

6. The Provider accepts the request and provides a Job id to the Consumer. The Provider creates the Machine and deploys it to the Consumer's cloud.

7. The Consumer can use the Job id to track progress of the machine creation.

8. Once the Job status shows Job as being completed successfully, the Consumer can start using the Machine. If the Consumer issues a Machine list query for his/her cloud, the created Machine is returned in the list.

6.1.2 Create a Machine by passing a Machine Template by value

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

a. reference to a Machine Configuration

b. reference to a Machine Image

c. reference to a Machine Admin entity

d. reference to a Volume Template / Volume Configuration + Volume Image

e. reference to a Network Template / Network Configuration

6.1.3 Create a Machine using a Consumer created Machine Template

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

1. Create and save a Machine Template. See the scenarios for creating a Machine Template.

2. Create a Machine by passing a reference to a Machine Template.

6.1.4 Create a Machine Template by specifying individual components

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

3250   1.   Consumer chooses the components for the Machine Template, such as:

3251   a.   a Machine Configuration

3252   b.   a Machine Image

3253   c.   a Machine Admin

3254   d.   a Volume Template / Volume Configuration + Volume Image

3255   e.   a Network Template / Network Configuration

3256   2.   Consumer makes a request to create a Machine Template by building a MachineTemplate that
3257   contains references to above chosen components. If a Consumer does not specify some
3258   components, the Provider can return an error or use default values.

3259   3.   The Provider validates that required components are present and creates the Machine Template.
3260   The Provider informs the Consumer of the location of the Machine Template. If the Consumer
3261   makes a request to browse Consumer created Machine Templates, this new template appears in
3262   the list.

3263   6.1.5   Create a Machine Template from a template file

3264   1.   Consumer starts the create template process. He passes a reference to a file that contains
3265   metadata for creating a Machine Template.

3266   2.   Provider accepts the request and provides a Job id to the Consumer.

3267   3.   The Provider parses the template file, validates it and creates the template. If required
3268   components are not present, the Provider can return an error or use default values.

3269   4.   The Provider informs the Consumer of the location of the template. If the Consumer makes a
3270   request to browse Consumer created Machine Templates, this new template appears in the list.

3271   6.1.6   Control System State (CMWG065)

3272   Our consumer controls the state of her Machine by Updating the status attribute of the entity
3273   corresponding to that Machine. For example, updating the value to STOPPED would stop the Machine.

3274   6.2   Machine Image Scenarios

3275   6.2.1   Create new Machine Image from an image file

3276   1.   Consumer starts the create Machine Image process. She passes a reference to the image file
3277   (e.g. an OVF file).

3278   2.   Provider accepts the request, validates the request, creates the Machine Image and saves it to
3279   the Consumer's cloud.

3280   3.   If the Consumer makes a request to browse Consumer created Machine Images, this new image
3281   appears in the list.

3282   6.2.2   Create new Machine Image from Machine instance

3283   1.   Consumer selects a Machine instance from his Machine list

3284   2.   Consumer requests to make an image of the selected Machine instance by providing a name and
3285   a location to save the image to.

3286   3. Provider accepts the request and creates the machine image and saves it to the location provided
3287   by the Consumer

3288   4. If the Consumer makes a request to browse Consumer created machine images, this new image
3289   appears in the list

6.3   System Scenario

3291 The following scenario serves to illustrate the creation of a System as a composite of Machines, Volumes,
3292 and Networks:

3293   A cloud consumer has a system template (e.g., machine, machine image, network, storage) that
3294   it wants to run on virtual resources provided by a cloud. It will upload the template and request
3295   the cloud to deploy it. The request may specify configurable options, such as the number of
3296   virtual system targets. The provider will provision the required resources and deploy the template.
3297   The provisioned resources will be monitored and data made available to the cloud consumer. The
3298   consumer may start, modify, and stop the system. The consumer will terminate its use when it is
3299   complete.

6.3.1   List System Templates (CMWG010)

3301 Since our cloud consumer is smart/lazy, the first thing she does is to verify that the System she wants to
3302 deploy does not already have an existing template in her target Site. By performing a `Read` operation on
3303 her target Site, our consumer obtains a list of references to the System Templates. Subsequent `Read`
3304 operations on the referenced System Templates allows our consumer to browse the available System
3305 Templates, and, via their `name` and `description` attributes, determine if her template is among them.

6.3.2   Create System Template

3307 Having ascertained that a template of her desired System does not already exist within her target site, our
3308 consumer creates a System Template by posting a System Template to the System Template Collection.

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

3311 Once her template is created, our consumer creates her System via that entity's `Create` operation using a
3312 reference to the recently created System Template. This action recursively creates Machines, Volumes,
3313 and Networks (along with their interrelationships) for every Machine Template, Volume Template, and
3314 Network Templates contained by the System Template.

3315 Our consumer monitors the progress of the `Create` operation by using the `Read` operation to poll `state`
3316 attribute of the newly created System and/or, optionally, using the `Read` operation to poll an associated
3317 `Job` entity.

6.3.4   Get Monitoring Information (CMWG066)

3319 Our consumer monitors her System and its constituent Machines, Volumes, and Networks by `Read`ing the
3320 appropriate entity.

6.3.5   Control System State (CMWG065)

3322 Our consumer controls the state of her System by `Updat`ing the `status` attribute of the entity
3323 corresponding to that System. For example, updating the value to STOPPED would stop the System.
3324 This action recursively updates the `status` attributes of all Machines contained within her system, with
3325 corresponding effects on the operational state of those machines. Individual Machines can be controlled
3326 in a similar fashion.

3327   **6.3.6   Remove System from a Site (CMWG051)**

3328   When our consumer is finished with her System, she removes it by invoking the `Delete` operation on the
3329   entity corresponding to that System. This action recursively removes the Machines, Volumes, and
3330   Networks contained within her system. The removal of Volumes and Networks would obviously be
3331   contingent on whether or not these entities were shared by other, active Machines.

3332   # 7   Security

3333   This specification considers two separate but related security domains. The first domain, API-level
3334   security, concerns the protection of the entities modeled by this specification. For example, insuring that
3335   unauthorized users are not allowed to alter a Machine instance. The second domain, resource-level
3336   security, deals with the protection of the underlying resources represented by these entities. For example,
3337   insuring that unauthorized users cannot login to the Linux instance corresponding to that Machine.

3338   ## 7.1   API Level Security

3339   ### 7.1.1   Authentication

3340   Except in cases where the access control policy allows for anonymous requests, the Provider SHALL
3341   authenticate all request messages and determine the identity of theConsumer. The techniques used to
3342   authenticate messages are outside the scope of this specification.

3343   Protocol bindings of the CIMI Model specification are encouraged to include requirements for the most
3344   common authentication mechanisms applicable to that protocol (e.g. the use of BasicAuth for protocols
3345   using HTTP).

3346   ### 7.1.2   Message Integrity

3347   Messages exchanged between the Consumer and the Provider SHOULD have message integrity
3348   protections applied. The mechanisms used to provide message integrity are outside the scope of this
3349   specification.

3350   Protocol bindings of the CIMI Model specification are encouraged to include requirements for the most
3351   common integrity mechanisms applicable to that protocol (e.g. the use of TLS for protocols using HTTP).

3352   ### 7.1.3   Message Confidentiality

3353   Messages exchanged between the Consumer and the Provider MAY have message confidentiality
3354   protections applied. The mechanisms used to provide message confidentiality are outside the scope of
3355   this specification.

3356   ### 7.1.4   Authorization

3357   The Provider SHOULD process messages only if authorized by access control policy, which may
3358   reference the Consumer's identity, the message type and content, and other contextual information when
3359   making this decision. The language in which this access control policy is expressed as well as the
3360   process by which these authorization decisions are made are outside the scope of this specification.

3361   ### 7.1.5   Multi-Tenancy

3362   In cases where a Provider uses multi-tenancy to support a set of Consumers, the operations in this
3363   specification are modeled under the assumption that each Consumer's view of the system (i.e. which
3364   entities are visible, discoverable, and accessible) is scoped to those entities provisioned for or created by
3365   that Consumer. To the Consumer it appears that the Provider is implementing a sole-use instance of the

3366 CIMI API (albeit one who's non-functional characteristics may be influenced the actions of invisible co-
3367 Consumers).

7.2   Resource Level Credentials

3369 This specification intentionally avoids constraining the type, nature, or operation of the resources
3370 represented by the entities that it defines. It is therefore outside the scope of this specification to define
3371 the mechanism(s) used to access the resource represented by the Machine entity. There is, however, an
3372 integration point between this specification and such mechanisms, namely the management of the
3373 credentials (user names, passwords, keys, etc.) used to provision such access. This information is
3374 encapsulated by the Machine Admin entity (described in Section 0).

3375   **Annex A (informative) – Change Log**

3376

Version	Date	Who	What
0.0.1	10/15/10	Gil, Jack	Initial Draft
0.0.2	10/19/10	Jack	Adding the attribute descriptions and high level operational descriptions on the entities
0.0.3	10/29/10	Gil	Add section on the "Initial Scenario" and the mapping of its required use cases to our model.
0.0.4	11/15/10	Gil	Removed 'definition' attribute from System Template, Machine Template, Volume Template, and Network Template (per 912). Added "networkInterfaces" attribute to Machine with sub-properties that define IP address – added "protocol", "subnet_mask", "default_gateway" and "dns_servers to Netowork entity (per 910). Removed inline issues and created issues 928, 929, and 930.
0.0.5	11/17/10	Gil	Change "Cloud Site" to "Site per 882. Added Job entity and removed 'progress' attributes per 911. Added structure to Machine/disks and Machine/volumes per 915.
0.0.6	12/01/10	Gil	Removed "jobs" attributes from System and System Template to complete 911. Added "capacity" and "format", removed "type" from Machine/volumes to complete 915.
0.0.7	12/10/10	Gil	Added Image entity to resolve 935. Added new initial scenario to resolve 994.
0.0.8	12/15/10	Gil	Removed "os" attribute from Machine Template to resolve 1032. Added quantity/units sub-properties to describe memory and disk sizes and capacities for Machines and Machine Templates to resolve 1009. Removed "based_on" attribute from System, Machine, Volume, and Network to resolve 1001. Removed all template property descriptions to the effect that "changes to [this attribute] should correspondingly evaluate the [objects] that have been instantiated based on this [object template]" to resolve 1005.
0.0.9	01/05/11	Gil	Add Update operation to all entities as resolution to 1003. Change "Image" entity to "Machine Image" and updated description to resolve 1026. Fixed some capitalization and terminology inconsistencies.
0.0.10	01/18/11	Gil	Add Provider entity as resolution to 1043. Change "params" attribute to "properties" in all entities with that attribute – resolves 1002.
0.0.11	01/19/11	Gil	Move "format" attribute from the Machine-Volume connection to Volume itself and remove "capacity" attribute from the Machine-Volume connection to resolve 956. Add "properties" attribute to all entities that lacked it as completion of 1002. Add "Read" operation to all entities that lacked it; homogenize the description of the "Read" operation across all entities - 1049
0.0.12	01/26/11	Gil	Further edits to tweak the resolution for 1043.

0.0.13	02/02/11	Gil	Added "Create new Machine Image from existing Machine" scenario as per the decision of the WG on 02/02/2011.
0.0.14	02/09/11	Gil	Added "job_time" property to Provider entity as resolution of 1038. Renamed Machine Template to Machine Configuration and created a new Machine Template entity that reflects the resolution of 1045.
0.0.15	02/21/11	Gil	Added definition of "Template" as resolution of 1063. Changed definitions of Machine Configuration and Machine image as resolution of 1069. Updated UML diagram to remove all relationships that weren't explicitly defined as attributes of the relevant entities.
0.0.16	03/07/11	Gil	Add additional text to description of Template in section 3.2 to resolve 1044. Add column to all entity tables to describe attribute data types to resolve 1073. Changed the 'templates' attribute of Site to 'systemtemplates' to resolve 1075. Changed the description of Machine Configuration to resolve 1079.
0.0.17	03/22/11	Gil	Change attributes on Job entity to resolve 1080. Added "guestInterface" attribute to Machine Configuration, Machine, Volume Template, and Volume to resolve 1083. Changed description of the operations on Machine Configuration to resolve 1084. Moved "hostname" attribute of Machine to a sub-property of the "networkinterface" to resolve 1087. Added "volumes" and "volumetemplates" attributes to Provider entity to resolve 1089. Removed "state" attributes from System Template and Machine Template to resolve 1093.
0.0.18	03/23/11	Gil	Add constraint on relative URIs to heading of Section 4.2, "Attributes", to resolve 1100. Add "volume" sub-attribute to "volumes" attribute of Machine to resolve 1110. Applied consistent camelCased naming to resolve 1111. Added definitions for "immutable/mutable" and "writeable/read-only" to heading of Section 4.2, "Attributes"; revised all mentions of immutable and mutable to agree with these definitions to resolve 1126.
0.0.19	03/30/11	Gil	Add table that defines attributes common to all entities and remove those same attributes from the entity tables to resolve 1094. Add Volume Configuration and Volume Image entities to resolve 1096.
0.0.20	04/06/11	Gil	Add sub-section that describes unit attributes in both base-10 and base-2 (e.g. kilobyte, kibibyte), change memory units to base-2 names (e.g. kibibyte, mebibyte), change unit designators to singular, add reference to IEC 80000-13:2008 – all to resolve 1101.
0.0.21	04/13/11	Doug	Renamed title of section 4.2 per 1153. Expanded the text for the create operation of a System per 999. Added the delete operation to the list of ops for Network per 1011. Converted all pointers to URIs instead of the name of the type its points to per 1129.
0.0.22	04/20/11	Doug	Added section 4.2.1 per 1113 and 1115. Modified the definition of Site.Create per 1132. Modified Site description per 1133. Moved 'bootable' from volume Config to Volume Image per 1137. Removed 'local' from Volume and Volume Config per 1138. Modified definition of Volume per 1139. Removed 'autoDelete' from Volume per 1140.
0.0.23	04/27/11	Gil	Changed description of unit values in Section 4.2.2.1 to resolve

			1157. Change places where "mutable" is meant as "writeable" to use "writeable"; add Update operation to Provider entity to resolve 1158.
0.0.24	05/11/11	Gil	Added Machine Admin entity to resolve 1164. Added text describing the requirements on when a Job entity is created to resolve 1166.
0.0.25	05/25/11	Gil	Added new Section 4.2 describing the facilities for retrieving metadata about the entities, added "Entity Metadata" to UML diagram, added "Type URI" to each entity type - to resolve 1135. Changed description of VolumeConfiguration:format and removed Volume:format to resolve 1136.
0.0.26	06/01/11	Gil	Added volumes and volumeTemplates attributes to the Machine Templates entity to resolved 1155. Changed description of Provider entity to resolve 1174. Chaged the description of the SystemTemplate.Update operation to resolve 1175. Added Section 6 "Security" (plus relevant definitions) to resolve 1178.
0.0.27	06/07/11	Gil	Merged Site and Provider entities into new Cloud Entry Point entity to resolve 1192 and 1196. Added operations and options to Machine and Machine template to resolve 1204. Replaced "Create and deploy a Machine using a Provider created Machine Template" scenario and added "Create a Machine by passing a Machine Template by value", and "Create a Machine using a User created Machine Template" to resolve 1205. Added "Create a Machine Template by specifying individual components" and "Create a Machine Template from a template file" scenarios to resolve 1206. Added "Create new Machine Image from an image file" and "Create new Machine Image from Machine instance" scenarios to resolve 1207.
0.0.28	06/09/2011	Gil	Added 'MeterTemplate' and 'Meter' entities and references to resolve 908. Added Event and EventLog entities and references to resolve 909. Added 'Cancel' operation and 'isCancellable' attribute to the Job entity to resolve 1012. Added terminology definition for "Configuration" to resolve 1191.
0.0.29	06/20/2011	Gil	Removed "Role in Use Cases" rows from entity tables to resolve 1223. Fixed miscellaneous typos and miscapitalizations to resolve 1226.
0.0.30	07/28/2011	Gil, Doug	Merged CM model and HTTP protocol documents. Added definitions of "Consumer" and "Provider" to Section 3, "Terms and Definitions"; made various changes to use these terms consistently throughout the document to resolve 1180. Fix serializations to resolve 1219. Added support for partial updates to resolve 1154. Fixed the CloudEntryPoint to point to the Collection entities to resolve 1238. Tweaked the uri field of EntityMetadata to resolve 1254. Added resolution of 1171 from HTTP protocol doc. Changed Network Template and Network entities, added VSP Template, VSP Configuration, and VSP to resolve 1010, 1085, 1086, and 1088.
0.0.31	08/01/2011	Doug	Removed text that duplicates the HTTP spec to resolve 1193.
0.0.32	08/22/2011	Doug	Added .../xs:any* to show explicit extensibility points to resolve 1271. Added section 4.1.10 (Serialization of Array) and s/*/+/g on array children to resolve 1270.
0.0.33	08/24/2011	Doug	Used CIMI and Cloud Infrastructure Management Interface where

			appropriate. Added section 5.1 which points to the CIMI-CIM and CIMI-RNG docs. Add WIP front-matter.
0.0.34	08/28/2011	Doug	s/The follow describes/The following describes/. Fixed the heading style on the "Entities" section - it lost its style at some point. Added the serialization headers for the EntityMetadata JSON/XML - to match the other entities in the model. s/Entity/EntityMetadata/ in the XML to match the entity type and be more descriptive.
0.0.35	08/31/2011	Gil/Doug	Re-factor ER diagram into separate sub-diagrams and re-organize sections around these diagrams. Changed title of doc and added resolution of issue 1310.

Bibliography

DMTF DSP-IS0102, Distributed Management Task Force, Inc., *Architecture for Managing Clouds White Paper 1.0*, http://dmtf.org/sites/default/files/standards/documents/DSP-IS0102_1.0.0.pdf

DMTF DSP-ISO103, Distributed Management Task Force, Inc., *Use Cases and Interactions for Managing Clouds 1.0.0*, http://www.dmtf.org/sites/default/files/standards/documents/DSP-IS0103_1.0.0.pdf

DMTF DSP-ISXXXX, Distributed Management Task Force, Inc., *Scoping Framework for Cloud Management Models and Protocol Requirements 0.1.5*, http://members.dmtf.org/apps/org/workgroup/cmwg/download.php/56339/Cloud%20Management%20Framework_v015.doc