



1

2

3

4

Document Identifier: DSP1035

Date: 2019-03-18

Version: 1.0.3

5

Host LAN Network Port Profile

6

Supersedes: 1.0.2

7

Document Class: Normative

8

Document Status: Published

9

Document Language: en-US

10 Copyright Notice

11 Copyright © 2008, 2010-2011, 2019 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26 withdrawn or modified after publication, and shall be indemnified and held harmless by any party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 This document's normative language is English. Translation into other languages is permitted.

CONTENTS

34	Foreword	6
35	Introduction.....	7
36	1 Scope	9
37	2 Normative references	9
38	3 Terms and definitions	9
39	4 Symbols and abbreviated terms.....	10
40	5 Synopsis	10
41	6 Description	11
42	7 Implementation requirements.....	12
43	7.1 Representing a network port.....	12
44	7.2 Representing a communication endpoint	15
45	7.3 Managing network endpoints.....	17
46	7.4 Representing multiple ports controlled from a single controller.....	17
47	8 Methods.....	19
48	8.1 CIM_NetworkPortConfigurationService.AddLANEndpoint().....	19
49	8.2 CIM_NetworkPort.RequestStateChange().....	21
50	8.3 CIM_LANEndpoint.RequestStateChange().....	22
51	8.4 CIM_PortController.RequestStateChange().....	23
52	8.5 Profile conventions for operations	24
53	8.6 CIM_ControlledBy.....	25
54	8.7 CIM_DeviceSAPImplementation	25
55	8.8 CIM_ElementCapabilities	25
56	8.9 CIM_EnabledLogicalElementCapabilities.....	25
57	8.10 CIM_HostedAccessPoint	26
58	8.11 CIM_HostedService	26
59	8.12 CIM_LANEndpoint	26
60	8.13 CIM_NetworkPort.....	27
61	8.14 CIM_NetworkPortConfigurationService	28
62	8.15 CIM_PortController	28
63	8.16 CIM_Realizes.....	28
64	8.17 CIM_ServiceAffectsElement	29
65	8.18 CIM_SystemDevice	29
66	9 Use cases.....	29
67	9.1 Object diagrams.....	29
68	9.2 Querying MAC address for an interface	33
69	9.3 Determining physical connector for a network address	33
70	9.4 Determining whether physical communication is possible	34
71	9.5 Correlating controller and port	34
72	9.6 Adding an endpoint to the port.....	34
73	9.7 Determining whether ElementName can be modified	36
74	9.8 Determining whether state management is supported	37
75	10 CIM Elements	37
76	10.1 CIM_ControlledBy.....	37
77	10.2 CIM_DeviceSAPImplementation	38
78	10.3 CIM_ElementCapabilities — LANEndpoint	38
79	10.4 CIM_ElementCapabilities — NetworkPort.....	38
80	10.5 CIM_ElementCapabilities — PortController	39
81	10.6 CIM_EnabledLogicalElementCapabilities — LANEndpoint.....	39
82	10.7 CIM_EnabledLogicalElementCapabilities — NetworkPort	39
83	10.8 CIM_EnabledLogicalElementCapabilities — PortController.....	40
84	10.9 CIM_HostedAccessPoint	40
85	10.10 CIM_HostedService	40

86 10.11 CIM_LANEndpoint 41

87 10.12 CIM_NetworkPort..... 41

88 10.13 CIM_NetworkPortConfigurationService 42

89 10.14 CIM_PhysicalConnector 42

90 10.15 CIM_PortController 43

91 10.16 CIM_Realizes..... 43

92 10.17 CIM_RegisteredProfile..... 44

93 10.18 CIM_ServiceAffectsElement 44

94 10.19 CIM_SystemDevice — CIM_NetworkPort 44

95 10.20 CIM_SystemDevice — CIM_PortController..... 45

96 ANNEX A (informative) Change log..... 46

97

98 **Figures**

99 Figure 1 – Host LAN Network Port Profile: Class diagram 12

100 Figure 2 – Registered Profile 30

101 Figure 3 – Single interface 31

102 Figure 4 – Single interface, separate card 32

103 Figure 5 – One controller for two ports 33

104 Figure 6 – Endpoint management supported..... 35

105 Figure 7 – Second endpoint added..... 36

106

107 **Tables**

108 Table 1 – Referenced profiles 11

109 Table 2 – CIM_NetworkPortConfigurationService.AddLANEndpoint() method: Return code values..... 21

110 Table 3 – CIM_NetworkPortConfigurationService.AddLANEndpoint() method: Parameters 21

111 Table 4 – CIM_NetworkPort.RequestStateChange() method: Return code values 21

112 Table 5 – CIM_NetworkPort.RequestStateChange() method: Parameters 22

113 Table 6 – CIM_LANEndpoint.RequestStateChange() method: Return code values 23

114 Table 7 – CIM_LANEndpoint.RequestStateChange() method: Parameters 23

115 Table 8 – CIM_PortController.RequestStateChange() method: Return code values 24

116 Table 9 – CIM_PortController.RequestStateChange() method: Parameters 24

117 Table 10 – Operations: CIM_ControlledBy 25

118 Table 11 – Operations: CIM_DeviceSAPImplementation 25

119 Table 12 – Operations: CIM_ElementCapabilities 25

120 Table 13 – Operations: CIM_HostedAccessPoint..... 26

121 Table 14 – Operations: CIM_HostedService 26

122 Table 15 – Operations: CIM_LANEndpoint..... 26

123 Table 16 – Operations: CIM_NetworkPort 27

124 Table 17 – Operations: CIM_PortController..... 28

125 Table 18 – Operations: CIM_Realizes 28

126 Table 19 – Operations: CIM_ServiceAffectsElement 29

127 Table 20 – Operations: CIM_SystemDevice 29

128 Table 21 – CIM Elements: Network Port Profile 37

129 Table 22 – Class: CIM_ControlledBy..... 37

130 Table 23 – Class: CIM_DeviceSAPImplementation 38

131 Table 24 – Class: CIM_ElementCapabilities — LANEndpoint..... 38

132 Table 25 – Class: CIM_ElementCapabilities — NetworkPort..... 38

133 Table 26 – Class: CIM_ElementCapabilities — PortController..... 39

134 Table 27 – Class: CIM_EnabledLogicalElementCapabilities — LANEndpoint..... 39

135 Table 28 – Class: CIM_EnabledLogicalElementCapabilities — NetworkPort 39

136 Table 29 – Class: CIM_EnabledLogicalElementCapabilities — PortController..... 40

137 Table 30 – Class: CIM_HostedAccessPoint 40

138 Table 31 – Class: CIM_HostedService 40

139 Table 32 – Class: CIM_LANEndpoint 41

140 Table 33 – Class: CIM_NetworkPort..... 41

141 Table 34 – Class: NetworkPortConfigurationService..... 42

142 Table 35 – Class: CIM_PhysicalConnector 42

143 Table 36 – Class: CIM_PortController 43

144 Table 37 – Class: CIM_Realizes..... 43

145 Table 38 – Class: CIM_RegisteredProfile..... 44

146 Table 39 – Class: CIM_ServiceAffectsElement 44

147 Table 40 – Class: CIM_SystemDevice 44

148 Table 41 – Class: CIM_SystemDevice 45

149

150

Foreword

151 The *Host LAN Network Port Profile* (DSP1035) was prepared by the Physical Platform Profiles Working
152 Group.

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

155 Acknowledgments

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

157 Editors:

- 158 • Jeff Hilland – Hewlett Packard Enterprise
- 159 • Aaron Merkin – IBM
- 160 • Hemal Shah – Broadcom

161 Contributors:

- 162 • Jon Hass – Dell
- 163 • Jeff Hilland – Hewlett Packard Enterprise
- 164 • John Leung – Intel
- 165 • Aaron Merkin – IBM
- 166 • Khachatur Papanyan – Dell
- 167 • Sivakumar Sathappan – AMD
- 168 • Hemal Shah – Broadcom
- 169 • Christina Shaw – Hewlett Packard Enterprise
- 170 • Enoch Suen – Dell
- 171 • Perry Vincent – Intel

172

Introduction

173 The information in this specification should be sufficient for a provider or consumer of this data to identify
174 unambiguously the classes, properties, methods, and values that shall be instantiated and manipulated to
175 represent and manage a network port that provides a LAN interface to a host and its associated
176 configuration information. The target audience for this specification is implementers who are writing CIM-
177 based providers or consumers of management interfaces that represent the component described in this
178 document.
179

180

181

Host LAN Network Port Profile

182 1 Scope

183 The *Host LAN Network Port Profile* extends the management capability of referencing profiles by adding
184 the capability to represent a network port that provides a LAN interface to a host system, its associated
185 controller, and network interfaces. Associations with the port's physical aspects and profile-
186 implementation version information are modeled in this profile.

187 2 Normative references

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

192 DMTF DSP0004, *CIM Infrastructure Specification 2.6*,
193 https://www.dmtf.org/sites/default/files/standards/documents/DSP0004_2.6.pdf

194 DMTF DSP0200, *CIM Operations over HTTP 1.3*,
195 https://www.dmtf.org/sites/default/files/standards/documents/DSP0200_1.3.pdf

196 DMTF DSP1001, *Management Profile Specification Usage Guide 1.0*,
197 https://www.dmtf.org/sites/default/files/standards/documents/DSP1001_1.0.pdf

198 DMTF DSP1011, *Physical Asset Profile 1.0*,
199 https://www.dmtf.org/sites/default/files/standards/documents/DSP1011_1.0.pdf

200 DMTF DSP1033, *Profile Registration Profile 1.0*,
201 https://www.dmtf.org/sites/default/files/standards/documents/DSP1033_1.0.pdf

202 ISO/IEC Directives, Part 2, *Rules for the structure and drafting of International Standards*,
203 <http://isotc.iso.org/livelink/livelink.exe?func=ll&objId=4230456&objAction=browse&sort=subtype>

204 3 Terms and definitions

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

207 The terms "shall" ("required"), "shall not", "should" ("recommended"), "should not" ("not recommended"),
208 "may", "need not" ("not required"), "can" and "cannot" in this document are to be interpreted as described
209 in [ISO/IEC Directives, Part 2](#), Clause 7. The terms in parentheses are alternatives for the preceding term,
210 for use in exceptional cases when the preceding term cannot be used for linguistic reasons. Note that
211 [ISO/IEC Directives, Part 2](#), Clause 7 specifies additional alternatives. Occurrences of such additional
212 alternatives shall be interpreted in their normal English meaning.

213 The terms "clause", "subclause", "paragraph", and "annex" in this document are to be interpreted as
214 described in [ISO/IEC Directives, Part 2](#), Clause 6.

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

218 The terms defined in [DSP0004](#), [DSP0223](#), and [DSP1001](#) apply to this document. The following additional
219 terms are used in this document.

220 3.1

221 conditional

222 indicates requirements to be followed strictly in order to conform to the document when the specified
223 conditions are met

224 3.2

225 mandatory

226 indicates requirements to be followed strictly in order to conform to the document and from which no
227 deviation is permitted

228 3.3

229 optional

230 indicates a course of action permissible within the limits of the document

231 3.4

232 referencing profile

233 indicates a profile that owns the definition of this class and can include a reference to this profile in its
234 "Related Profiles" table

235 4 Symbols and abbreviated terms

236 The abbreviations defined in [DSP0004](#), [DSP0223](#), and [DSP1001](#) apply to this document. The following
237 additional abbreviations are used in this document.

238 4.1

239 DNS

240 Domain Name System

241 4.2

242 DHCP

243 Dynamic Host Configuration Protocol

244 4.3

245 LAN

246 Local Area Network

247 5 Synopsis

248 **Profile Name:** Host LAN Network Port

249 **Version:** 1.0.2

250 **Organization:** DMTF

251 **CIM Schema version:** 2.22

252 **Central Class:** CIM_NetworkPort

253 **Scoping Class:** CIM_ComputerSystem

254 This abstract profile specification shall not be directly implemented; implementations shall be based on a
255 profile specification that specializes the requirements of this profile.

256 The *Host LAN Network Port Profile* extends the management capability of referencing profiles by adding
 257 the capability to represent a network port that provides a LAN interface in a managed system. This profile
 258 includes a specification of the network port, associated controller, associated network endpoint, and the
 259 realization of the connection in a physical connector.

260 CIM_NetworkPort shall be the Central Class of this profile. The instance of CIM_NetworkPort shall be the
 261 Central Instance of this profile. CIM_ComputerSystem shall be the Scoping Class of this profile. The
 262 instance of CIM_ComputerSystem with which the Central Instance is associated through an instance of
 263 CIM_SystemDevice shall be the Scoping Instance of this profile.

264 Table 1 identifies profiles on which this profile has a dependency.

265 **Table 1 – Referenced profiles**

Profile Name	Organization	Version	Description
Profile Registration	DMTF	1.0	Mandatory
Physical Asset	DMTF	1.0	Optional. See 7.1.6.

266 6 Description

267 The *Host LAN Network Port Profile* describes a network port and, optionally, an associated controller,
 268 associated network interfaces, and the realization of the connection in a physical connector.

269 The following functionality is mandatory within the scope of this profile:

- 270 • a specification of the network port and related hardware
- 271 • network interfaces active over the network port

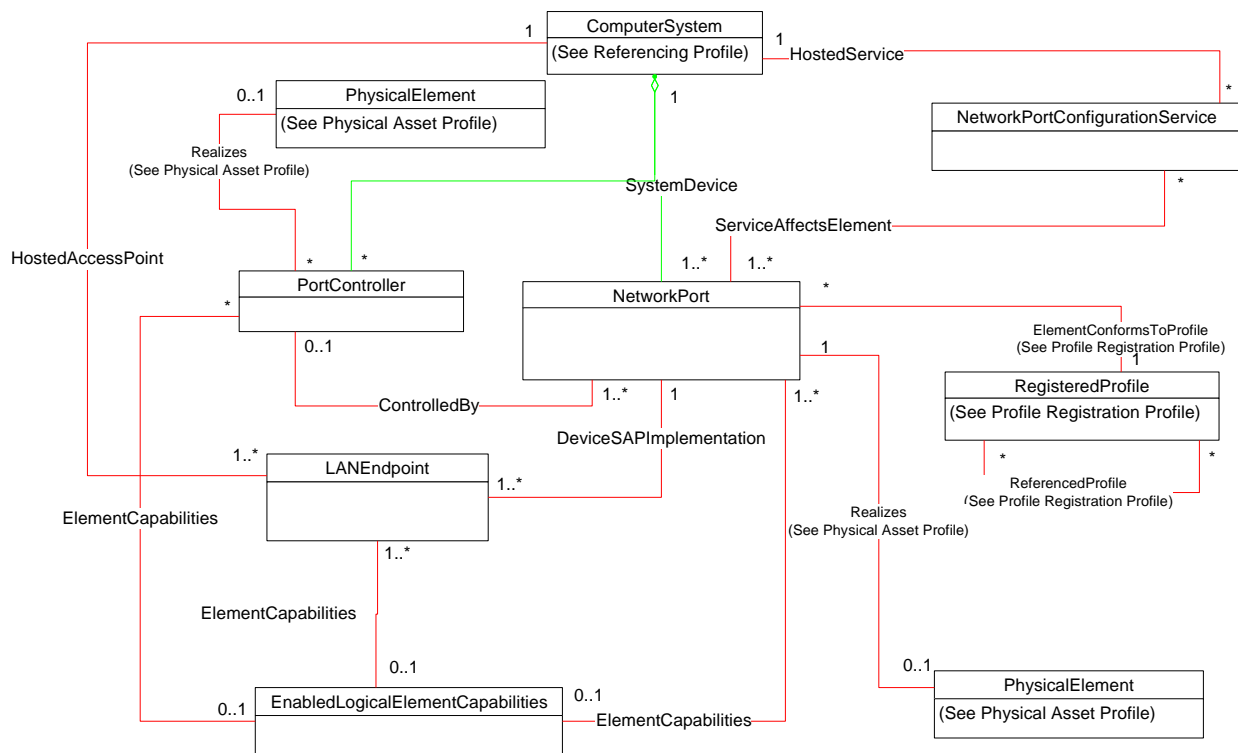
272 The following functionality is optional within the scope of this profile:

- 273 • modeling of the controller and its relationship with the network port

274 The following functionality is not covered in this profile:

- 275 • modeling of the networks in which the network interface participates

276 Figure 1 represents the class schema for the *Host LAN Network Port Profile*. For simplicity, the prefix
 277 CIM_ has been removed from the names of the classes. The CIM_NetworkPort class represents a
 278 network port of the system with one or more communication endpoints (that is, a communication
 279 interface) represented through CIM_LANEndpoint. A given CIM_LANEndpoint on the network port is
 280 identified by a MAC address to which the network port will respond. A network port can have an
 281 associated controller. The controller is represented by an instance of CIM_PortController. The
 282 relationship between the controller and port is modeled through the CIM_ControlledBy association. The
 283 CIM_NetworkPortConfigurationService class provides the ability to manage network interfaces associated
 284 with a network port.



285

286

Figure 1 – Host LAN Network Port Profile: Class diagram

287 7 Implementation requirements

288 This clause details the requirements related to the arrangement of instances and properties of instances
 289 for implementations of this profile.

290 7.1 Representing a network port

291 An instance of CIM_NetworkPort shall represent the network port.

292 7.1.1 CIM_NetworkPort.EnabledState — Enabled but Offline

293 A value of 6 (Enabled but Offline) shall indicate that the underlying device is enabled but cannot
 294 communicate with the physical network. For example, this state is appropriate if the network cable is not
 295 attached to the physical connector.

296 7.1.2 Network Port state management is supported — conditional

297 When management of the state of a Network Port is supported, exactly one instance of
 298 CIM_EnabledLogicalElementCapabilities shall be associated with the CIM_NetworkPort instance through
 299 an instance of CIM_ElementCapabilities.

300 Support for managing the state of the Network Port is optional behavior. This clause describes the CIM
 301 elements and behaviors that shall be implemented when this behavior is supported.

302 **Conditional determination:** A client can determine whether state management is supported as follows:

- 303 1) Find the CIM_EnabledLogicalElementCapabilities instance that is associated with the
 304 CIM_NetworkPort instance.

305 2) Query the value of the RequestedStatesSupported property. If at least one value is specified,
306 state management is supported.

307 **7.1.2.1 CIM_EnabledLogicalElementCapabilities**

308 When state management is supported, exactly one instance of CIM_EnabledLogicalElementCapabilities
309 shall be associated with the CIM_NetworkPort instance through an instance of the
310 CIM_ElementCapabilities association and it shall be subject to the conditions in this clause.

311 **7.1.2.1.1 CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported**

312 The RequestedStatesSupported property may contain one or more of the following values: 2 (Enabled), 3
313 (Disabled), or 11 (Reset).

314 **7.1.2.2 CIM_NetworkPort.RequestedState**

315 When the CIM_NetworkPort.RequestStateChange() method is successfully invoked, the value of the
316 RequestedState property shall be the value of the RequestedState parameter. If the method is not
317 successfully invoked, the value of the RequestedState property is indeterminate.

318 The CIM_NetworkPort.RequestedState property shall have one of the values specified in the
319 CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported property or 5 (No Change).

320 **7.1.2.3 CIM_NetworkPort.EnabledState**

321 When the RequestedState parameter has a value of 2 (Enabled) or 3 (Disabled) and the
322 CIM_NetworkPort.RequestStateChange() method completes successfully, the value of the EnabledState
323 property shall equal the value of the CIM_NetworkPort.RequestedState property.

324 If the method does not complete successfully, the value of the EnabledState property is indeterminate.

325 The EnabledState property shall have the value 2 (Enabled), 3 (Disabled), or 6 (Enabled but Offline).

326 **7.1.3 Network Port state management is not supported**

327 This clause describes the CIM elements and behaviors that shall be implemented when management of
328 the Network Port state is not supported.

329 **7.1.3.1 CIM_EnabledLogicalElementCapabilities**

330 When state management is not supported, exactly one instance of
331 CIM_EnabledLogicalElementCapabilities may be associated with the CIM_NetworkPort instance through
332 an instance of the CIM_ElementCapabilities association and it shall be subject to the conditions in this
333 clause.

334 **7.1.3.1.1 CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported**

335 The CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported property shall not contain any
336 values.

337 **7.1.3.2 CIM_NetworkPort.RequestedState**

338 The RequestedState property shall have the value 12 (Not Applicable).

339 **7.1.3.3 CIM_NetworkPort.EnabledState**

340 The EnabledState property shall have one of the following values: 2 (Enabled), 3 (Disabled), 5 (Not
341 Applicable), or 6 (Enabled but Offline).

342 **7.1.4 Modifying ElementName is supported — conditional**

343 The CIM_NetworkPort.ElementName property may support being modified by the ModifyInstance
344 operation. See 8.13.1.1. This behavior is conditional. This clause describes the CIM elements and
345 behavior requirements when an implementation supports client modification of the
346 CIM_NetworkPort.ElementName property.

347 **Client Determination:** A client can determine whether it can modify the ElementName as follows:

- 348 1) Find the CIM_EnabledLogicalElementCapabilities instance that is associated with the
349 CIM_NetworkPort instance.
- 350 2) Query the value of the ElementNameEditSupported property of the instance. If the value is
351 TRUE, the client can modify the CIM_NetworkPort.ElementName property.

352 **7.1.4.1 CIM_EnabledLogicalElementCapabilities**

353 An instance of CIM_EnabledLogicalElementCapabilities shall be associated with the CIM_NetworkPort
354 instance through an instance of CIM_ElementCapabilities.

355 **7.1.4.1.1 CIM_EnabledLogicalElementCapabilities.ElementNameEditSupported**

356 This property shall have a value of TRUE when the implementation supports client modification of the
357 CIM_NetworkPort.ElementName property.

358 **7.1.4.1.2 CIM_EnabledLogicalElement.MaxElementNameLen**

359 The MaxElementNameLen property shall be implemented.

360 **7.1.5 Modifying ElementName is not supported**

361 This clause describes the CIM elements and behaviors that shall be implemented when the
362 CIM_NetworkPort.ElementName does not support being modified by the ModifyInstance operation.

363 **7.1.5.1 CIM_EnabledLogicalElementCapabilities**

364 An instance of CIM_EnabledLogicalElementCapabilities may be associated with the CIM_NetworkPort
365 instance through an instance of CIM_ElementCapabilities.

366 **7.1.5.1.1 CIM_EnabledLogicalElementCapabilities.ElementNameEditSupported**

367 This property shall have a value of FALSE when the implementation does not support client modification
368 of the CIM_NetworkPort.ElementName property.

369 **7.1.5.1.2 CIM_EnabledLogicalElement.MaxElementNameLen**

370 The MaxElementNameLen property may be implemented. The MaxElementNameLen property is
371 irrelevant in this context.

372 **7.1.6 Representing the physical packaging**

373 Support for representing the physical packaging of the network device is optional. The physical packaging
374 may be modeled using one or more instances of CIM_PhysicalElement in accordance with [DSP1011](#).

375 In addition, an implementation may use an instance of CIM_PhysicalConnector to represent the physical
376 connector. When an implementation instruments an instance of CIM_PhysicalConnector to represent the
377 physical connector of the network device for connecting to the network, the instance of
378 CIM_PhysicalConnector shall be compliant with [DSP1011](#). Instrumentation of the CIM_Realizes class is
379 conditional. If a corresponding instance of CIM_PhysicalConnector is instantiated, it shall be associated
380 to the corresponding CIM_NetworkPort via a CIM_Realizes instance.

381 **7.2 Representing a communication endpoint**

382 At least one instance of CIM_LANEndpoint shall represent a communication endpoint at the data-link
383 layer.

384 **7.2.1 Endpoint identified by hardware MAC**

385 There shall be exactly one instance of CIM_LANEndpoint in which the MACAddress property has the
386 same value as the PermanentAddress property of the associated CIM_NetworkPort instance.

387 **7.2.2 Communication endpoint identified by assigned MAC**

388 For each communication endpoint of the network port, there shall be exactly one instance of
389 CIM_LANEndpoint in which the MACAddress property contains the value of a MAC address to which the
390 network port will respond.

391 **7.2.3 Relationship between the interface and port**

392 For each instance of CIM_LANEndpoint, one instance of CIM_DeviceSAPImplementation shall associate
393 the CIM_LANEndpoint with the CIM_NetworkPort.

394 **7.2.4 Endpoint state management is supported — conditional**

395 When management of the state of a port endpoint is supported, exactly one instance of
396 CIM_EnabledLogicalElementCapabilities shall be associated with the CIM_LANEndpoint instance
397 through an instance of CIM_ElementCapabilities.

398 Support for managing the state of the port endpoint is optional behavior. This clause describes the CIM
399 elements and behaviors that shall be implemented when this behavior is supported.

400 **7.2.4.1 CIM_EnabledLogicalElementCapabilities**

401 When state management is supported, exactly one instance of CIM_EnabledLogicalElementCapabilities
402 shall be associated with the CIM_LANEndpoint instance through an instance of the
403 CIM_ElementCapabilities association.

404 **7.2.4.1.1 CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported**

405 The RequestedStatesSupported property may contain zero or more of the following values: 2 (Enabled),
406 3 (Disabled), or 11 (Reset).

407 **7.2.4.2 CIM_LANEndpoint.RequestedState**

408 When the CIM_LANEndpoint.RequestStateChange() method is successfully invoked, the value of the
409 RequestedState property shall be the value of the RequestedState parameter. If the method is not
410 successfully invoked, the value of the RequestedState property is indeterminate.

411 The CIM_LANEndpoint.RequestedState property shall have one of the values specified in the
412 CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported property or 5 (No Change).

413 **7.2.4.3 CIM_LANEndpoint.EnabledState**

414 When the RequestedState parameter has a value of 2 (Enabled) or 3 (Disabled) and the
415 CIM_LANEndpoint.RequestStateChange() method completes successfully, the value of the EnabledState
416 property shall equal the value of the CIM_LANEndpoint.RequestedState property.

417 If the method does not complete successfully, the value of the EnabledState property is indeterminate.
418 The EnabledState property shall have the value 2 (Enabled) or 3 (Disabled).

419 **7.2.5 Endpoint state management is not supported**

420 This clause describes the CIM elements and behaviors that shall be implemented when management of
421 the endpoint state is not supported.

422 **7.2.5.1 CIM_EnabledLogicalElementCapabilities**

423 When state management is not supported, exactly one instance of
424 CIM_EnabledLogicalElementCapabilities may be associated with the CIM_LANEndpoint instance through
425 an instance of the CIM_ElementCapabilities association.

426 **7.2.5.1.1 CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported**

427 The CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported property shall not contain any
428 values.

429 **7.2.5.2 CIM_LANEndpoint.RequestedState**

430 The RequestedState property shall have the value 12 (Not Applicable).

431 **7.2.5.3 CIM_LANEndpoint.EnabledState**

432 The EnabledState property shall have one of the following values: 2 (Enabled), 3 (Disabled), or 5 (Not
433 Applicable).

434 **7.2.6 Modifying ElementName is supported — conditional**

435 The CIM_LANEndpoint.ElementName property may support being modified by the ModifyInstance
436 operation. See 8.12.2.2. This behavior is conditional. This clause describes the CIM elements and
437 behavior requirements when an implementation supports client modification of the
438 CIM_LANEndpoint.ElementName property.

439 **7.2.6.1 CIM_EnabledLogicalElementCapabilities**

440 An instance of CIM_EnabledLogicalElementCapabilities shall be associated with the CIM_LANEndpoint
441 instance through an instance of CIM_ElementCapabilities.

442 **7.2.6.1.1 CIM_EnabledLogicalElementCapabilities.ElementNameEditSupported**

443 This property shall have a value of TRUE when the implementation supports client modification of the
444 CIM_LANEndpoint.ElementName property.

445 **7.2.6.1.2 CIM_EnabledLogicalElement.MaxElementNameLen**

446 The MaxElementNameLen property shall be implemented.

447 **7.2.7 Modifying ElementName is not supported**

448 This clause describes the CIM elements and behaviors that shall be implemented when the
449 CIM_LANEndpoint.ElementName does not support being modified by the ModifyInstance operation.

450 **7.2.7.1 CIM_EnabledLogicalElementCapabilities**

451 An instance of CIM_EnabledLogicalElementCapabilities may be associated with the CIM_LANEndpoint
452 instance through an instance of CIM_ElementCapabilities.

453 7.2.7.1.1 CIM_EnabledLogicalElementCapabilities.ElementNameEditSupported

454 This property shall have a value of FALSE when the implementation does not support client modification
455 of the CIM_LANEndpoint.ElementName property.

456 7.2.7.1.2 CIM_EnabledLogicalElement.MaxElementNameLen

457 The MaxElementNameLen property may be implemented. The MaxElementNameLen property is
458 irrelevant in this context.

459 7.3 Managing network endpoints

460 An implementation may support the creation and deletion of network endpoints for the network port.

461 When an implementation supports the creation of network endpoints, there shall be an instance of
462 CIM_NetworkPortConfigurationService. An instance of CIM_ServiceAffectsElement is conditional. When
463 an instance of CIM_NetworkPortConfigurationService is instrumented, there shall be an instance of
464 CIM_ServiceAffectsElement that references the Central Instance and the
465 CIM_NetworkPortConfigurationService instance. The CIM_NetworkPortConfigurationService instance
466 shall be associated to an instance of CIM_ComputerSystem through an instance of CIM_HostedService.
467 A network endpoint can be created using the AddLANEndpoint() method of the
468 CIM_NetworkPortConfigurationService, as described in 8.1.

469 An implementation can remove a network endpoint by using the intrinsic DeleteInstance operation
470 defined in 8.12.1.

471 7.4 Representing multiple ports controlled from a single controller

472 In some implementations, a single chip or device provides multiple network interfaces to a system. In
473 other implementations, there is a one-to-one correspondence between the controller component and the
474 actual network interface. An implementation may explicitly instrument the relationship between the
475 controller and interfaces. This behavior is optional. When this optional behavior is supported, the
476 requirements outlined in this clause shall be met.

477 A client can determine if the port controller is modeled by looking for an instance of CIM_PortController
478 that is associated with the Central Instance of this profile through an instance of CIM_ControlledBy.

479 7.4.1 Modeling the controller

480 An instance of CIM_PortController shall represent the controller.

481 7.4.2 Relationship between controller and port

482 For each port controlled by the controller, an instance of CIM_ControlledBy shall associate the instance of
483 CIM_PortController with the instance of CIM_NetworkPort.

484 7.4.3 Controller state management is supported — conditional

485 When management of the state of a port controller is supported, exactly one instance of
486 CIM_EnabledLogicalElementCapabilities shall be associated with the CIM_PortController instance
487 through an instance of CIM_ElementCapabilities.

488 Support for managing the state of the port controller is optional behavior. This clause describes the CIM
489 elements and behaviors that shall be implemented when this behavior is supported.

490 **Conditional determination:** A client can determine whether state management is supported as follows:

- 491 1) Find the CIM_EnabledLogicalElementCapabilities instance that is associated with the
492 CIM_PortController instance.

493 2) Query the value of the RequestedStatesSupported property. If at least one value is specified,
494 state management is supported.

495 **7.4.3.1 CIM_EnabledLogicalElementCapabilities**

496 When state management is supported, exactly one instance of CIM_EnabledLogicalElementCapabilities
497 shall be associated with the CIM_PortController instance through an instance of the
498 CIM_ElementCapabilities association.

499 **7.4.3.1.1 CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported**

500 The RequestedStatesSupported property may contain zero or more of the following values: 2 (Enabled),
501 3 (Disabled), or 11 (Reset).

502 **7.4.3.2 CIM_PortController.RequestedState**

503 When the CIM_PortController.RequestStateChange() method is successfully invoked, the value of the
504 RequestedState property shall be the value of the RequestedState parameter. If the method is not
505 successfully invoked, the value of the RequestedState property is indeterminate.

506 The CIM_PortController.RequestedState property shall have one of the values specified in the
507 CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported property or 5 (No Change).

508 **7.4.3.3 CIM_PortController.EnabledState**

509 When the RequestedState parameter has a value of 2 (Enabled) or 3 (Disabled) and the
510 CIM_PortController.RequestStateChange() method completes successfully, the value of the
511 EnabledState property shall equal the value of the CIM_PortController.RequestedState property.

512 If the method does not complete successfully, the value of the EnabledState property is indeterminate.
513 The EnabledState property shall have the value 2 (Enabled) or 3 (Disabled).

514 **7.4.4 Controller state management is not supported**

515 This clause describes the CIM elements and behaviors that shall be implemented when management of
516 the controller state is not supported.

517 **7.4.4.1 CIM_EnabledLogicalElementCapabilities**

518 When state management is not supported, exactly one instance of
519 CIM_EnabledLogicalElementCapabilities may be associated with the CIM_PortController instance
520 through an instance of the CIM_ElementCapabilities association.

521 **7.4.4.1.1 CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported**

522 The CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported property shall not contain any
523 values.

524 **7.4.4.2 CIM_PortController.RequestedState**

525 The RequestedState property shall have the value 12 (Not Applicable).

526 **7.4.4.3 CIM_PortController.EnabledState**

527 The EnabledState property shall have one of the following values: 2 (Enabled), 3 (Disabled), or 5 (Not
528 Applicable).

529 7.4.5 Modifying ElementName is supported—conditional

530 The CIM_PortController.ElementName property may support being modified by the ModifyInstance
531 operation. See 8.15.1.1. This behavior is conditional. This clause describes the CIM elements and
532 behavior requirements when an implementation supports client modification of the
533 CIM_PortController.ElementName property.

534 **Client determination:** A client can determine whether it can modify the ElementName as follows:

- 535 1) Find the CIM_EnabledLogicalElementCapabilities instance that is associated with the
536 CIM_PortController instance.
- 537 2) Query the value of the ElementNameEditSupported property of the instance. If the value is
538 TRUE, the client can modify the CIM_PortController.ElementName property.

539 7.4.5.1 CIM_EnabledLogicalElementCapabilities

540 An instance of CIM_EnabledLogicalElementCapabilities shall be associated with the CIM_PortController
541 instance through an instance of CIM_ElementCapabilities.

542 7.4.5.1.1 CIM_EnabledLogicalElementCapabilities.ElementNameEditSupported

543 The ElementNameEditSupported property shall have a value of TRUE when the implementation supports
544 client modification of the CIM_PortController.ElementName property.

545 7.4.5.1.2 CIM_EnabledLogicalElement.MaxElementNameLen

546 The MaxElementNameLen property shall be implemented.

547 7.4.6 Modifying ElementName is not supported

548 This clause describes the CIM elements and behaviors that shall be implemented when the
549 CIM_PortController.ElementName does not support being modified by the ModifyInstance operation.

550 7.4.6.1 CIM_EnabledLogicalElementCapabilities

551 An instance of CIM_EnabledLogicalElementCapabilities may be associated with the CIM_PortController
552 instance through an instance of CIM_ElementCapabilities.

553 7.4.6.1.1 CIM_EnabledLogicalElementCapabilities.ElementNameEditSupported

554 The ElementNameEditSupported property shall have a value of FALSE when the implementation does
555 not support client modification of the CIM_PortController.ElementName property.

556 7.4.6.1.2 CIM_EnabledLogicalElement.MaxElementNameLen

557 The MaxElementNameLen property may be implemented. The MaxElementNameLen property is
558 irrelevant in this context.

559 8 Methods

560 This clause details the requirements for supporting intrinsic operations and extrinsic methods for the CIM
561 elements defined by this profile.

562 8.1 CIM_NetworkPortConfigurationService.AddLANEndpoint()

563 The AddLANEndpoint() method is used to create a new endpoint on a network port. This method shall be
564 supported when the CIM_NetworkPortConfigurationService is instrumented. When this method is
565 invoked, the implementation shall attempt to create a new instance of CIM_LANEndpoint. The

566 MACAddress property of the CIM_LANEndpoint instance shall have the value of the Address parameter
567 of the method invocation.

568 When the LANID parameter is specified in the method invocation, the LANID property of the
569 CIM_LANEndpoint instance shall have the value of the LANID parameter. When the LANID parameter is
570 not specified in the method invocation, the LANID property of the CIM_LANEndpoint instance shall have
571 a value of NULL.

572 When the AliasAddresses parameter is specified in the method invocation, the AliasAddresses property
573 of the CIM_LANEndpoint instance shall have the value of the AliasAddresses parameter. When the
574 AliasAddresses parameter is not specified in the method invocation, the AliasAddresses property of the
575 CIM_LANEndpoint instance shall have a value of NULL.

576 When the GroupAddresses parameter is specified in the method invocation, the GroupAddresses
577 property of the CIM_LANEndpoint instance shall have the value of the GroupAddresses parameter. When
578 the GroupAddresses parameter is not specified in the method invocation, the GroupAddresses property
579 of the CIM_LANEndpoint instance shall have a value of NULL.

580 Before creating the instance of CIM_LANEndpoint, the implementation shall verify that the communication
581 endpoint represented by the resultant CIM_LANEndpoint instance is valid for the CIM_NetworkPort
582 instance that is identified by the Port parameter of the method invocation. If the resultant
583 CIM_LANEndpoint represents a valid endpoint for the identified CIM_NetworkPort instance, the
584 implementation shall create the following instances:

- 585 • the instance of CIM_LANEndpoint described in the preceding paragraph
- 586 • an instance of CIM_DeviceSAPImplementation that references the newly created instance of
587 CIM_LANEndpoint and the instance of CIM_NetworkPort that is identified by the Port parameter
588 of the method invocation
- 589 • an instance of CIM_HostedAccessPoint that references the CIM_LANEndpoint and references
590 the instance of CIM_ComputerSystem with which the instance of CIM_NetworkPort that the Port
591 parameter identified is associated through the CIM_SystemDevice association

592 If an implementation is unable to create the three required instances, the implementation shall not create
593 any of the instances and shall return a value of 2 (Error Occurred) as the return code of the method
594 invocation. A method invocation might fail, for example, if a network port supports *N* communication
595 endpoints, *N* communication endpoints are already associated with the network port, and the client
596 attempts to create another endpoint.

597 Detailed requirements of the AddLANEndpoint() method are specified in Table 2 and Table 3.

598 No standard messages are defined.

599 **Table 2 – CIM_NetworkPortConfigurationService.AddLANEndpoint() method: Return code values**

Value	Description
0	Request was successfully executed.
2	Error occurred

600 **Table 3 – CIM_NetworkPortConfigurationService.AddLANEndpoint() method: Parameters**

Qualifiers	Name	Type	Description/Values
IN, REQ	Port	CIM_NetworkPort REF	None
OUT	Endpoint	CIM_LANEndpoint REF	None
IN, REQ	Address	string	None
IN	LANID	string	None
IN	AliasAddresses	string	None
IN	GroupAddresses	string	None

601 **8.2 CIM_NetworkPort.RequestStateChange()**

602 Invocation of the RequestStateChange() method changes the element’s state to the value specified in the
 603 RequestedState parameter. The 2 (Enabled) and 3 (Disabled) values of the RequestedState parameter
 604 shall correspond to enabling or disabling the network interface that the CIM_NetworkPort instance
 605 represents. A value of 11 (Reset) for the RequestedState parameter shall be equivalent to disabling and
 606 then enabling the network interface that the CIM_NetworkPort instance represents.

607 Detailed requirements of the RequestStateChange() method are specified in Table 4 and Table 5.

608 No standard messages are defined.

609 Invoking the RequestStateChange() method multiple times could result in earlier requests being
 610 overwritten or lost.

611 **Table 4 – CIM_NetworkPort.RequestStateChange() method: Return code values**

Value	Description
0	Request was successfully executed
2	Error occurred
0x1000	Job started: REF returned to started CIM_ConcreteJob

612

Table 5 – CIM_NetworkPort.RequestStateChange() method: Parameters

Qualifiers	Name	Type	Description/Values
IN, REQ	RequestedState	uint16	Valid state values: 2 (Enabled) 3 (Disabled) 11 (Reset)
OUT	Job	CIM_ConcreteJob REF	Returned if job started
IN, REQ	TimeoutPeriod	datetime	Client specified maximum amount of time the transition to a new state is supposed to take: 0 or NULL – No time requirements <interval> – Maximum time allowed

613 8.2.1.1 CIM_NetworkPort.RequestStateChange() ConditionalSupport

614 When an instance of CIM_EnabledLogicalElementCapabilities is associated with the CIM_NetworkPort
615 instance and the CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported property contains
616 at least one value, the CIM_NetworkPort.RequestStateChange() method shall be implemented and
617 supported. The CIM_NetworkPort.RequestStateChange() method shall not return a value of 1 (Not
618 Supported).

619 8.3 CIM_LANEndpoint.RequestStateChange()

620 Invocation of the RequestStateChange() method changes the element's state to the value specified in the
621 RequestedState parameter. The 2 (Enabled) and 3 (Disabled) values of the RequestedState parameter
622 will correspond to enabling or disabling the endpoint that the CIM_LANEndpoint instance represents. A
623 value of 11 (Reset) for the RequestedState parameter shall be equivalent to disabling and then enabling
624 the endpoint that the CIM_LANEndpoint instance represents.

625 Detailed requirements of the RequestStateChange() method are specified in Table 6 and Table 7.

626 No standard messages are defined.

627 Invoking the RequestStateChange method multiple times could result in earlier requests being overwritten
628 or lost.

629 **Table 6 – CIM_LANEndpoint.RequestStateChange() method: Return code values**

Value	Description
0	Request was successfully executed
2	Error occurred
0x1000	Job started: REF returned to started CIM_ConcreteJob

630 **Table 7 – CIM_LANEndpoint.RequestStateChange() method: Parameters**

Qualifiers	Name	Type	Description/Values
IN, REQ	RequestedState	uint16	Valid state values: 2 (Enabled) 3 (Disabled) 11 (Reset)
OUT	Job	CIM_ConcreteJob REF	Returned if job started
IN, REQ	TimeoutPeriod	datetime	Client specified maximum amount of time the transition to a new state is supposed to take: 0 or NULL – No time requirements <interval> – Maximum time allowed

631 **8.3.1.1 CIM_LANEndpoint.RequestStateChange() supported**

632 When an instance of CIM_EnabledLogicalElementCapabilities is associated with the CIM_LANEndpoint
633 instance and the CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported property contains
634 at least one value, the CIM_LANEndpoint.RequestStateChange() method shall be implemented and
635 supported. The CIM_LANEndpoint.RequestStateChange() method shall not return a value of 1 (Not
636 Supported).

637 **8.4 CIM_PortController.RequestStateChange()**

638 Invocation of the RequestStateChange() method changes the element’s state to the value specified in the
639 RequestedState parameter. The 2 (Enabled) and 3 (Disabled) values of the RequestedState parameter
640 shall correspond to enabling or disabling the controller that the CIM_PortController instance represents. A
641 value of 11 (Reset) for the RequestedState parameter shall be equivalent to disabling and then enabling
642 the controller that the CIM_PortController instance represents.

643 Detailed requirements of the RequestStateChange() method are specified in Table 8 and Table 9.

644 No standard messages are defined.

645 Invoking the RequestStateChange method multiple times could result in earlier requests being overwritten
646 or lost.

647 **Table 8 – CIM_PortController.RequestStateChange() method: Return code values**

Value	Description
0	Request was successfully executed.
2	Error occurred
0x1000	Job started: REF returned to started CIM_ConcreteJob

648 **Table 9 – CIM_PortController.RequestStateChange() method: Parameters**

Qualifiers	Name	Type	Description/Values
IN, REQ	RequestedState	uint16	Valid state values: 2 (Enabled) 3 (Disabled) 11 (Reset)
OUT	Job	CIM_ConcreteJob REF	Returned if job started
IN, REQ	TimeoutPeriod	datetime	Client specified maximum amount of time the transition to a new state is supposed to take: 0 or NULL – No time requirements <interval> – Maximum time allowed

649 **8.4.1.1 CIM_PortController.RequestStateChange() supported**

650 When an instance of CIM_EnabledLogicalElementCapabilities is associated with the CIM_PortController
651 instance and the CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported property contains
652 at least one value, the CIM_PortController.RequestStateChange() method shall be implemented and
653 supported. The CIM_PortController.RequestStateChange() method shall not return a value of 1 (Not
654 Supported).

655 **8.5 Profile conventions for operations**

656 For each profile class (including associations), the implementation requirements for operations, including
657 those in the following default list, are specified in class-specific subclauses of this clause.

658 The default list of operations is as follows:

- 659 • GetInstance
- 660 • Associators
- 661 • AssociatorNames
- 662 • References
- 663 • ReferenceNames
- 664 • EnumerateInstances
- 665 • EnumerateInstanceNames

666 **8.6 CIM_ControlledBy**

667 Table 10 lists implementation requirements for operations. If implemented, these operations shall be
 668 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 10, all operations
 669 in the default list in 8.5 shall be implemented as defined in [DSP0200](#).

670 NOTE Related profiles may define additional requirements on operations for the profile class.

671 **Table 10 – Operations: CIM_ControlledBy**

Operation	Requirement	Messages
Associators	Unspecified	None
AssociatorNames	Unspecified	None
References	Unspecified	None
ReferenceNames	Unspecified	None

672 **8.7 CIM_DeviceSAPImplementation**

673 Table 11 lists implementation requirements for operations. If implemented, these operations shall be
 674 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 11, all operations
 675 in the default list in 8.5 shall be implemented as defined in [DSP0200](#).

676 NOTE Related profiles may define additional requirements on operations for the profile class.

677 **Table 11 – Operations: CIM_DeviceSAPImplementation**

Operation	Requirement	Messages
Associators	Unspecified	None
AssociatorNames	Unspecified	None
References	Unspecified	None
ReferenceNames	Unspecified	None

678 **8.8 CIM_ElementCapabilities**

679 Table 12 lists implementation requirements for operations. If implemented, these operations shall be
 680 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 12, all operations
 681 in the default list in 8.5 shall be implemented as defined in [DSP0200](#).

682 NOTE Related profiles may define additional requirements on operations for the profile class.

683 **Table 12 – Operations: CIM_ElementCapabilities**

Operation	Requirement	Messages
Associators	Unspecified	None
AssociatorNames	Unspecified	None
References	Unspecified	None
ReferenceNames	Unspecified	None

684 **8.9 CIM_EnabledLogicalElementCapabilities**

685 All operations in the default list in 8.5 shall be implemented as defined in [DSP0200](#).

686 NOTE Related profiles may define additional requirements on operations for the profile class.

687 **8.10 CIM_HostedAccessPoint**688 Table 13, all operations in the default list in 8.5 shall be implemented as defined in [DSP0200](#).

689 NOTE Related profiles may define additional requirements on operations for the profile class.

690 **Table 13 – Operations: CIM_HostedAccessPoint**

Operation	Requirement	Messages
Associators	Unspecified	None
AssociatorNames	Unspecified	None
References	Unspecified	None
ReferenceNames	Unspecified	None

691 **8.11 CIM_HostedService**692 Table 14, all operations in the default list in 8.5 shall be implemented as defined in [DSP0200](#).

693 NOTE Related profiles may define additional requirements on operations for the profile class.

694 **Table 14 – Operations: CIM_HostedService**

Operation	Requirement	Messages
Associators	Unspecified	None
AssociatorNames	Unspecified	None
References	Unspecified	None
ReferenceNames	Unspecified	None

695 **8.12 CIM_LANEndpoint**696 Table 15 lists implementation requirements for operations. If implemented, these operations shall be
697 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 15, all operations
698 in the default list in 8.5 shall be implemented as defined in [DSP0200](#).

699 NOTE Related profiles may define additional requirements on operations for the profile class.

700 **Table 15 – Operations: CIM_LANEndpoint**

Operation	Requirement	Messages
DeleteInstance	Optional. See 8.12.1.	None
ModifyInstance	Optional. See 8.12.2.	None

701 **8.12.1 CIM_LANEndpoint — DeleteInstance**702 An implementation may support the DeleteInstance operation for instances of CIM_LANEndpoint. When
703 the implementation supports the DeleteInstance operation, it may support the operation for some or all of
704 the CIM_LANEndpoint instances implemented. When the DeleteInstance operation is supported for an
705 instance of CIM_LANEndpoint, the implementation shall delete the instance of CIM_LANEndpoint and the
706 instances of CIM_DeviceSAPImplementation and CIM_HostedAccessPoint that reference the
707 CIM_LANEndpoint instance.708 The implementation shall not support the DeleteInstance operation for the CIM_LANEndpoint instance
709 that is identified in 7.2.1.

710 **8.12.2 CIM_LANEndpoint — ModifyInstance**

711 This details the requirements for the ModifyInstance operation that is applied to an instance of
712 CIM_LANEndpoint.

713 **8.12.2.1 CIM_LANEndpoint.MACAddress**

714 The ModifyInstance operation shall not modify the MACAddress property of a CIM_LANEndpoint
715 instance.

716 **8.12.2.2 CIM_LANEndpoint.ElementName**

717 When an instance of CIM_EnabledLogicalElementCapabilities is associated with the CIM_LANEndpoint
718 instance and the CIM_EnabledLogicalElementCapabilities.ElementNameEditSupported property has a
719 value of TRUE, the implementation shall allow the ModifyInstance operation to change the value of the
720 ElementName property of the CIM_LANEndpoint instance. The ModifyInstance operation shall enforce
721 the length restriction specified in the MaxElementNameLen property of the
722 CIM_EnabledLogicalElementCapabilities instance.

723 When an instance of CIM_EnabledLogicalElementCapabilities is not associated with the
724 CIM_LANEndpoint instance, or the ElementNameEditSupported property of the
725 CIM_EnabledLogicalElementCapabilities instance has a value of FALSE, the implementation shall not
726 allow the ModifyInstance operation to change the value of the ElementName property of the
727 CIM_LANEndpoint instance.

728 **8.13 CIM_NetworkPort**

729 Table 16 lists implementation requirements for operations. If implemented, these operations shall be
730 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 16Table 10, all
731 operations in the default list in 8.5 shall be implemented as defined in [DSP0200](#).

732 NOTE Related profiles may define additional requirements on operations for the profile class.

733 **Table 16 – Operations: CIM_NetworkPort**

Operation	Requirement	Messages
ModifyInstance	Optional. See 8.13.1.1.	None

734 **8.13.1 CIM_NetworkPort — ModifyInstance operation**

735 This details the specific requirements for the ModifyInstance operation that is applied to an instance of
736 CIM_NetworkPort.

737 **8.13.1.1 CIM_NetworkPort.ElementName**

738 When an instance of CIM_EnabledLogicalElementCapabilities is associated with the CIM_NetworkPort
739 instance and the CIM_EnabledLogicalElementCapabilities.ElementNameEditSupported property has a
740 value of TRUE, the implementation shall allow the ModifyInstance operation to change the value of the
741 ElementName property of the CIM_NetworkPort instance. The ModifyInstance operation shall enforce the
742 length restriction specified in the MaxElementNameLen property of the
743 CIM_EnabledLogicalElementCapabilities instance.

744 When an instance of CIM_EnabledLogicalElementCapabilities is not associated with the
745 CIM_NetworkPort instance, or the ElementNameEditSupported property of the
746 CIM_EnabledLogicalElementCapabilities instance has a value of FALSE, the implementation shall not
747 allow the ModifyInstance operation to change the value of the ElementName property of the
748 CIM_NetworkPort instance.

749 8.14 CIM_NetworkPortConfigurationService

750 All operations in the default list in 8.5 shall be implemented as defined in [DSP0200](#).

751 NOTE Related profiles may define additional requirements on operations for the profile class.

752 8.15 CIM_PortController

753 Table 17 lists implementation requirements for operations. If implemented, these operations shall be
754 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 17, all operations
755 in the default list in 8.5 shall be implemented as defined in [DSP0200](#).

756 NOTE Related profiles may define additional requirements on operations for the profile class.

757 **Table 17 – Operations: CIM_PortController**

Operation	Requirement	Messages
ModifyInstance	Optional. See 8.15.1.1.	None

758 8.15.1 CIM_PortController — ModifyInstance operation

759 This clause details the specific requirements for the ModifyInstance operation that is applied to an
760 instance of CIM_PortController.

761 8.15.1.1 CIM_PortController.ElementName property

762 When an instance of CIM_EnabledLogicalElementCapabilities is associated with the CIM_PortController
763 instance and the CIM_EnabledLogicalElementCapabilities.ElementNameEditSupported property has a
764 value of TRUE, the implementation shall allow the ModifyInstance operation to change the value of the
765 ElementName property of the CIM_PortController instance. The ModifyInstance operation shall enforce
766 the length restriction specified in the MaxElementNameLen property of the
767 CIM_EnabledLogicalElementCapabilities instance.

768 When an instance of CIM_EnabledLogicalElementCapabilities is not associated with the
769 CIM_PortController instance, or the ElementNameEditSupported property of the
770 CIM_EnabledLogicalElementCapabilities instance has a value of FALSE, the implementation shall not
771 allow the ModifyInstance operation to change the value of the ElementName property of the
772 CIM_PortController instance.

773 8.16 CIM_Realizes

774 Table 19 lists implementation requirements for operations. If implemented, these operations shall be
775 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 19, all operations
776 in the default list in 8.5 shall be implemented as defined in [DSP0200](#).

777 NOTE Related profiles may define additional requirements on operations for the profile class.

778 **Table 18 – Operations: CIM_Realizes**

Operation	Requirement	Messages
Associators	Unspecified	None
AssociatorNames	Unspecified	None
References	Unspecified	None
ReferenceNames	Unspecified	None

779

780 **8.17 CIM_ServiceAffectsElement**

781 Table 19 lists implementation requirements for operations. If implemented, these operations shall be
 782 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 19, all operations
 783 in the default list in 8.5 shall be implemented as defined in [DSP0200](#).

784 NOTE Related profiles may define additional requirements on operations for the profile class.

785 **Table 19 – Operations: CIM_ServiceAffectsElement**

Operation	Requirement	Messages
Associators	Unspecified	None
AssociatorNames	Unspecified	None
References	Unspecified	None
ReferenceNames	Unspecified	None

786 **8.18 CIM_SystemDevice**

787 Table 20 lists implementation requirements for operations. If implemented, these operations shall be
 788 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 20, all operations
 789 in the default list in 8.5 shall be implemented as defined in [DSP0200](#).

790 NOTE Related profiles may define additional requirements on operations for the profile class.

791 **Table 20 – Operations: CIM_SystemDevice**

Operation	Requirement	Messages
Associators	Unspecified	None
AssociatorNames	Unspecified	None
References	Unspecified	None
ReferenceNames	Unspecified	None

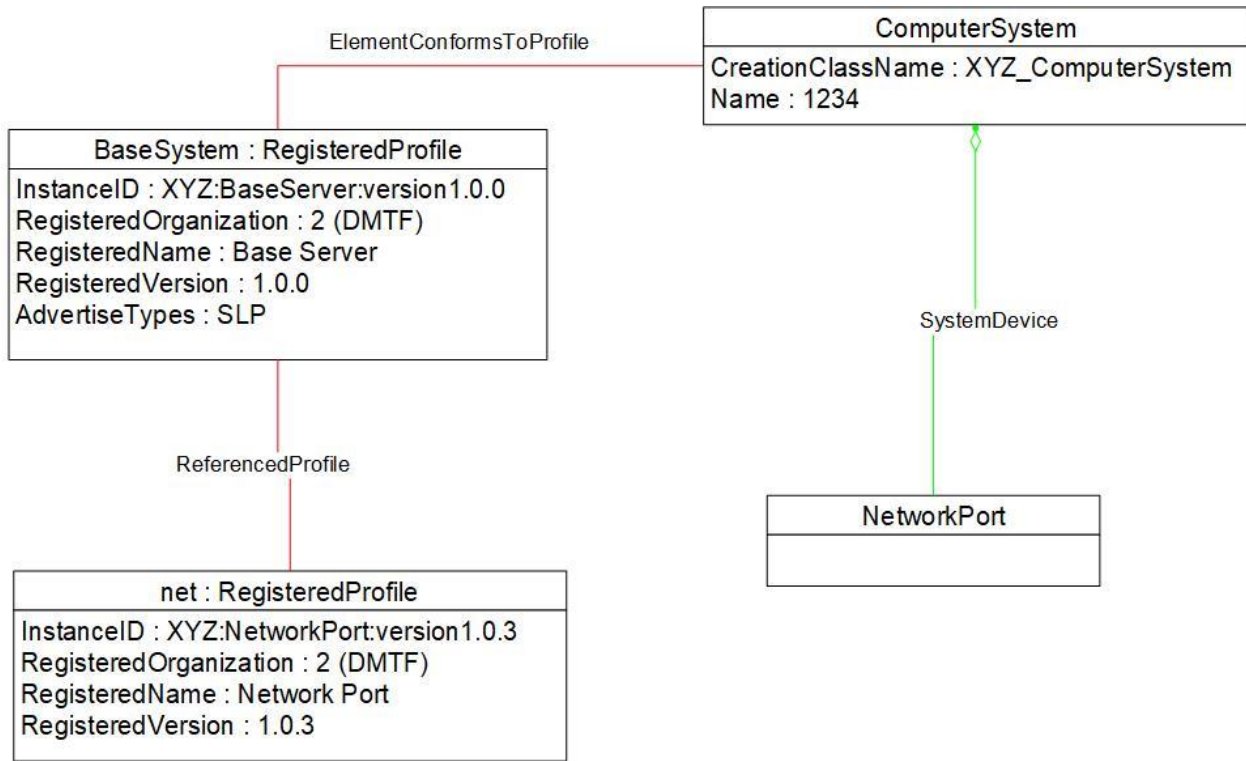
792 **9 Use cases**

793 This clause contains object diagrams and use cases for the *Host LAN Network Port Profile*.

794 **9.1 Object diagrams**

795 The object diagram in Figure 2 shows how instances of CIM_RegisteredProfile are used to identify the
 796 version of the *Host LAN Network Port Profile* with which an instance of CIM_NetworkPort and its
 797 associated instances are conformant. An instance of CIM_RegisteredProfile exists for each profile that is
 798 instrumented in the system. One instance of CIM_RegisteredProfile identifies the *DMTF Base Server*
 799 *Profile*, version 1.0.0. The other instance identifies the *DMTF Network Port Profile*, version 1.0.0.

800 The CIM_NetworkPort instance is scoped to an instance of CIM_ComputerSystem. This instance of
 801 CIM_ComputerSystem is conformant with the *DMTF Base Server Profile*, version 1.0.0 as indicated by
 802 the CIM_ElementConformsToProfile association to the CIM_RegisteredProfile instance. The Scoping
 803 Instance in Figure 2 is the CIM_ComputerSystem instance. The Central Instance is the CIM_NetworkPort.
 804 The CIM_ReferencedProfile relationship between *BaseSystem* and *net* places the CIM_NetworkPort
 805 instance within the scope of *net*. Thus, the CIM_NetworkPort instance is conformant with the *Host LAN*
 806 *Network Port Profile*, version 1.0.0.

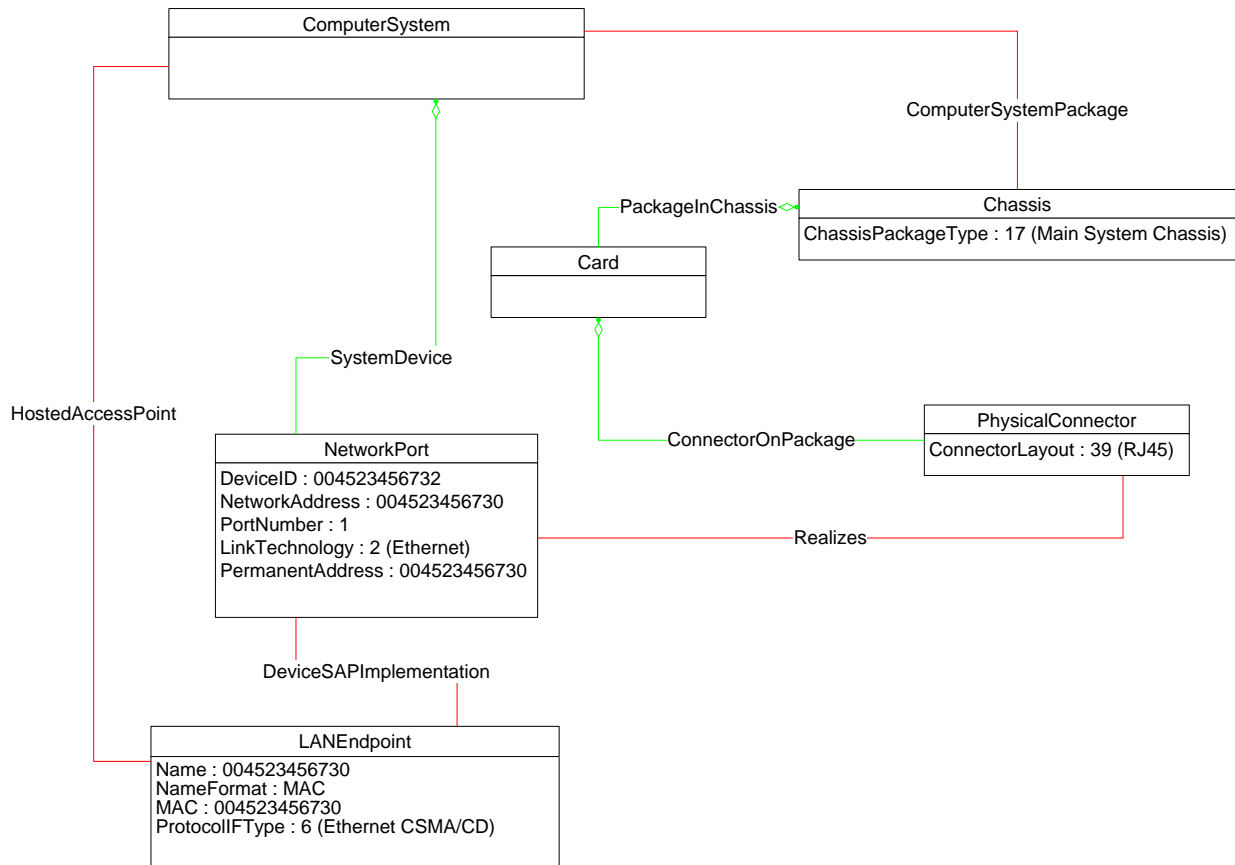


807

808

Figure 2 – Registered Profile

809 Figure 3 is a simple object diagram for a single network port with a single active network interface. The
 810 network port is represented by an instance of CIM_NetworkPort. The active interface is represented by an
 811 instance of CIM_LANEndpoint, which is associated with the CIM_NetworkPort instance through the
 812 CIM_DeviceSAPImplementation association. In the system modeled, the network port is reached through
 813 an RJ-45 connector located directly on the motherboard of the system. This connection is indicated by the
 814 CIM_Realizes association between the CIM_NetworkPort instance and the CIM_PhysicalConnector
 815 instance.

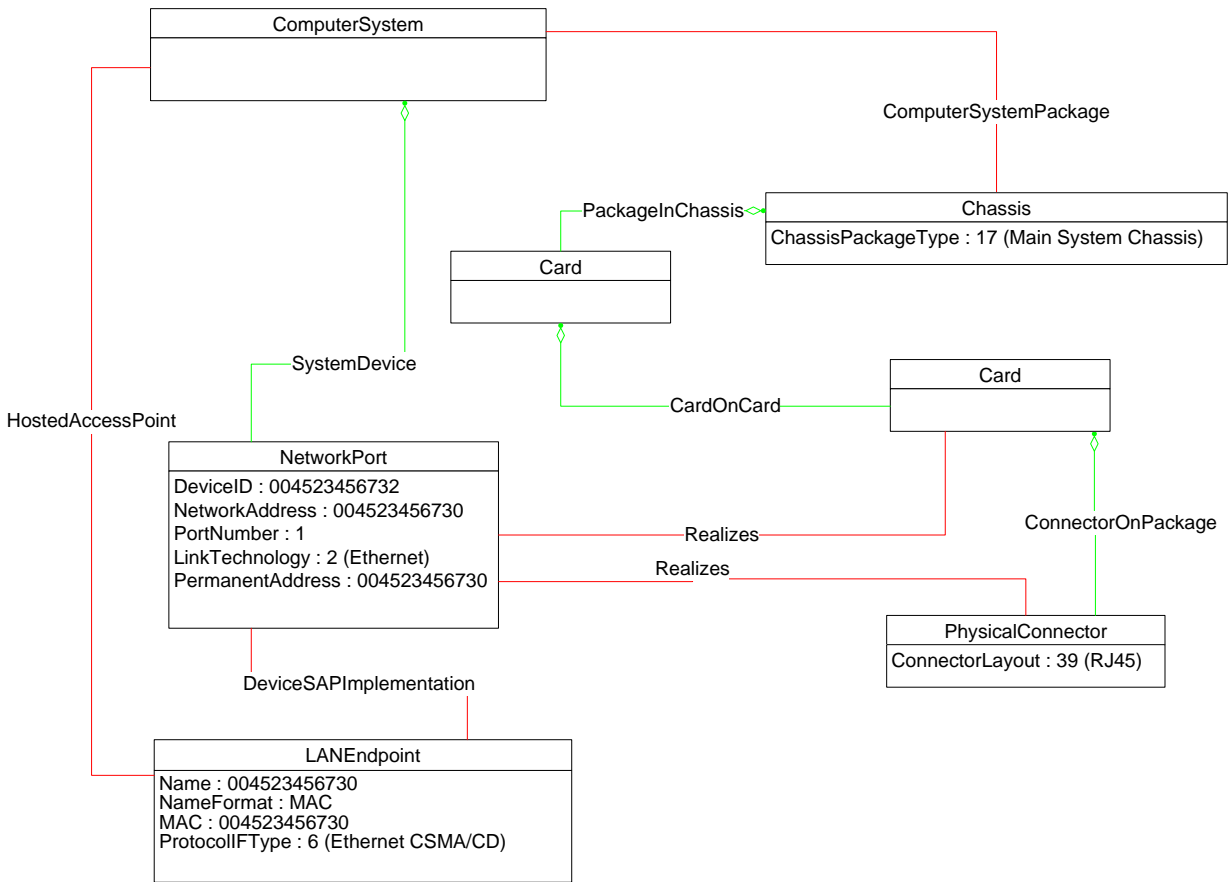


816

817

Figure 3 – Single interface

818 The object diagram in Figure 4 illustrates the classes used to represent a network device located on a
 819 card that is plugged into a system board.

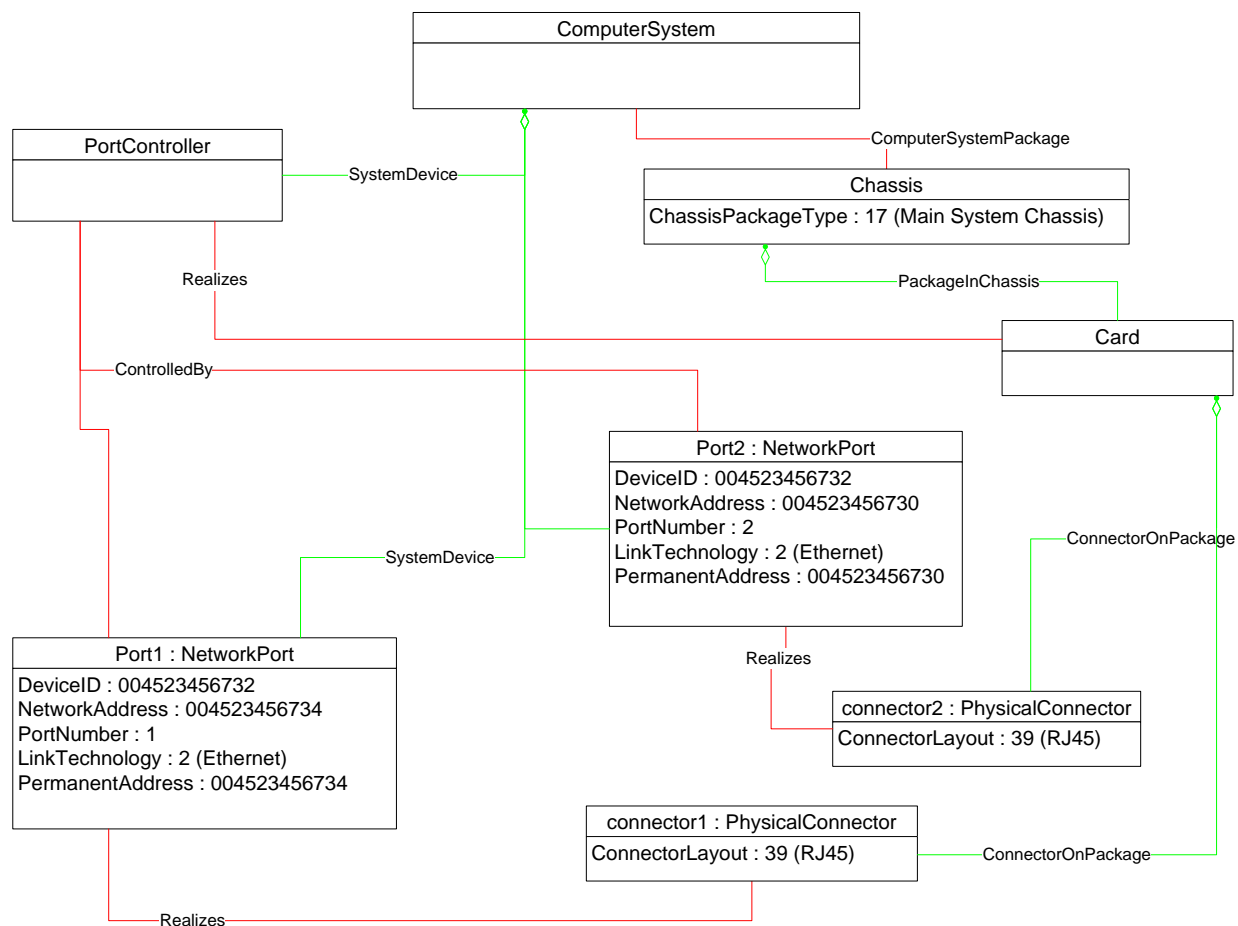


820

821

Figure 4 – Single interface, separate card

822 The object diagram in Figure 5 provides an example of the classes used to represent a single controller
 823 that controls two network ports. The controller is represented by an instance of CIM_PortController. Each
 824 port is represented by an instance of CIM_NetworkPort. The ports being controlled by the port controller
 825 are indicated by the CIM_ControlledBy associations between the CIM_PortController instance and the
 826 CIM_NetworkPort instances. Each port has a single RJ-45 connector associated with it.



827

828 **Figure 5 – One controller for two ports**

829 **9.2 Querying MAC address for an interface**

830 A client can determine the MAC addresses in use for a network interface as follows:

- 831 1) Find all instances of CIM_LANEndpoint that are associated with the CIM_NetworkPort instance
- 832 through instances of CIM_DeviceSAPImplementation.
- 833 2) Query the MACAddress property of each instance of CIM_LANEndpoint.

834 **9.3 Determining physical connector for a network address**

835 One or more MAC addresses may be associated with a given physical network interface. It is useful for a
 836 client to be able to determine which CIM_PhysicalConnector is associated with a given network address.

- 837 1) Find the instance of CIM_NetworkPort that is associated with the CIM_LANEndpoint instance
838 through an instance of CIM_DeviceSAPImplementation.
- 839 2) Find the instance of CIM_PhysicalConnector that is associated with the CIM_NetworkPort
840 instance through an instance of CIM_Realizes.

841 **9.4 Determining whether physical communication is possible**

842 A client can determine whether the physical link for a Network interface is present as follows:

843 Query the value of the CIM_NetworkPort.EnabledState property. If the value of the property is
844 "Enabled but Offline", there is a problem with the underlying physical link.

845 **9.5 Correlating controller and port**

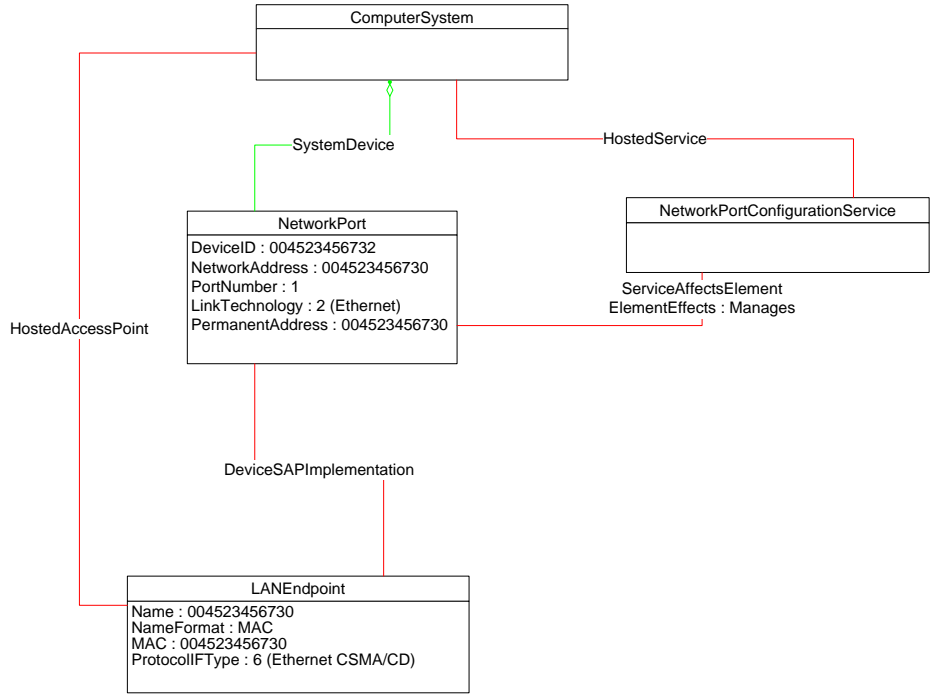
846 Multiple network ports may be controlled by a single controller. A client can determine which controller
847 controls a network port as follows:

848 Find the instance of CIM_PortController that is associated with the CIM_NetworkPort instance
849 through an instance of CIM_ControlledBy.

850 **9.6 Adding an endpoint to the port**

851 Some implementations support creating additional endpoints associated with the network port. A client
852 can determine whether the implementation supports adding endpoints to a port by looking for an instance
853 of CIM_NetworkPortConfigurationService that is associated with the CIM_NetworkPort instance through
854 an instance of CIM_ServiceAffectsElement. The client can then invoke the AddLANEndpoint() method on
855 the CIM_NetworkPortConfigurationService instance, specifying a MAC address, LAN ID, and so on.

856 Figure 6 illustrates an example of a single endpoint associated with the network port. The endpoint
857 corresponds to the real physical address burned into the network port.

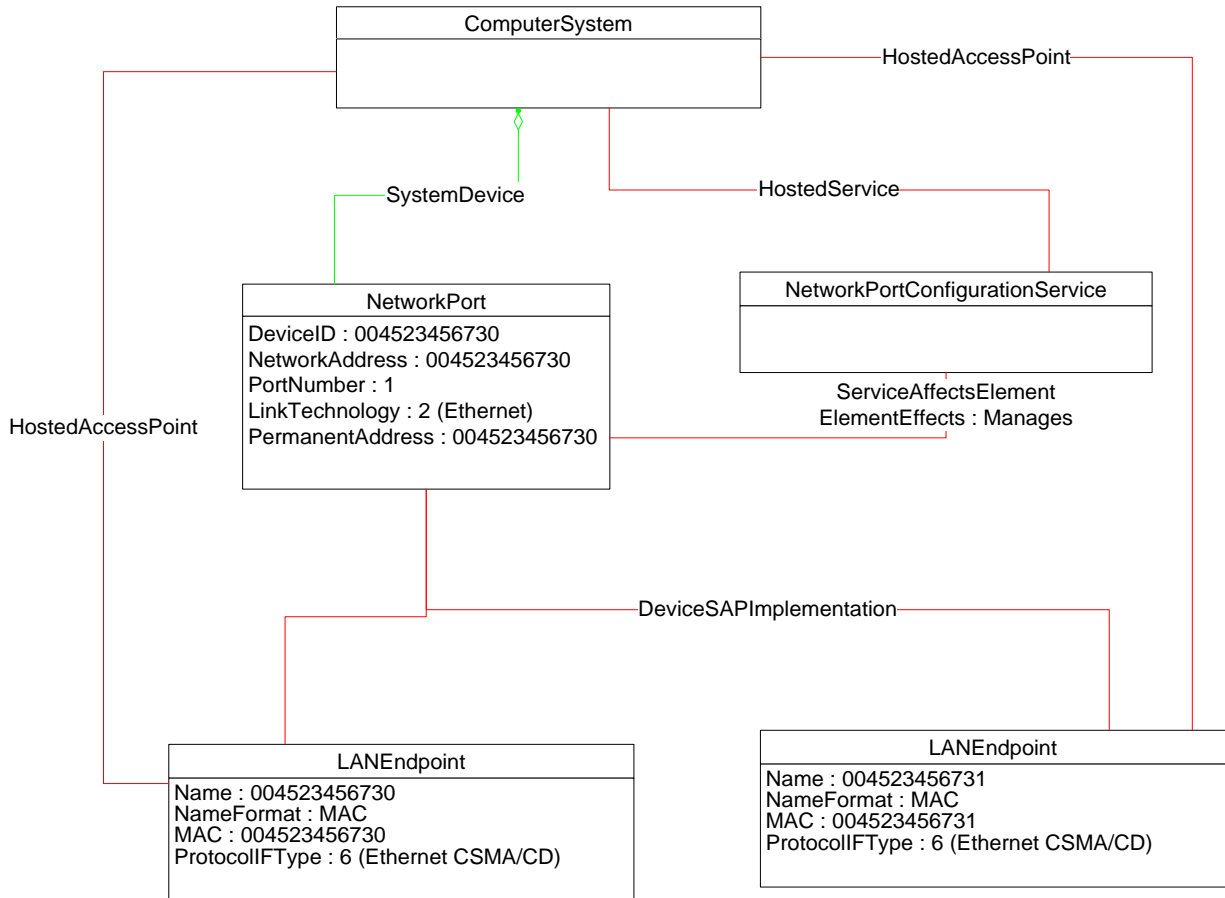


858

859

Figure 6 – Endpoint management supported

860 To add an endpoint to the port in Figure 6, the client invokes the AddLANEndpoint() method and
 861 specifies a value of 004523456731 for the address parameter. Method invocation is successful, and an
 862 additional CIM_LANEndpoint is created and associated with the CIM_NetworkPort instance. This result is
 863 illustrated in Figure 7. Each endpoint is identified by its MAC address property.



864

865

Figure 7 – Second endpoint added

866 **9.7 Determining whether ElementName can be modified**

867 For a given instance of CIM_LANEndpoint, CIM_PortController, or CIM_NetworkPort, a client can
 868 determine whether it can modify the ElementName as follows:

- 869 1) Find the CIM_EnabledLogicalElementCapabilities instance that is associated with the target
 870 instance.
- 871 2) Query the value of the ElementNameEditSupported property of the
 872 CIM_EnabledLogicalElementCapabilities instance. If the value is TRUE, the client can modify
 873 the ElementName property of the target instance.

874 **9.8 Determining whether state management is supported**

875 For a given instance of CIM_LANEndpoint, CIM_PortController, or CIM_NetworkPort, a client can
 876 determine whether state management is supported as follows:

- 877 1) Find the CIM_EnabledLogicalElementCapabilities instance that is associated with the
 878 CIM_LANEndpoint instance.
- 879 2) Query the value of the RequestedStatesSupported property. If at least one value is specified,
 880 state management is supported.

881 **10 CIM Elements**

882 Table 21 shows the instances of CIM Elements for this profile. Instances of the CIM Elements shall be
 883 implemented as described in Table 21. Clauses 7 (“Implementation Requirements”) and 8 (“Methods”)
 884 may impose additional requirements on these elements.

885 **Table 21 – CIM Elements: Network Port Profile**

Element Name	Requirement	Description
Classes		
CIM_ControlledBy	Optional	See 10.1.
CIM_DeviceSAPImplementation	Mandatory	See 10.2.
CIM_ElementCapabilities	Mandatory	See 10.3, 10.4, and 10.5.
CIM_EnabledLogicalElementCapabilities	Optional	See 10.6, 10.7, and 10.8.
CIM_HostedAccessPoint	Mandatory	See 10.9.
CIM_HostedService	Conditional	See 10.10.
CIM_LANEndpoint	Mandatory	See 10.11.
CIM_NetworkPort	Mandatory	See 10.12.
CIM_NetworkPortConfigurationService	Optional	See 10.13.
CIM_PhysicalConnector	Optional	See 10.14 .
CIM_PortController	Optional	See 10.15.
CIM_Realizes	Conditional	See 7.1.6 and 10.16.
CIM_RegisteredProfile	Mandatory	See 10.17.
CIM_ServiceAffectsElement	Conditional	See 7.3 and 10.18.
CIM_SystemDevice	Mandatory	See 10.19 and 10.20.
Indications		
None defined in this profile		

886 **10.1 CIM_ControlledBy**

887 CIM_ControlledBy is used to associate an instance of CIM_NetworkPort with the instance of
 888 CIM_PortController that controls the port, if the port controller is modeled. Table 22 provides information
 889 about the properties of CIM_ControlledBy.

890 **Table 22 – Class: CIM_ControlledBy**

Properties	Requirement	Description
Antecedent	Mandatory	See 7.4.2. Cardinality 0..1
Dependent	Mandatory	See 7.4.2. Cardinality 1..*

891 **10.2 CIM_DeviceSAPImplementation**

892 CIM_DeviceSAPImplementation is used to associate the CIM_LANEndpoint instance with the
 893 CIM_NetworkPort instance that provides the network access. Table 23 provides information about the
 894 properties of CIM_DeviceSAPImplementation.

895 **Table 23 – Class: CIM_DeviceSAPImplementation**

Properties	Requirement	Description
Antecedent	Mandatory	This property shall be an instance of CIM_NetworkPort. Cardinality 1..*
Dependent	Mandatory	This property shall be an instance of CIM_LANEndpoint. Cardinality 1..*

896 **10.3 CIM_ElementCapabilities — LANEndpoint**

897 CIM_ElementCapabilities is used to associate an instance of CIM_EnabledLogicalElementCapabilities
 898 with an instance of CIM_LANEndpoint. Table 24 provides information about the properties of
 899 CIM_ElementCapabilities in this context.

900 **Table 24 – Class: CIM_ElementCapabilities — LANEndpoint**

Properties	Requirement	Description
ManagedElement	Mandatory	Key This property shall be a reference to an instance of CIM_LANEndpoint. Cardinality 1..*
Capabilities	Mandatory	This property shall be a reference to the instance of CIM_EnabledLogicalElementCapabilities. Cardinality 0..1

901 **10.4 CIM_ElementCapabilities — NetworkPort**

902 CIM_ElementCapabilities is used to associate an instance of CIM_EnabledLogicalElementCapabilities
 903 with an instance of CIM_NetworkPort. Table 25 provides information about the properties of
 904 CIM_ElementCapabilities in this context.

905 **Table 25 – Class: CIM_ElementCapabilities — NetworkPort**

Properties	Requirement	Description
ManagedElement	Mandatory	This property shall be a reference to an instance of CIM_NetworkPort. Cardinality 1..*
Capabilities	Mandatory	This property shall be a reference to the instance of CIM_EnabledLogicalElementCapabilities. Cardinality 0..1

906 **10.5 CIM_ElementCapabilities — PortController**

907 CIM_ElementCapabilities is used to associate an instance of CIM_EnabledLogicalElementCapabilities
 908 with an instance of CIM_PortController. Table 26 provides information about the properties of
 909 CIM_ElementCapabilities in this context.

910 **Table 26 – Class: CIM_ElementCapabilities — PortController**

Properties	Requirement	Description
ManagedElement	Mandatory	This property shall be a reference to an instance of CIM_PortController. Cardinality 1..*
Capabilities	Mandatory	This property shall be a reference to the instance of CIM_EnabledLogicalElementCapabilities. Cardinality 0..1

911 **10.6 CIM_EnabledLogicalElementCapabilities — LANEndpoint**

912 CIM_EnabledLogicalElementCapabilities is used to indicate support for managing the state of the network
 913 interface. Table 27 provides information about the properties of CIM_EnabledLogicalElementCapabilities
 914 in this context.

915 **Table 27 – Class: CIM_EnabledLogicalElementCapabilities — LANEndpoint**

Properties	Requirement	Description
InstanceID	Mandatory	None
RequestedStatesSupported	Mandatory	See 7.2.4.1.1 and 7.2.5.1.1.
ElementNameEditSupported	Mandatory	See 7.2.6.1.1 and 7.2.7.1.1.
MaxElementNameLen	Conditional	See 7.2.6.1.2 and 7.2.7.1.2.

916 **10.7 CIM_EnabledLogicalElementCapabilities — NetworkPort**

917 CIM_EnabledLogicalElementCapabilities is used to indicate support for managing the state of the network
 918 port. Table 28 provides information about the properties of CIM_EnabledLogicalElementCapabilities in
 919 this context.

920 **Table 28 – Class: CIM_EnabledLogicalElementCapabilities — NetworkPort**

Properties	Requirement	Description
InstanceID	Mandatory	None
RequestedStatesSupported	Mandatory	See 7.1.2.1.1 and 7.1.3.1.1.
ElementNameEditSupported	Mandatory	See 7.1.4.1.1 and 7.1.5.1.1.
MaxElementNameLen	Conditional	See 7.1.4.1.2 and 7.1.5.1.2.

921 **10.8 CIM_EnabledLogicalElementCapabilities — PortController**

922 CIM_EnabledLogicalElementCapabilities is used to indicate support for managing the state of the port
 923 controller. Table 29 provides information about the properties of CIM_EnabledLogicalElementCapabilities
 924 in this context.

925 **Table 29 – Class: CIM_EnabledLogicalElementCapabilities — PortController**

Properties	Requirement	Description
InstanceID	Mandatory	None
RequestedStatesSupported	Mandatory	See 7.4.3.1.1 and 7.4.4.1.1.
ElementNameEditSupported	Mandatory	See 7.4.5.1.1 and 7.4.6.1.1.
MaxElementNameLen	Conditional	See 7.4.5.1.2 and 7.4.6.1.2.

926 **10.9 CIM_HostedAccessPoint**

927 CIM_HostedAccessPoint is used to relate a CIM_LANEndpoint instance to its scoping
 928 CIM_ComputerSystem instance. Table 30 provides information about the properties of
 929 CIM_HostedAccessPoint.

930 **Table 30 – Class: CIM_HostedAccessPoint**

Properties	Requirement	Description
Antecedent	Mandatory	This property shall be a reference to the Scoping Instance. Cardinality 1
Dependent	Mandatory	This property shall be a reference to an instance of CIM_LANEndpoint. Cardinality 1..*

931 **10.10 CIM_HostedService**

932 CIM_HostedService is used to associate the CIM_NetworkPortConfigurationService instance with the
 933 CIM_ComputerSystem instance to which it is scoped. Table 31 provides information about the properties
 934 of CIM_HostedService.

935 **Table 31 – Class: CIM_HostedService**

Properties	Requirement	Description
Antecedent	Mandatory	This property shall be a reference to the Scoping Instance. Cardinality 1
Dependent	Mandatory	This property shall be a reference to CIM_NetworkPortConfigurationService. Cardinality *

936 **10.11 CIM_LANEndpoint**

937 CIM_LANEndpoint represents a MAC address to which the network port will respond on the LAN. Table
 938 32 provides information about the properties of CIM_LANEndpoint.

939 **Table 32 – Class: CIM_LANEndpoint**

Properties and Methods	Requirement	Description
SystemCreationClassName	Mandatory	None
CreationClassName	Mandatory	None
SystemName	Mandatory	None
Name	Mandatory	None
NameFormat	Mandatory	None
ProtocolIFType	Mandatory	None
MACAddress	Mandatory	None
LANID	Optional	See 8.1.
AliasAddresses	Optional	See 8.1.
GroupAddresses	Optional	See 8.1.
RequestedState	Mandatory	See 7.2.4.2 and 7.2.5.2.
EnabledState	Mandatory	See 7.2.5.3 and 7.2.4.3.
ElementName	Mandatory	See 7.2.6 and 7.2.7.
RequestStateChange()	Conditional	See 8.3.

940 **10.12 CIM_NetworkPort**

941 CIM_NetworkPort represents the hardware and device aspects of a physical network interface. Table 33
 942 provides information about the properties of CIM_NetworkPort.

943 **Table 33 – Class: CIM_NetworkPort**

Properties and Methods	Requirement	Description
SystemCreationClassName	Mandatory	None
CreationClassName	Mandatory	None
SystemName	Mandatory	None
Name	Mandatory	None
Speed	Optional	A value of 0 (zero) shall indicate that the actual value is unknown.
LinkTechnology	Mandatory	None
PermanentAddress	Mandatory	This property shall be a character string of length 0 to 64. pattern.{0,64}
MaxSpeed	Optional	A value of 0 (zero) shall indicate that the actual value is unknown.
RequestedSpeed	Optional	A value of 0 (zero) shall indicate that the actual value is unknown.
DeviceID	Mandatory	None
EnabledState	Mandatory	See 7.1.2.3 and 7.1.3.3.
RequestedState	Mandatory	See 7.1.2.2 and 7.1.3.2.
ElementName	Mandatory	See 7.1.4 and 7.1.5.
RequestStateChange()	Conditional	See 8.2.

944 **10.13 CIM_NetworkPortConfigurationService**

945 CIM_NetworkPortConfigurationService represents the ability to add endpoints to the network port. Table
 946 34 provides information about the properties of CIM_NetworkPortConfigurationService.

947 **Table 34 – Class: NetworkPortConfigurationService**

Properties and Methods	Requirement	Description
SystemCreationClassName	Mandatory	None
CreationClassName	Mandatory	None
SystemName	Mandatory	None
Name	Mandatory	None
ElementName	Mandatory	This property shall be formatted as a free-form string of variable length. (pattern “.*”)
AddLANEndpoint()	Mandatory	See 8.1.

948 **10.14 CIM_PhysicalConnector**

949 CIM_PhysicalConnector is used to represent the physical connector that connects the network port to the
 950 physical network. This class is defined by [DSP1011](#). The behavior specified in Table 35 is in addition to
 951 that specified by [DSP1011](#).

952 **Table 35 – Class: CIM_PhysicalConnector**

Properties	Requirement	Description
ConnectorLayout	Mandatory	None

953 **10.15 CIM_PortController**

954 CIM_PortController represents a network controller. Table 36 provides information about the properties of
 955 CIM_PortController.

956 **Table 36 – Class: CIM_PortController**

Properties	Requirement	Description
ControllerType	Mandatory	None
ProtocolSupported	Mandatory	None
MaxNumberControlled	Mandatory	A value of 0 (zero) shall indicate that the actual value is unknown.
SystemCreationClassName	Mandatory	None
SystemName	Mandatory	None
CreationClassName	Mandatory	None
Name	Mandatory	None
DeviceID	Mandatory	None
EnabledState	Mandatory	See 7.4.3.3.
RequestedState	Mandatory	See 7.4.3.2.
ElementName	Mandatory	See 7.4.5 and 7.4.6.
RequestStateChange()	Conditional	See 8.4.

957 **10.16 CIM_Realizes**

958 The CIM_Realizes association is used to associate the CIM_NetworkPort with an instance of
 959 CIM_PhysicalConnector when an instance of CIM_PhysicalConnector is instrumented. This class is
 960 defined by [DSP1011](#). The behavior specified in Table 37 is in addition to that specified by [DSP1011](#).

961 **Table 37 – Class: CIM_Realizes**

Properties	Requirement	Description
Antecedent	Mandatory	This property shall be a reference to CIM_PhysicalConnector. Cardinality 0..1
Dependent	Mandatory	This property shall be a reference to the Central Instance. Cardinality 1..*

962 **10.17 CIM_RegisteredProfile**

963 CIM_RegisteredProfile identifies the *Host LAN Network Port Profile* in order for a client to determine
 964 whether an instance of CIM_LogicalModule is conformant with this profile. The CIM_RegisteredProfile
 965 class is defined by [DSP1033](#). With the exception of the mandatory values specified for the properties in
 966 Table 38, the behavior of the CIM_RegisteredProfile instance is in accordance with the constraints
 967 specified in [DSP1033](#).

968 **Table 38 – Class: CIM_RegisteredProfile**

Properties	Requirement	Description
RegisteredName	Mandatory	This property shall have a value of "Host LAN Network Port".
RegisteredVersion	Mandatory	This property shall have a value of "1.0.3".
RegisteredOrganization	Mandatory	This property shall have a value of 2(DMTF).

969 NOTE Previous versions of this document included the suffix "Profile" for the RegisteredName value. If
 970 implementations querying for the RegisteredName value find the suffix "Profile", they should ignore the
 971 suffix, with any surrounding white spaces, before any comparison is done with the value as specified in this
 972 document.

973 **10.18 CIM_ServiceAffectsElement**

974 CIM_ServiceAffectsElement is used to associate an instance of CIM_NetworkPortConfigurationService
 975 with an instance of CIM_NetworkPort that the service is able to configure. Table 39 provides information
 976 about the properties of CIM_ServiceAffectsElement.

977 **Table 39 – Class: CIM_ServiceAffectsElement**

Properties	Requirement	Description
AffectingElement	Mandatory	This property shall be a reference to the instance of CIM_NetworkPortConfigurationService. Cardinality *
AffectedElement	Mandatory	This property shall be a reference to an instance of CIM_NetworkPort. Cardinality 1..*
ElementAffects	Mandatory	Matches 5 (Manages)

978 **10.19 CIM_SystemDevice — CIM_NetworkPort**

979 CIM_SystemDevice is used to associate an instance of CIM_NetworkPort with the instance of
 980 CIM_ComputerSystem to which the CIM_NetworkPort is scoped. Table 40 provides information about the
 981 properties of CIM_SystemDevice.

982 **Table 40 – Class: CIM_SystemDevice**

Properties	Requirement	Description
GroupComponent	Mandatory	This property shall be a reference to an instance of CIM_ComputerSystem. Cardinality 1
PartComponent	Mandatory	This property shall be a reference to CIM_NetworkPort. Cardinality 1..*

983 **10.20 CIM_SystemDevice — CIM_PortController**

984 CIM_SystemDevice is used to associate an instance of CIM_PortController with an instance of
 985 CIM_ComputerSystem when CIM_PortController is implemented. Table 41 provides information about the
 986 properties of CIM_SystemDevice.

987 **Table 41 – Class: CIM_SystemDevice**

Properties	Requirement	Description
GroupComponent	Mandatory	This property shall be a reference to an instance of CIM_ComputerSystem. Cardinality 1
PartComponent	Mandatory	This property shall be a reference to CIM_PortController. Cardinality *

988

989
990
991
992**ANNEX A
(informative)****Change log**

Version	Date	Description
1.0.0	2008-06-03	
1.0.1	2010-09-15	DMTF Standard release
1.0.2	2011-04-07	Errata version. Added operations tables in clause 8 for CIM_DeviceSAPImplementation and CIM_Realizes.
1.0.3	2019-03-18	This errata addresses these issues: <ul style="list-style-type: none">• Updated RegisteredVersion to reflect errata version number in clause 10.17• Updated RegisteredOrganization description to reflect correct value of 2 for DMTF in clause 10.17 and figures in clause 9.1

993