
**Cloud Infrastructure Management
Interface (CIMI) Model and RESTful
HTTP-based Protocol — An Interface for
Managing Cloud Infrastructure**

*Model d'interface de management de l'infrastructure du nuage
informatique (CIMI) et protocole RESTful basé HTTP — Une interface
pour le management de l'infrastructure du nuage informatique*

Sample Document

get full document from standards.iteh.ai

Sample Document

get full document from standards.iteh.ai



COPYRIGHT PROTECTED DOCUMENT

© ISO/IEC 2015

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized otherwise in any form or by any means, electronic or mechanical, including photocopying, or posting on the internet or an intranet, without prior written permission. Permission can be requested from either ISO at the address below or ISO's member body in the country of the requester.

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Published in Switzerland

CONTENTS

34	Foreword	7
35	1 Scope	9
36	1.1 Document structure	9
37	1.2 Document versioning scheme	9
38	1.3 Typographical conventions	9
39	2 Normative references	10
40	3 Terms and definitions	11
41	4 HTTP-based protocol	14
42	4.1 Introduction	14
43	4.1.1 Protocol evolution and client expectations	14
44	4.1.2 XML namespaces	14
45	4.1.3 URI space	14
46	4.1.4 Media types	15
47	4.1.5 Request headers	15
48	4.1.6 Request query parameters	15
49	4.2 Protocol operations	21
50	4.2.1 Common CRUD operations	22
51	4.3 OVF support	29
52	5 Model	30
53	5.1 Resource wrappers	30
54	5.2 Extensibility	31
55	5.3 Identifiers	31
56	5.4 Attribute constraints	32
57	5.5 Data types and their serialization	33
58	5.5.1 boolean	33
59	5.5.2 dateTime	33
60	5.5.3 duration	33
61	5.5.4 integer	34
62	5.5.5 string	34
63	5.5.6 ref	34
64	5.5.7 map	35
65	5.5.8 structure	35
66	5.5.9 byte[]	36
67	5.5.10 URI	36
68	5.5.11 Arrays	36
69	5.5.12 Collections	37
70	5.5.13 "Any" type	41
71	5.5.14 Empty attribute values	41
72	5.6 Units	41
73	5.7 Relationship semantics	42
74	5.8 Operations	42
75	5.9 Alternative model formats	43
76	5.10 Resources	43
77	5.10.1 Common attributes	43
78	5.11 Resource metadata	45
79	5.11.1 Serialization of attribute value constraints	49
80	5.11.2 Capabilities	51
81	5.11.3 ResourceMetadataCollection Resource	54
82	5.12 Cloud Entry Point	55
83	5.12.1 Operations	61
84	5.13 System Resources and relationships	61
85	5.13.1 System	62

86	5.13.2	SystemCollection Resource	80
87	5.13.3	SystemTemplate Resource	81
88	5.13.4	SystemTemplateCollection Resource	87
89	5.14	Machine Resources and relationships	88
90	5.14.1	Machine	89
91	5.14.2	MachineCollection	107
92	5.14.3	MachineTemplate	109
93	5.14.4	MachineTemplateCollection Resource	116
94	5.14.5	MachineConfiguration Resource	117
95	5.14.6	MachineConfigurationCollection Resource	119
96	5.14.7	MachineImage Resource	120
97	5.14.8	MachineImageCollection Resource	124
98	5.14.9	Credential Resource	125
99	5.14.10	CredentialCollection Resource	126
100	5.14.11	CredentialTemplate Resource	127
101	5.14.12	CredentialTemplateCollection Resource	128
102	5.15	Volume Resources and relationships	130
103	5.15.1	Volume	131
104	5.15.2	VolumeCollection Resource	135
105	5.15.3	VolumeTemplate Resource	136
106	5.15.4	VolumeTemplateCollection Resource	138
107	5.15.5	VolumeConfiguration Resource	139
108	5.15.6	VolumeConfigurationCollection Resource	141
109	5.15.7	VolumeImage Resource	142
110	5.15.8	VolumeImageCollection Resource	144
111	5.16	Network Resources and relationships	145
112	5.16.1	Network	145
113	5.16.2	NetworkCollection Resource	153
114	5.16.3	NetworkTemplate Resource	154
115	5.16.4	NetworkTemplateCollection Resource	156
116	5.16.5	NetworkConfiguration Resource	157
117	5.16.6	NetworkConfigurationCollection Resource	158
118	5.16.7	NetworkPort	160
119	5.16.8	NetworkPortCollection Resource	164
120	5.16.9	NetworkPortTemplate Resource	165
121	5.16.10	NetworkPortTemplateCollection Resource	168
122	5.16.11	NetworkPortConfiguration Resource	169
123	5.16.12	NetworkPortConfigurationCollection Resource	170
124	5.16.13	Address Resource	171
125	5.16.14	AddressCollection Resource	173
126	5.16.15	AddressTemplate Resource	174
127	5.16.16	AddressTemplateCollection Resource	176
128	5.16.17	ForwardingGroup Resource	177
129	5.16.18	ForwardingGroupCollection Resource	180
130	5.16.19	ForwardingGroupTemplate Resource	181
131	5.16.20	ForwardingGroupTemplateCollection Resource	182
132	5.17	Monitoring Resources and relationships	183
133	5.17.1	Job Resource	184
134	5.17.2	JobCollection Resource	188
135	5.17.3	Meter Resource	189
136	5.17.4	MeterCollection Resource	195
137	5.17.5	MeterTemplate Resource	196
138	5.17.6	MeterTemplateCollection Resource	197
139	5.17.7	MeterConfiguration Resource	198
140	5.17.8	MeterConfigurationCollection Resource	201
141	5.17.9	EventLog Resource	202

142	5.17.10 EventLogCollection Resource	205
143	5.17.11 EventLogTemplate Resource	206
144	5.17.12 EventLogTemplateCollection Resource	207
145	5.17.13 Event Resource	208
146	6 Security considerations	216
147	ANNEX A (normative) OVF support in CIMI	217
148	ANNEX B (informative) XML Schema.....	219
149	ANNEX C (informative) Change log.....	220
150		

151 Figures

152	Figure 1 - Cloud Entry Point.....	56
153	Figure 2 - System Resources.....	62
154	Figure 3 - Machine Resources	89
155	Figure 4 - Volume Resources	130
156	Figure 5 - Network Resources	145
157	Figure 6 - Monitoring Resources.....	184
158		

159 Tables

160	Table 1 – XML namespaces	14
161	Table 2 – Named structure.....	35
162	Table 3 – Converting a relative URI to an absolute URI	36
163	Table 4 – Numerical equivalents for attributes.....	42
164	Table 5 – Common attributes.....	43
165	Table 7 – Capability URIs	51
166	Table 8 – CloudEntryPoint attributes	56
167	Table 9 – System attributes	63
168	Table 10 – SystemSystem attributes	67
169	Table 11 – SystemMachine attributes.....	68
170	Table 12 – SystemCredential attributes.....	70
171	Table 13 – SystemVolume attributes	71
172	Table 14 – SystemNetwork attributes	72
173	Table 15 – SystemNetworkPort attributes	74
174	Table 16 – SystemAddress attributes	75
175	Table 17 – SystemForwardingGroup attributes	76
176	Table 18 – SystemTemplate attributes	82
177	Table 19 – Machine attributes.....	89
178	Table 20 – Disk attributes	93
179	Table 21 – MachineVolume attributes	94
180	Table 22 – MachineNetworkInterface attributes	96
181	Table 23 – MachineNetworkInterfaceAddress attributes	98
182	Table 24 – MachineSnapshot attributes	100
183	Table 25 – MachineTemplate attributes.....	109
184	Table 26 – MachineConfiguration attributes	117
185	Table 27 – MachineImage attributes.....	120

ISO/IEC 19831:2015 (E)

186	Table 28 – Credential attributes	125
187	Table 29 – UserName/Password attributes	125
188	Table 30 – Public key attributes	125
189	Table 31 – CredentialTemplate attributes	127
190	Table 32 – Volume attributes	131
191	Table 33 – VolumeVolumeImage attributes	133
192	Table 34 – VolumeTemplate attributes	136
193	Table 35 – VolumeConfiguration attributes	140
194	Table 36 – VolumeImage attributes	142
195	Table 37 – Network attributes	145
196	Table 38 – NetworkTemplate attributes	154
197	Table 39 – NetworkConfiguration attributes	157
198	Table 40 – NetworkPort attributes	160
199	Table 41 – NetworkPortTemplate attributes	165
200	Table 42 – NetworkPortConfiguration attributes	169
201	Table 43 – Address attributes	171
202	Table 44 – AddressTemplate attributes	174
203	Table 45 – ForwardingGroup attributes	178
204	Table 46 – ForwardingGroupNetwork attributes	179
205	Table 47 – ForwardingGroupTemplate attributes	181
206	Table 48 – Job attributes	185
207	Table 49 – Meter attributes	189
208	Table 50 – Sample attributes	192
209	Table 51 – MeterTemplate attributes	196
210	Table 52 – MeterConfiguration attributes	198
211	Table 53 – aspect URIs	200
212	Table 54 – EventLog attributes	202
213	Table 55 – EventLogTemplate attributes	206
214	Table 56 – Event attributes	208
215	Table 57 – type URIs	211
216		
217		

218

Foreword

219 The *Cloud Infrastructure Management Interface (CIMI) Model and RESTful HTTP-based Protocol*
 220 specification (DSP0263) was prepared by the DMTF Cloud Management Working Group. It defines a
 221 logical model for the management of resources within the Infrastructure as a Service domain.

222 DMTF is a not-for-profit association of industry members dedicated to promoting enterprise and systems
 223 management and interoperability.

224 Acknowledgments

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

226 Editors (past and present):

- 227 • Marios Andreou – Red Hat
- 228 • Doug Davis – IBM
- 229 • Jacques Durand – Fujitsu
- 230 • Gilbert Pilz – Oracle

231 Contributors:

- 232 • Ghazanfar Ali – ZTE Corporation
- 233 • Marios Andreou – Red Hat
- 234 • Keith Bankston – Microsoft Corporation
- 235 • Winston Bumpus – VMware Inc.
- 236 • Nathan Burkhart – Microsoft Corporation
- 237 • Mark Carlson – Oracle
- 238 • Steve Carter – Novell
- 239 • Junsheng Chu – ZTE Corporation
- 240 • Josh Cohen – Microsoft Corporation
- 241 • Derek Coleman – Hewlett-Packard Company
- 242 • John Crandall – Brocade Communications Systems
- 243 • Doug Davis – IBM
- 244 • Jim Davis – WBEM Solutions
- 245 • Fernando de la Iglesia – Telefónica
- 246 • Hiroshi Dempo – NEC Corporation
- 247 • Jacques Durand – Fujitsu
- 248 • Yigal Edery – Microsoft Corporation
- 249 • George Ericson – EMC
- 250 • Colleen Evans – Microsoft Corporation
- 251 • Norbert Floeren – Ericsson AB
- 252 • Robert Freund – Hitachi, Ltd.
- 253 • Fermín Galán – Telefónica
- 254 • Krishnan Gopalan – Microsoft Corporation
- 255 • Kazunori Iwasa – Fujitsu
- 256 • Mark Johnson – IBM
- 257 • Bhumip Khasnabish – ZTE Corporation
- 258 • Dies Köper – Fujitsu
- 259 • Vincent Kowalski – BMC Software
- 260 • Ruby Krishnaswamy – France Telecom Group
- 261 • Lawrence Lamers – VMware Inc.
- 262 • Paul Lipton – CA Technologies
- 263 • James Livingston – NEC Corporation
- 264 • Vince Lubsey – Virtustream Inc.

ISO/IEC 19831:2015 (E)

- 265 • David Lutterkort – Red Hat
- 266 • Fred Maciel – Hitachi, Ltd.
- 267 • Andreas Maier – IBM
- 268 • Ashok Malhotra – Oracle
- 269 • Jeff Mischkinisky – Oracle
- 270 • Jesus Molina – Fujitsu
- 271 • Efraim Moscovich – CA Technologies
- 272 • Bryan Murray – Hewlett-Packard Company
- 273 • Steven Neely – Cisco
- 274 • Ryuichi Ogawa – NEC Corporation
- 275 • John Parchem – Microsoft Corporation
- 276 • Shishir Pardikar – Citrix Systems Inc.
- 277 • Miguel Peñalvo – Telefónica
- 278 • Gilbert Pilz – Oracle
- 279 • Alvaro Polo – Telefónica
- 280 • Enrico Ronco – Telecom Italia
- 281 • Federico Rossini – Telecom Italia
- 282 • Matthew Rutkowski – IBM
- 283 • Tom Rutt – Fujitsu
- 284 • Hemal Shah – Broadcom
- 285 • Nihar Shah – Microsoft Corporation
- 286 • Alan Sill – Texas Tech University
- 287 • Zhexuan Song – Huawei
- 288 • Marvin Waschke – CA Technologies
- 289 • Eric Wells – Hitachi, Ltd.
- 290 • Jeff Wheeler – Huawei
- 291 • Maarten Wiggers – Fujitsu
- 292 • Daniel Wilson – Ericsson AB
- 293 • Steve Winkler – SAP AG
- 294 • Jack Yu – Oracle
- 295 • Aaron Zhang – Huawei
- 296 • HengLiang Zhang – Huawei
- 297

298 Cloud Infrastructure Management Interface (CIMI) Model and 299 RESTful HTTP-based Protocol

300 1 Scope

301 This specification describes the model and protocol for management interactions between a cloud
302 Infrastructure as a Service (IaaS) Provider and the Consumers of an IaaS service. The basic resources of
303 IaaS (machines, storage, and networks) are modeled with the goal of providing Consumer management
304 access to an implementation of IaaS and facilitating portability between cloud implementations that
305 support the specification. This document specifies a Representational State Transfer (REST)-style
306 protocol using HTTP. However, the underlying model is not specific to HTTP, and it is possible to map it
307 to other protocols as well.

308 CIMI addresses the management of the lifecycle of infrastructure provided by a Provider. CIMI does not
309 extend beyond infrastructure management to the control of the applications and services that the
310 Consumer chooses to run on the infrastructure provided as a service by the Provider. Although CIMI may
311 be to some extent applicable to other cloud service models, such as Platform as a Service ("PaaS") or
312 Storage as a Service ("SaaS"), these uses are outside the design goals of CIMI.

313 1.1 Document structure

314 This document defines a model and a RESTful HTTP-based protocol.

315 The core REST patterns are defined first and, after each resource is defined, any HTTP-specific
316 information for that resource is specified.

317 1.2 Document versioning scheme

318 This document adheres to the versioning scheme defined in clause 6.3 of [DSP4004](#).

319 As the specification changes over time certain features might be deprecated. These are identified in the
320 specification and should not be supported. Each of these deprecated features is clearly denoted in the
321 clause in which they were previously defined.

322 1.3 Typographical conventions

323 This specification uses the following conventions:

324 In the narrative text of the specification:

- 325 • The regular or narrative font is Arial.
- 326 • Proper CIMI nouns such as Resource names, attribute names, operation names, reserved
327 variable names are in *Courier* font. (e.g. *Machine*, *volumes*, *\$expand*). The plural form
328 applies to such names to indicate several instances of such Resources (e.g. *Machines*,
329 *Systems*).
- 330 • Examples text are in small *Courier* font and over a darker background.
- 331 • Quotes are used for any text that needs be distinguished as name or value of a particular
332 concept (e.g. the "value constraints" attribute, the "Resource Name" column, a "false" value). In
333 such cases, the string in quotes is always qualified by the concept it is an instance of.
- 334 • Names for CIMI concepts that may be common English words but have a very specific meaning
335 in CIMI, are in narrative font but capitalized, e.g. *Provider*, *Consumer*, *Resource*, *Collection*.

ISO/IEC 19831:2015 (E)

336 When used in their common English sense they remain lower-case. However, CIMI modeling
337 concepts that are used in a commonly understood manner remain in lower-case, such as:
338 attribute, operation.

339 Inside tables describing the Resource data model:

- 340 • The narrative font is used for all terms, as the table structure qualifies them sufficiently.
- 341 • Where textual descriptions are introduced, the rules for narrative text apply.
- 342 • If a name is used as types (i.e., names of embedded structures as well as atomic types such as
343 "integer", "string"), are in *italic*.
- 344 • Names that are just placeholders for actual names that may vary with each model instance, are
345 between < > (e.g., <componentTemplate>).

346 Where the serialization of Resources is described, a pseudo-schema notation is used with the following
347 conventions:

- 348 • Values in *italics* indicate data types instead of literal values.
- 349 • Characters are appended to items to indicate cardinality:
 - 350 – "?" (0 or 1)
 - 351 – "*" (0 or more)
 - 352 – "+" (1 or more)
- 353 • Vertical bars, "|", denote choice. For example, "a|b" means a choice between "a" and "b".
- 354 • Parentheses, "(" and ")", are used to indicate the scope of the operators "?", "*", "+" and "|".
- 355 • Ellipses (i.e., "...") indicate points of extensibility. Note that the lack of an ellipses does not mean
356 no extensibility point exists, rather it is just not explicitly called out - usually for the sake of
357 brevity.

358 Operation names Create, Update, Delete, Read are abstract operations that convey the semantics of
359 concrete corresponding operations, such as HTTP methods or CIMI operation URIs.

360 2 Normative references

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

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

367 DMTF DSP0243, *Open Virtualization Format Specification 1.1*,
368 http://www.dmtf.org/sites/default/files/standards/documents/DSP0243_1.1.pdf

369 DMTF DSP1001, *Management Profile Specification Usage Guide 1.1*,
370 http://www.dmtf.org/standards/published_documents/DSP1001_1.1.pdf

371 DMTF DSP4004, *DMTF Release Process 2.4*,
372 http://www.dmtf.org/sites/default/files/standards/documents/DSP4004_2.4.pdf

373 IANA HTTP Header Registry, <http://www.iana.org/assignments/message-headers/perm-headers.html>

- 374 IEC 80000-13:2008, International Organization for Standardization, Geneva, Switzerland, *Quantities and*
 375 *units – Part 13: Information science and technology*, April 2008,
 376 http://www.iso.org/iso/catalogue_detail?csnumber=31898
- 377 IETF RFC2616, R. Fielding et al, *Hypertext Transfer Protocol -- HTTP/1.1*,
 378 <http://www.ietf.org/rfc/rfc2616.txt>
- 379 IETF RFC3986, T. Berners-Lee et al, *Uniform Resource Identifiers (URI): Generic Syntax*, August 1998,
 380 <http://www.ietf.org/rfc/rfc3986.txt>
- 381 IETF RFC4627, D. Crockford, *The application/json Media Type for JavaScript Object Notation (JSON)*,
 382 July 2006, <http://www.ietf.org/rfc/rfc4627.txt>
- 383 IETF RFC5246, T. Dierks and E. Rescorla, *The Transport Layer Security (TLS) Protocol Version 1.2*,
 384 <http://www.ietf.org/rfc/rfc5246.txt>
- 385 ISO 8601:2004, International Organization for Standardization, Geneva, Switzerland, *Data elements and*
 386 *interchange formats -- Information interchange - - Representation of dates and times*, March 2008,
 387 http://www.iso.org/iso/catalogue_detail.htm?csnumber=40874
- 388 ISO/IEC 14977:1996, Roger S. Scowen, *Extended BNF — A generic base standard*. Software
 389 Engineering Standards Symposium 1993.
 390 http://www.iso.org/iso/catalogue_detail?csnumber=26153
- 391 ISO/IEC Directives, Part 2, *Rules for the structure and drafting of International Standards*,
 392 <http://isotc.iso.org/livelink/livelink.exe?func=ll&objId=4230456&objAction=browse&sort=subtype>
- 393 NIST Special Publication 800-145, Peter Mell and Timothy Grance, *The NIST Definition of Cloud*
 394 *Computing*, Sept. 2011, <http://csrc.nist.gov/publications/nistpubs/800-145/SP800-145.pdf>
- 395 NIST Special Publication 500-292, Fang Liu, Jin Tong, Jian Mao, Robert Bohn, John Messina, Lee
 396 Badger and Dawn Leaf, *NIST Cloud Computing Reference Architecture*, Sept. 2011,
 397 [http://collaborate.nist.gov/wiki-cloud-](http://collaborate.nist.gov/wiki-cloud-computing/pub/CloudComputing/ReferenceArchitectureTaxonomy/NIST_SP_500-292_-_090611.pdf)
 398 [computing/pub/CloudComputing/ReferenceArchitectureTaxonomy/NIST_SP_500-292_-_090611.pdf](http://collaborate.nist.gov/wiki-cloud-computing/pub/CloudComputing/ReferenceArchitectureTaxonomy/NIST_SP_500-292_-_090611.pdf)
- 399 Representational State Transfer, Roy Fielding, Doctoral dissertation, University of California, *Architectural*
 400 *Styles and the Design of Network-based Software Architectures (Chapter 5)*, 2000,
 401 http://www.ics.uci.edu/~fielding/pubs/dissertation/rest_arch_style.htm
- 402 XMLSchema - Part 1, World Wide Web Consortium (W3C) Recommendation, H. Thompson, et al.,
 403 Editors, *XML Schema Part 1: Structures Second Edition*, 28 October 2004,
 404 <http://www.w3.org/TR/xmlschema-1/>
- 405 XMLSchema - Part 2, World Wide Web Consortium (W3C) Recommendation, P. Biron, A. Malhotra,
 406 Editors, *XML Schema Part 2: Datatypes (Second Edition)*, 28 October 2004,
 407 <http://www.w3.org/TR/xmlschema-2/>

408 3 Terms and definitions

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

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

ISO/IEC 19831:2015 (E)

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

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

422 The terms defined in [DSP4004](#), [DSP0223](#), and [DSP1001](#) apply to this document. The following additional
423 terms are used in this document.

424 **3.1** 425 **authentication**

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

429 **3.2** 430 **authorization**

431 The process of verifying that an authenticated principal (person, service, etc.) has permission to perform
432 certain operations (e.g., read, update) on specific Resources. (Also known as Access Control.)

433 **3.3** 434 **cloud**

435 Synonymous with "cloud computing" as defined in section 2 of the NIST Definition of Cloud Computing
436 [\[SP800-145\]](#).

437 **3.4** 438 **Cloud Service Consumer**

439 A category of actors that includes the Consumer Business Manager (who approves business and
440 financial expenditures for consumed services; accounts for used service instances; establishes business
441 relationships; sets up accounts, budget, and terms; etc.); the Consumer Service Administrator (who
442 requests service instances and changes to service instances; purchases services within the business
443 relationship; creates Service Users (including policies); allocates resources, such as computer and
444 storage; generates reports, such as usage; etc.); and Service Users (who use service instances provided
445 by a Cloud Service Provider). The term "Consumer" is used if the indicated action or activity could involve
446 one or more of the above actors. In cases where the distinction between the actors in this category is
447 relevant, the more detailed term is used.

448 For purposes of comparison and alignment, it should be noted that a Cloud Service Consumer is
449 equivalent to the "Cloud Consumer" actor defined in the NIST Reference Architecture [\[SP500-292\]](#).

450 **3.5** 451 **Cloud Service Provider**

452 A category of actors that includes the Service Operations Manager (who manages the technical
453 infrastructure required for providing cloud services; monitors and measures performance and utilization
454 against SLAs; provides reports from monitoring and measurement; etc.); Service Business Manager (who
455 offers all types of services developed by cloud service developers; accounts for services potentially
456 offered by service Providers themselves and services offered on behalf of cloud service developers;
457 establishes a portfolio of business relationships; and sets up accounts and terms for Consumers, etc.);
458 and Service Transition Manager (who enables a customer to use the cloud service, including
459 "onboarding", integration, and process adoption; defines and creates service offerings based on
460 Templates and Configurations that can be used by Consumers and are populated into the catalog; etc.).
461 The term "Provider" is used if the indicated action or activity could involve one or more of the above
462 actors. In cases where the distinction between the actors in the category is relevant, the more detailed
463 term is used.

464 For purposes of comparison and alignment, it should be noted that a Cloud Service Provider is equivalent
465 to the “Cloud Provider” actor defined in the NIST Reference Architecture [[SP500-292](#)].

466 **3.6**
467 **Collection**

468 A particular kind of Resource that contains a collection of other Resources and has a representation and
469 serialization defined in this specification. Synonym for “CIMI collection”.

470 **3.7**
471 **Configuration**

472 A set of metadata, the values of which serve as the parameters of a discrete conformation of a specific
473 type of virtual resource.

474 **3.8**
475 **Infrastructure as a Service (IaaS)**

476 A cloud computing service model defined in section 2 of the NIST Definition of Cloud Computing [[SP800-145](#)].
477

478 **3.9**
479 **message confidentiality**

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

481 **3.10**
482 **message integrity**

483 A quality of a message that allows a receiver of that message to determine whether the contents of the
484 message have been altered since its creation.

485 **3.11**
486 **Resource**

487 A representation of an entity managed by the [Cloud Service] Provider that is generally available to the
488 [Cloud Service] Consumer to access or operate on by the way of the interface described in this
489 specification. Synonym for “CIMI resource”.

490 **3.12**
491 **Template**

492 Synonym for “CIMI template”. A Resource that represents the set of metadata and instructions used to
493 instantiate some other Resource (e.g., a `MachineTemplate` is used to create `Machines`). Templates
494 may aggregate other metadata Resources such as other Templates, Configurations, and Images. For
495 example, a `MachineTemplate` refers to a `MachineConfiguration` and a `MachineImage`.

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

- 498 1. By reference - allow Consumers to reference a Template (that exists as a Resource in the
499 Provider) as part of the instantiation operation.
- 500 2. By value - allow Consumers to dynamically provide the Template information as part of the
501 instantiation operation.
- 502 3. Reference with overrides - allow Consumers to reference a Template (that exists as a Resource
503 in the Provider) and provide additional values that override the attributes of that Template as part
504 of the instantiation operation.

505 4 HTTP-based protocol

506 4.1 Introduction

507 All operations are based on the *HyperText Transfer Protocol (HTTP)*, version 1.1 [RFC2616]. Each
508 request is sent by using an HTTP verb such as PUT, GET, DELETE, HEAD, or POST and includes a
509 message body in either JSON or XML format. Each response uses a standard HTTP status code, whose
510 semantics are interpreted in the context of the particular request that was made. Each Resource in the
511 model has a MIME type that further contextualizes the payload of requests and responses.

512 Resources in the model are identified by URIs, and each Resource's representation shall contain an "ID"
513 attribute, of type URI, that acts as a "self pointer." This URI shall be unique within the context of the
514 Provider's implementation. Dereferencing (through an HTTP GET) the URI of a Resource yields a
515 representation of the Resource containing attributes and links to associated Resources. To begin
516 operations, a client shall know the URI to the main entry point of a Provider - also known as the "Cloud
517 Entry Point" Resource. All other Resources within the environment shall then be discoverable by the way
518 of the iterative following of links to associated Resources within each Resource retrieved.

519 4.1.1 Protocol evolution and client expectations

520 Future versions of this specification structure changes in such a way that clients that conform to an earlier
521 version of this specification continue to work, and are not be adversely affected by the evolution of the
522 protocol. Clients are expected to follow a few simple rules to ensure this compatibility:

- 523 1. Clients shall not assume that the serializations shown for responses in this specification are
524 complete. In particular, clients shall accept responses that contain data mixed in with the
525 serializations shown here, and shall ignore such data. However, per clause 4.2.1.3, clients shall
526 include unknown data in PUT requests to update Resources.
- 527 2. Clients shall not assume anything about the operations supported by a server. They are expected
528 to discover operations that are supported (and permissible) by navigating to Resources from the
529 cloud entry point. The serializations of Resources encountered indicate which operations are
530 supported by the server.

531 4.1.2 XML namespaces

532 Table 1 lists the XML namespaces that are used in this specification. The choice of any namespace prefix
533 is arbitrary and not semantically significant.

534

Table 1 – XML namespaces

Prefix	XML Namespaces	Specification
cimi	http://schemas.dmtf.org/cimi/1	This specification
xs	http://www.w3.org/2001/XMLSchema	XML Schema Part2

535 4.1.3 URI space

536 While URIs returned by Providers are to be treated as opaque by Consumers, and Consumers shall not
537 make assumptions about the layout of the URIs or the structures of the URIs for the Resources, a
538 Consumer may augment URIs with any well-defined query parameters that are supported by the Provider
539 as defined in clause 4.1.6.

540 The sample URIs used in this specification are not normative and the patterns used shall not be
541 interpreted as guidance for implementations. For example, any of the following URIs might be used by
542 Providers to reference a particular Machine Resource:

```

543 http://example.com/machines/12345
544 http://example.com/machines?id=12345
545 http://example.com/12345
546 http://example.com/Cloud/resource?id=12345

```

547 4.1.4 Media types

548 In this specification, Resource and response representations are encoded either in JSON, as specified in
549 [RFC4627](#) or in XML. If serialized in JSON, the media-type for CIMI resources shall be "application/json".
550 If serialized in XML, the media-type shall be "application/xml".

551 In the JSON serialization of CIMI representations sent by Providers, there shall be an additional attribute
552 on the root object called "resourceURI" that contains the unique URI that is associated with the type of
553 CIMI resource being serialized.

554 Note that this requirement applies even if the `$select` attribute is used to subset the Resource being
555 acted upon.

556 In the XML serialization of Collection representations sent by Providers there shall be a `resourceURI`
557 attribute, as shown in the example XML serialization of Collections in clause 5.5.12.

558 This attribute is optional for Consumers to include. If included, this attribute's value shall match the
559 "typeURI" attribute of the corresponding `ResourceMetadata` Resource (see clause 5.11), if
560 `ResourceMetadata` is supported. This value shall also be equivalent to the wrapping element of the
561 XML serialization; in other words, the namespace of the wrapper element concatenated a "/" and then its
562 localName.

563 Any CIMI resource implemented by a Provider shall have representations in JSON and XML. The client
564 implementation may thus use either JSON or XML in requests with any server implementation, and may
565 request a specific serialization using server-driven content negotiation (using the Accept request header).

566 4.1.5 Request headers

567 This specification uses general-header, request-header, and entity-header headers as defined in
568 [RFC2616](#) in request messages to provide metadata about the message. Applications using messages
569 defined in this specification shall use headers consistent with the requirements of [RFC2616](#).

570 4.1.6 Request query parameters

571 Providers may choose to include query parameters as part of the URIs returned to Consumers.
572 Consumers shall include those query parameters when sending messages to those URIs. If Providers
573 choose to define query parameters care should be taken to avoid conflicts with CIMI defined query
574 parameters.

575 To modify the behavior of the Provider when processing request messages, Consumers may augment
576 request URIs as described in the following clauses. As stated in clause 4.1.3, URIs returned from
577 Providers are to be treated as opaque by Consumers; however, it is the responsibility of the Consumer to
578 understand the use of the query parameters defined in the following clauses and ensure correctness
579 when making a request.

580 Unsupported, or unknown, query parameters shall be silently ignored by Providers. Consumers may
581 examine the `CloudEntryPoint`'s capabilities to determine whether support of these query parameters is
582 enabled.