



1  
2  
3  
4  
5  
  
6  
7  
  
8  
9

**Document Number: DSP0264**

**Version: 1.0.0c**

**Date: 2012-09-27**

# **Cloud Infrastructure Management Interface - Common Information Model (CIMI-CIM)**

## **A CIM Representation of the CIMI model**

### **Information for Work-in-Progress version:**

**IMPORTANT:** This document 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, it may still change, perhaps profoundly. This document is available for public review and comment until the stated expiration date.

**It expires on: 2013-02-01**

**Provide any comments through the DMTF Feedback Portal:**

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

**Document Type: Specification**

**Document Status: Work In Progress**

**Document Language: US-EN**

---

10 Copyright Notice  
 11 Copyright © 2012 Distributed Management Task Force, Inc. (DMTF). All rights reserved.

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 26 withdrawn or modified after publication, and shall be indemnified and held harmless by any party  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

33 **Contents**

34 1 Scope ..... 6  
 35 2 References ..... 6  
 36 3 Terms and Definitions ..... 6  
 37 4 CIMI CIM Translation ..... 6  
 38 4.1 CIM Formal Model ..... 7  
 39 4.2 Translation Rules ..... 7  
 40 4.2.1 Common Resource Attributes ..... 7  
 41 4.2.2 Resource Metadata ..... 7  
 42 4.2.3 Resource Translation Rules ..... 7  
 43 5 CIMI CIM MOF Representation Examples ..... 12  
 44 5.1 Ordinary Class ..... 12  
 45 5.1.1 CIMI\_BaseElement ..... 12  
 46 5.1.2 CIMI\_Machine ..... 13  
 47 5.1.3 CIMI\_Disk ..... 15  
 48 5.2 Association ..... 15  
 49 5.2.1 CIMI\_MachineEventLog ..... 15  
 50 5.2.2 CIMI\_MachineLatestSnapshot ..... 16  
 51 5.3 Structure ..... 16  
 52 5.3.1 CIMI\_MachineTemplateVolumes ..... 16

53

54

## FIGURES

55 **No table of figures entries found.**

56

## Foreword

57 This document is a deliverable from the DMTF Cloud Management Working Group. It defines a CIM  
58 representation for the Cloud Infrastructure Management Interface [CIMI] logical model. See the CIMI  
59 specification [CIMI] for more information. This document assumes that the reader is familiar with the  
60 concepts defined in the CIM Infrastructure Specification 2.6 (DSP0004).

## 61 Acknowledgments

62 The authors wish to acknowledge the following people.

### 63 Editors:

- 64 • Bankston, J. Keith – Microsoft Corporation
- 65 • Burkhart, Nathan - Microsoft Corporation
- 66 • Cohen, Josh - Microsoft Corporation
- 67 • Ericson, George – EMC
- 68 • Jim Davis – WS, Inc.

### 69 Contributors:

- 70 • Ali, Ghazanfar - ZTE Corporation
- 71 • Andreou, Marios - Red Hat
- 72 • Bankston, J. Keith – Microsoft Corporation
- 73 • Bumpus, Winston - VMware Inc.
- 74 • Burkhart, Nathan - Microsoft Corporation
- 75 • Carlson, Mark - Oracle
- 76 • Carter, Steve - Novell
- 77 • Chu, Junsheng - ZTE Corporation
- 78 • Cohen, Josh - Microsoft Corporation
- 79 • Coleman, Derek - Hewlett-Packard Company
- 80 • Crandall, John - Brocade Communications Systems
- 81 • Davis, Doug - IBM
- 82 • Davis, Jim - WBEM Solutions
- 83 • de la Iglesia, Fernando - Telefónica
- 84 • Dempo, Hiroshi - NEC Corporation
- 85 • Durand, Jacques - Fujitsu
- 86 • Edery, Yigal - Microsoft Corporation
- 87 • Ericson, George - EMC
- 88 • Evans, Colleen - Microsoft Corporation
- 89 • Floeren, Norbert - Ericsson AB
- 90 • Freund, Robert - Hitachi, Ltd.
- 91 • Galán, Fermín - Telefónica
- 92 • Gopalan, Krishnan - Microsoft Corporation
- 93 • Iwasa, Kazunori - Fujitsu
- 94 • Johnson, Mark - IBM
- 95 • Khasnabish, Bhumi - ZTE Corporation
- 96 • Kowalski, Vincent - BMC Software
- 97 • Krishnaswamy, Ruby - France Telecom Group
- 98 • Lamers, Lawrence - VMware Inc.
- 99 • Lipton, Paul - CA Technologies
- 100 • Livingston, James - NEC Corporation
- 101 • Lubsey, Vince - Virtustream Inc.

- 102 • Lutterkort, David - Red Hat
- 103 • Maciel, Fred - Hitachi, Ltd.
- 104 • Maier, Andreas - IBM
- 105 • Malhotra, Ashok - Oracle
- 106 • Mischkinsky, Jeff - Oracle
- 107 • Molina, Jesus - Fujitsu
- 108 • Moscovich, Efraim - CA Technologies
- 109 • Murray, Bryan - Hewlett-Packard Company
- 110 • Neely, Steven – Cisco
- 111 • Ogawa, Ryuichi - NEC Corporation
- 112 • Parchem, John - Microsoft Corporation
- 113 • Pardikar, Shishir - Citrix Systems Inc.
- 114 • Peñalvo, Miguel - Telefónica
- 115 • Pilz, Gilbert - Oracle
- 116 • Polo, Alvaro - Telefónica
- 117 • Ronco, Enrico - Telecom Italia
- 118 • Rossini, Federico - Telecom Italia
- 119 • Rutkowski, Matthew - IBM
- 120 • Rutt, Tom - Fujitsu
- 121 • Shah, Hemal - Broadcom
- 122 • Shah, Nihar - Microsoft Corporation
- 123 • Sill, Alan - Open Grid Forum
- 124 • Song, Zhexuan - Huawei
- 125 • Waschke, Marvin - CA Technologies
- 126 • Wells, Eric - Hitachi, Ltd.
- 127 • Wheeler, Jeff - Huawei
- 128 • Wiggers, Maarten - Fujitsu
- 129 • Winkler, Steve - SAP AG
- 130 • Yu, Jack - Oracle
- 131 • Zhang, Aaron - Huawei
- 132 • Zhang, HengLiang - Huawei
- 133

## 134 **1 Scope**

135 This document makes use of the common meta-model used by CIM, the Common Information Model to  
136 describe the CIMI logical model. This is defined in DSP004, **CIM Infrastructure Specification 2.7**

## 137 **2 References**

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

141 DMTF DSP0263, Cloud Infrastructure Management Interface (CIMI) Model and REST Interface over  
142 HTTP, An Interface for Managing Cloud Infrastructure version 1.0.0,  
143 [http://dmtof.org/sites/default/files/standards/documents/DSP0263\\_1.0.0.pdf](http://dmtof.org/sites/default/files/standards/documents/DSP0263_1.0.0.pdf)

144 DMTF DSP0004, Common Information Model (CIM) Infrastructure version 2.7,  
145 [http://dmtof.org/sites/default/files/standards/documents/DSP0004\\_2.7.0.pdf](http://dmtof.org/sites/default/files/standards/documents/DSP0004_2.7.0.pdf)

## 146 **3 Terms and Definitions**

### 147 **3.1 CIM (Common Information Model)**

148 CIM (Common Information Model) defined by DSP0004 as:

- 149 1. The name of the meta-model used to define schemas (e.g., the CIM schema or extension  
150 schemas).
- 151 2. The name of the schema published by the DMTF (i.e., the CIM schema).

152 This specification describes the translation between the CIM meta-model and CIMI Resources.

### 153 **3.2 CIM Schema**

154 The schema published by the DMTF that defines the Common Information Model. It is divided into a core  
155 model and a common model. Extension schemas are defined outside of the DMTF and are not  
156 considered part of the CIM schema.

### 157 **3.3 MOF (Managed Object Format)**

158 A DMTF defined language used to create CIM metamodel conformant representations of model elements.  
159 The Managed Object Format (MOF) is an Interface Definition Language (IDL) based on ISO/IEC  
160 14750:1999. CIM Infrastructure specification ( DSP0004) ANNEX A provides a complete description of  
161 the MOF language.

### 162 **3.4 Ordinary class**

163 A class that is neither an association class nor an indication class.

## 164 **4 CIMI CIM Translation**

165 Transformation of the CIMI CIM into CIM metamodel conformant representations enables access of the  
166 services defined by CIMI in CIM based environments. Such environments encompass a broad range of  
167 supported operating systems, languages, platforms, protocols, and other technologies.

168 This specification describes transformations in a manner that enables any CIM metamodel conformant  
169 representation. This document will utilize MOF for examples of such transformations.

## 170 **4.1 CIM Formal Model**

171 CIM representations of model resources are independent of access protocol and implementation  
172 technologies.

173 The use of CIM representations enables CIMI resources to be managed together with other key cloud  
174 foundation resources such as storage, virtual machines, hardware, and operating systems that are also  
175 use CIM representations.

176 A conformant CIMI CIM Service provider shall provide CIM representations of CIMI resources that are  
177 consistent with the formal definitions of the CIMI model according to the transformations described in this  
178 specification.

179 The DMTF provides MOF representations of CIMI resources that are transformed according to this  
180 specification.

181 Although some of the CIMI CIM classes correspond to existing CIM schema, for example CIMI\_Job, no  
182 attempt has been made to derive from the CIM schema.

## 183 **4.2 Translation Rules**

184 The following sections define normative rules for translating between the CIMI resources as defined in the  
185 Cloud Infrastructure Management Interface [CIMI](#) and their representation in CIM. Though all examples  
186 are represented using MOF format, this is only one of the formats that is used to represent CIM class  
187 definitions.

### 188 **4.2.1 Common Resource Attributes**

189 CIMI CIM ordinary classes inherit from a class named CIMI\_BaseElement. This class defines the  
190 common attributes that are shared by all CIMI resources as described in [CIMI](#) section 5.7.1.

191 The class definition for CIMI\_BaseElement shall contain a property for each Attribute defined in [CIMI](#)  
192 section 5.7.1. These properties shall be derived using the Attribute translation rules defined in section  
193 4.3.3, except as noted below.

194 The “id” attribute shall be a property of type string. The “id” property shall have the “Key” qualifier. This  
195 property shall be the key property for all instances of CIMI ordinary classes.

196 See section 5 for a non-normative reference of the MOF representation of CIMI\_BaseElement.

### 197 **4.2.2 Resource Metadata**

198 Resource metadata defined in [CIMI](#) section 5.11 shall be defined in CIM following the rules defined below  
199 in sections 4.3.3. For the purposes of this document, resource metadata is information about provider-  
200 defined constraints, capabilities, or features. Resource metadata shall be represented in the same way as  
201 any other resource.

### 202 **4.2.3 Resource Translation Rules**

203 The rules described in this section produce an ordinary class definition and some number of auxiliary  
204 structure and association definitions for each resource defined in [CIMI](#). The CIM classes represented by  
205 the MOF files in section 5 conform to these rules.

206 Each CIMI resource is translated first to a CIM class definition. This will result in the definition of that  
207 class and some number other auxiliary structure, class, and association class definitions.

208 **4.2.3.1 Ordinary class definitions**

209 The schema name for ordinary class definitions shall be “CIMI” and the class name for each resource  
 210 shall be the Name of the resource as defined in [CIMI](#) and separated by an underscore, “\_”. For example,  
 211 the CIMI resource Machine would translate to class named “CIMI\_Machine”.

212 Each ordinary class shall inherit from CIMI\_BaseElement, which defines the common attributes as  
 213 specified in [CIMI](#) section 5.7.1.

214 The following CIM qualifiers apply to each ordinary class definition.

215 **Table 4-1: Qualifiers for ordinary classes**

CIM Qualifier	Value
Description	The description qualifier shall be specified with the text following the heading of the clause that defines the resource in the CIMI specification.
UMLPackagePath	The UMLPackagePath qualifier shall be specified according to the following ABNF: <code>"CIMI:" resourceName</code> resourceName is the name of the corresponding CIMI resource.
Version	The version qualifier shall be specified with the value of the CIMI specification version.

216 Each attribute of a CIMI resource is translated into either a property or an association class definition.  
 217 This following defines the rules for how to translate the attribute:

- 218 1) If the attribute is a reference or a collection, a CIM association class is created as specified in  
 219 4.2.3.2.
- 220 2) If the attribute is a simple type, a CIM property is created with a primitive type as specified in  
 221 4.2.3.4.
- 222 3) If the attribute is a Map, a well known structure named “CIMI\_Map” is used, see 4.2.3.5. The  
 223 property name shall be the same name as the CIMI attribute name and the data type shall be  
 224 String. If the CIMI attribute is an array, the property shall be an array. Table 4-2 specifies the  
 225 structure property qualifiers.
- 226 4) If the attribute is a Structure, a CIM structure is created as specified in 4.2.3.3. The property  
 227 name shall be the same name as the CIMI attribute name and the data type shall be String. If the  
 228 CIMI attribute is an array, the property shall be an array. Table 4-2 specifies the structure  
 229 property qualifiers.

230 **Table 4-2: Structure Property Qualifiers**

CIM Qualifier	Value
Description	The Description qualifier shall be specified with the text provided in the description of the attribute.
EmbeddedInstance	The EmbeddedInstance qualifier shall be specified with the name of the structure (e.g. CIMI_Volume)
Read	The Read qualifier shall be specified with value False if the Consumer Constraints listed in the description specifies "write-only"
Required	The Required qualifier shall be specified with no value if the Provider Constraints listed in the description specifies support mandatory.
Write	The Write qualifier shall be specified if the Consumer Constraints listed in the description specifies "read-write" or "write-only"

231 Each Operation in [CIMI](#) that is not an intrinsic operation shall be included as a method in the CIM class  
 232 definition. The following specifies how to map a method

- 233 • Method Name - The method name in CIM shall be the link URL as defined in [CIMI](#) with the prefix  
 234 “http://www.dmtf.org/cimi/action/” removed. For example, the Operation supported by the



235 Machine resource that is defined in [CIMI](#) with the link <http://www.dmtf.org/cimi/action/start> is  
 236 defined in CIM with a method named start.

- 237 • Return Value – The return value shall be of type uint32.
- 238 • Input Parameters – If the method includes any input paramters, the name of the input parameter  
 239 will be the same as the parameter name specified in the CIMI. The data type shall map the same  
 240 as for Simple Properties, see 4.2.3.4. Table 4-3 specifies the rules for qualifiers for IN qualified  
 241 parameters.

242 **Table 4-3: IN Parameter Qualifiers**

CIM Qualifier	Value
Description	The Description qualifier shall be specified with the text provided in the description of the parameter.
IN	The In qualifier shall be specified with a value of True.
Required	The Required qualifier shall be specified with no value if the parameter is specified as mandatory. This qualifier shall not be specified if the parameter is optional.
Units	The Units qualifier shall be specified if the description defines the value as a programmable unit liseded in DSP0004 (e.g. KiloBytes, Percent, Seconds, ...)
Values	The Values qualifier shall be specified if the parameter type is string and the description includes the phrase, "Allowable values include:" The qualifier value is the array of strings specified by the values listed in the description.

- 243
- 244 • Output Parameters – If the method includes any output paramters, the name of the input  
 245 parameter will be the same as the parameter name specified in the CIMI. The data type shall map  
 246 the same as for Simple Properties, see 4.2.3.4. Table 4-3 specifies the rules for qualifiers for IN  
 247 qualified parameters.

248 **Table 4-4: OUT Parameter Qualifiers**

CIM Qualifier	Value
Description	The Description qualifier shall be specified with the text provided in the description of the parameter.
IN	The IN qualifier shall be specified with a value of False.
OUT	The OUT qualifier shall be specified with a value of True.
Required	The Required qualifier shall be specified with no value if the parameter is specified as mandatory. This qualifier shall not be specified if the parameter is optional.
Units	The Units qualifier shall be specified if the description defines the value as a programmable unit liseded in DSP0004 (e.g. KiloBytes, Percent, Seconds, ...)
Values	The Values qualifier shall be specified if the parameter type is string and the description includes the phrase, "Allowable values include:" The qualifier value is the array of strings specified by the values listed in the description.

249

250 **4.2.3.2 Association class definitions**

251 If the attribute of the CIMI resource (excluding structures) is a reference or a collection, an association  
 252 class shall be created. The association class name shall be the concatenation of "CIMI", an underscore,  
 253 "\_", the name of the resource as defined in [CIMI](#) and the corresponding CIMI attribute name with an initial  
 254 capital letter. For example, the association with the class name of CIMI\_MachineNetwork. Table 4-5  
 255 specifies the rules for qualifiers for association classes.

256 **Table 4-5: Qualifiers for association classes**

CIM Qualifier	Value
Association	The Association qualifier shall be specified first and with no value.

CIM Qualifier	Value
Description	The Description qualifier shall be specified. The value should be the text "<classname> <attributename> association", for example "CIM_Machine eventLog association".
UMLPackagePath	The UMLPackagePath qualifier shall be specified with the value according to the following ABNF: <pre>"CIMI:" resourceName referenceName</pre> Where <code>resourceName</code> is the name of the CIMI resource that defines the CIMI attribute that is translated into the association class and <code>referenceName</code> is the name of the CIMI attribute that caused creation of this association. The <code>referenceName</code> is specified with an initial capital letter.
Version	The Version qualifier shall be specified with the value of the version of the CIMI specification.

257 The association shall include two reference properties. The first is a reference to the CIM class  
 258 representing the CIMI resource that included the reference or collection property. The description shall  
 259 be "The <classname>", where <classname> is the classname for example, CIMI\_Machine. The second  
 260 shall be a reference to the CIM class corresponding to the referenced or collected CIMI resource. The  
 261 description shall be the description of the original CIMI attribute. Table 4-6 specifies the rules for qualifiers  
 262 that apply to reference properties.

263 **Table 4-6: Reference Property Qualifiers**

CIM Qualifier	Value
Key	The Key qualifier shall be specified as the first qualifier with no arguments.
Description	The Description qualifier shall be specified with the text provided in the description of the attribute.
Min	The Min qualifier shall be specified if the minimum number of referenced instances is not 0.
Max	The Max qualifier shall be specified with a value if the maximum number of referenced instances is not unlimited. If the CIMI attribute is not an array, the Max qualifier shall be specified with a value of one (1)

264

265 **4.2.3.3 Structure definitions**

266 A structure will be created if the attribute of a CIMI resource is a structure. The structure class name shall  
 267 be the concatenation of "CIMI", an underscore, "\_", the name of the resource as defined in [CIMI](#) and the  
 268 corresponding CIMI attribute name with an initial capital letter. For example,  
 269 CIMI\_MachineConfigurationDisks.

270 If the CIMI resource is a map, the following structure

271 The following CIM qualifiers apply to each structure definition.

272 **Table 4-7: Qualifiers for structures**

CIM Qualifier	Value
Indication	The Indication qualifier shall be specified with no arguments first
Structure	The Structure qualifier shall be specified with no arguments second
Description	The Description qualifier shall be specified with the text following the CIMI attribute that references this structure.
UMLPackagePath	The UMLPackagePath qualifier shall be specified with the value according to the following ABNF: <pre>"CIMI:" resourceName</pre> Where <code>resourceName</code> is the name of the corresponding CIMI resource.
Version	The Version qualifier shall be specified with the version of the CIMI specification

273 For each attribute of the CIMI structure, a property shall be created. The following rules apply:

- 274 1) If the attribute has a simple type, then it translates to a CIM property with a primitive type, see  
275 4.2.3.4.
- 276 2) If the attribute is a ref, then it translates the same as if it were a URI, see 4.2.3.4.
- 277 3) If the attribute is a Map, a well known structure named "CIM\_Map" is used, see 4.2.3.5. The  
278 property name shall be the same name as the CIMI attribute name and the data type shall be  
279 String. If the CIMI attribute is an array, the property shall be an array. Table 4-2 specifies the  
280 structure property qualifiers.
- 281 4) If the attribute is a Structure, a CIM structure is created as specified in 4.2.3.3. The property  
282 name shall be the same name as the CIMI attribute name and the data type shall be String. If the  
283 CIMI attribute is an array, the property shall be an array. Table 4-2 specifies the structure  
284 property qualifiers.

285 **4.2.3.4 Simple Properties**

286 The CIMI defines a set of data type (section 5.5)

287 Table 4-8 defines the translation between CIMI and CIM primitive types.

288 **Table 4-8: Primitive Type mapping**

CIMI	MOF
boolean	boolean
dateTime	datetime
duration	datetime
integer	uint8 sint8 uint16 sint16 uint32 sint32 uint64 sint64
string	string
byte[]	uint8[]
URI	string

289 The property name of a CIMI attribute with a primitive type shall be the same as the CIMI attribute name.  
290 The property type shall be the CIM primitive type from Table 4-8. There are multiple mappings for the  
291 CIMI integer type. The modeler may exercise judgment. However if there is any doubt, sint64 should be  
292 chosen. If the CIMI specification attribute is an array, then the CIM property shall be an array. Table 4-9  
293 defines qualifiers that apply to simple properties.

294 **Table 4-9: Simple Property Qualifiers**

CIM Qualifier	Value
Description	The Description qualifier shall be specified with the text provided in the description of the attribute.
Read	The Read qualifier shall be specified with value False if the Consumer Constraints listed in the description specifies "write-only"
Reference	The Reference qualifier shall be specified if the CIMI type is URI.
Required	The Required qualifier shall be specified with no value if the Provider Constraints listed in the description specifies support mandatory.
Units	The Units qualifier shall be specified if the description defines the value as a programmable unit listed in DSP0004 (e.g. KiloBytes, Percent, Seconds, ...)
Values	The Values qualifier shall be specified if the attribute type is string and the description includes the phrase, "Allowable values include:" The qualifier value is the array of strings specified by the highlighted values listed in the description.
Write	The Write qualifier shall be specified with no value if the Consumer Constraints listed in the

	description specifies "read-write" or "write-only"
--	--

#### 295 4.2.3.5 Map

296 CIMI defines a Map of key/value pairs. The following structure is used to represent a Map.

```

297
298 [Indication, Structure, Version("1.0.0"),
299     Description("CIMI Map"),
300     UMLPackagePath ( "CIMI::Map" )]
301 CIMI_Map {
302
303     [Description("The key.")]
304     string Key;
305
306     [Description("The value.")]
307     string Value;
308 }
```

## 309 5 CIMI CIM MOF Representation Examples

310 The following sections shows examples of CIMI entities represented as CIM MOF classes.

311 The normative CIM meta-model representations are published by the DMTF. The representations are  
312 published in MOF and other formats.

313 The following non-normative copies of the MOF files are provided for illustration. Where any differences  
314 occur between the published MOF files and the copies below, the published MOF files shall be  
315 considered authoritative.

316 The Cloud Infrastructure Management Interface classes are defined in a schema with the prefix CIMI and  
317 derived from a common root class CIMI\_BaseElement, which does not derive from any DMTF standard  
318 CIM schema class.

### 319 5.1 Ordinary Class

#### 320 5.1.1 CIMI\_BaseElement

321 Defined in: CIMI\_BaseElement.mof

```

322 [Abstract, Version ( "1.0.0" ),
323     UMLPackagePath ( "CIMI::BaseElement" ),
324     Description ( "Common properties for all CMWG classes" )]
325 class CIMI_BaseElement {
326
327     [Key, Description (
328         "The unique self-reference to this resource; assigned upon"
329         "resource creation. This attribute value shall be unique in the"
330         "Provider's cloud."
331     )]
332     string id;
333
334     [Required, Write, Description (
```

```

335     "The human readable name of this resource; assigned by the "
336     "creator as a part of the resource creation input." )]
337     string name;
338
339     [Required, Write, Description (
340     "The human readable description of this resource; assigned "
341     "by the creator as a part of the resource creation input." )]
342     string description;
343
344     [Description (
345     "The timestamp when this resource was created. The format "
346     "should be unambiguous, and the value is immutable")]
347     datetime created;
348
349     [Description (
350     "The time at which the last explicit attribute update "
351     "was made on the resource. Note, while operations such "
352     "as \"stop\" do implicitly modify the \"state\" attribute "
353     "it does not change the \"updated_time\"." )]
354     datetime updated;
355 };

```

## 356 5.1.2 CIMI\_Machine

357 Defined in: CIMI\_Machine.mof

```

358     [Version("1.0.0"), Description(
359     "An instantiated compute resource that encapsulates both CPU and Memory."),
360     UMLPackagePath ( "CIMI::Machine" )]
361     Class CIMI_Machine : CIMI_BaseElement {
362
363     [Required, Description(
364     "The operational state of the Machine.\n"
365     "Allowable values include:\n"
366     "CREATING: The Machine is in the process of being created. "
367     "Allowable action when in this state is: delete.\n"
368     "STARTING: The Machine is in the process of being started. "
369     "Allowable actions when in this state are: start, restart, "
370     "stop, and delete.\n"
371     "STARTED: The Machine is available and ready for use. Allowable actions "
372     "when in this state are: stop, restart, pause, suspend, capture, "
373     "and delete.\n"
374     "STOPPING: The Machine is in the process of being stopped. Allowable "
375     "actions when in this state are: start, restart, stop, and delete. "
376     "STOPPED: This value is the virtual equivalent of powering off a physical "
377     "Machine. There is no saved CPU or memory state. Allowable actions when "
378     "in this state are: start, restart, capture, and delete.\n"
379     "PAUSING: The Machine in the process of being PAUSED. Allowable actions "
380     "when in this state are: start, restart, and delete.\n"
381     "PAUSED: In this state the Machine and its virtual resources remain "

```

```

382     "instantiated and resources remain allocated, similar to the STARTED "
383     "state, but the Machine and its virtual resources are not enabled to "
384     "perform tasks. Allowable actions when in this state are: start, restart, "
385     "capture, and delete.\n"
386     "SUSPENDING: The Machine is in the process of being suspended. Allowable "
387     "actions when in this state are: start, restart, and delete.\n"
388     "SUSPENDED: In this state the Machine and its virtual resources are stored "
389     "on non-volatile storage. The Machine and its resources are not enabled to "
390     "perform tasks. Allowable actions when in this state are: start, restart, "
391     "capture, and delete.\n"
392     "DELETING: The Machine is in the process of being deleted. Allowable "
393     "action when in this state is: delete.\n"
394     "ERROR: The Provider has detected an error in the Machine. Allowable "
395     "actions when in this state are: start, restart, stop, and delete.\n"
396     "PAUSED and SUSPENDED states are optional and Providers may choose to "
397     "support them or not.\n"
398     "Providers may define additional values.")
399     Values{"CIMI_CREATING", "CIMI_STARTING", "CIMI_STOPPING", "CIMI_STOPPED",
400           "CIMI_PAUSING", "CIMI_PAUSED", "CIMI_SUSPENDING",
401           "CIMI_SUSPENDED", "CIMI_DELETING", "CIMI_ERROR",
402           "CIMI_PAUSED", "CIMI_SUSPENDED"}]
403     String state;
404
405     [Description("The amount of CPU that this Machine has.")]
406     Uint32 cpu;
407
408     [Required,
409     Description(
410         "The size of the memory (RAM) allocated to this Machine.\n\n"
411         "When this value is increased, it implies that the Machine is allocated "
412         "more RAM, and vice versa when the value is decreased.")]
413     Uint64 memory;
414
415     [Description(
416         "The CPU architecture that will be supported by Machines created by using "
417         "this configuration.\n"
418         "Allowable values include: 68000, Alpha, ARM, Itanium, MIPS, PA_RISC, "
419         "POWER, PowerPC, x86, x86_64, z/Architecture, SPARC. Providers may define "
420         "additional values."),
421     Values{"CIMI_68000", "CIMI_Alpha", "CIMI_ARM", "CIMI_Itanium", "CIMI_MIPS",
422           "CIMI_PA_RISC", "CIMI_POWER", "CIMI_PowerPC", "CIMI_x86",
423           "CIMI_x86_64", "CIMI_z/Architecture", "CIMI_SPARC"}]
424     String cpuArch;
425 };

```

### 426 5.1.3 CIMI\_Disk

427 Defined in: CIMI\_Disk.mof

```
428     [Version("1.0.0"), Description(  
429         "The size of the memory (RAM) allocated to this Machine. "  
430         "When this value is increased, it implies that the Machine is allocated more "  
431         "RAM, and vice versa when the value is decreased. "  
432         "This attribute has the following sub-attributes that serve to describe it:")]  
433 CIMI_Disk {  
434  
435     [Required, Description(  
436         "The initial capacity, in kilobytes, of the disk. "),  
437         Units ( "KiloBytes" )]  
438     String capacity;  
439  
440     [Description(  
441         "Operating System specific location(path) in its namespace where this disk "  
442         "will first appear. Note, once deployed Consumers might move where this "  
443         "Disk is located.\n"  
444         "Support of this attribute indicates that the Provider can report this "  
445         "information back to the Consumer.")]  
446     String initialLocation;  
447 };
```

## 448 5.2 Association

### 449 5.2.1 CIMI\_MachineEventLog

450 Defined in: CIMI\_MachineEventLog.mof

```
451     [Association, Version("1.0.0"),  
452     Description("CIMI_Machine eventLog association"),  
453     UMLPackagePath ( "CIMII::Machine" )]  
454 CIMI_MachineEventLog {  
455  
456     [Key, Description("The CIMI_Machine")]  
457     CIMI_Machine REF machine;  
458  
459     [Key, MAX(1), Description(  
460         "A reference to the EventLog of this Machine.")]  
461     CIMI_EventLog REF eventLog;  
462 };
```

## 463 5.2.2 CIMI\_MachineLatestSnapshot

464 Defined in: CIMI\_MachineLatestSnapshot.mof

```

465 [Association, Version("1.0.0"),
466     Description("CIM_Machine latestSnapshot association"),
467     UMLPackagePath ( "CIMI::Machine" )]
468 CIMI_MachineLatestSnapshot {
469
470     [Key, Description("The CIMI_Machine")]
471     CIMI_Machine REF machine;
472
473     [Key, Max(1), Description(
474         "A reference to the SNAPSHOT representing the latest state captured for "
475         "this Machine (either most recent Snapshot or the last Snapshot reverted "
476         "to)."]
477     CIMI_MachineImage REF latestSnapshot;
478 };

```

## 479 5.3 Structure

### 480 5.3.1 CIMI\_MachineTemplateVolumes

481 Defined in: CIMI\_MachineTemplateVolumes.mof

```

482 [Indication, Structure, Version("1.0.0"),
483     Description(
484         "CIMI_Machine NetworkInterfaces association."),
485     UMLPackagePath ( "CIMI::MachineTemplateVolumes" )]
486 CIMI_MachineTemplateVolumes {
487
488     [Description(
489         "An Operating System specific location(path) in its namespace where "
490         "the Volume will appear. Support of this attribute indicates that the "
491         "Provider allows for Consumers to choose where the Volume will appear.")
492     String initialLocation;
493
494     [Required, Reference, Description(
495         "Reference to the Volume that will be connected.")
496     String volume;
497

```



498  
499  
500**ANNEX A**  
**(informative)**  
**Change log**

<b>Version</b>	<b>Date</b>	<b>Description</b>
1.0.0a	09/07/2011	Released as a Work in Progress
1.0.0b	06/13/2012	Released as a Work in Progress
1.0.0c	09/27/2012	Work in Progress

501

## Bibliography

502 **DMTF DSP-ISO102**, Distributed Management Task Force, Inc., *Architecture for Managing Clouds White*  
503 *Paper 1.0*, [http://dmf.org/sites/default/files/standards/documents/DSP-ISO102\\_1.0.0.pdf](http://dmf.org/sites/default/files/standards/documents/DSP-ISO102_1.0.0.pdf)

504 **DMTF DSP-ISO103**, Distributed Management Task Force, Inc., *Use Cases and Interactions for Managing*  
505 *Clouds 1.0.0*, [http://www.dmtf.org/sites/default/files/standards/documents/DSP-ISO103\\_1.0.0.pdf](http://www.dmtf.org/sites/default/files/standards/documents/DSP-ISO103_1.0.0.pdf)

506