



1
2 **Document Number: DSP1035**
3 **Date: 2011-06-30**
4 **Version: 1.0.2**

5 **Host LAN Network Port Profile**

- 6 **Document Type:** Specification
7 **Document Status:** DMTF Standard
8 **Document Language:** en-US

9 Copyright Notice

10 Copyright © 2008-2011 Distributed Management Task Force, Inc. (DMTF). All rights reserved.

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

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25 withdrawn or modified after publication, and shall be indemnified and held harmless by any party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

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

CONTENTS

32	Foreword	6
33	Introduction	7
34	1 Scope	9
35	2 Normative References.....	9
36	3 Terms and Definitions	9
37	4 Symbols and Abbreviated Terms	10
38	5 Synopsis	11
39	6 Description	11
40	7 Implementation Requirements	12
41	7.1 Representing a Network Port.....	12
42	7.2 Representing a Communication Endpoint	15
43	7.3 Managing Network Endpoints.....	17
44	7.4 Representing Multiple Ports Controlled from a Single Controller	17
45	8 Methods.....	20
46	8.1 CIM_NetworkPortConfigurationService.AddLANEndpoint().....	20
47	8.2 CIM_NetworkPort.RequestStateChange().....	21
48	8.3 CIM_LANEndpoint.RequestStateChange()	22
49	8.4 CIM_PortController.RequestStateChange()	23
50	8.5 Profile Conventions for Operations.....	24
51	8.6 CIM_ControlledBy.....	25
52	8.7 CIM_DeviceSAPIImplementation	25
53	8.8 CIM_ElementCapabilities	25
54	8.9 CIM_EnabledLogicalElementCapabilities.....	25
55	8.10 CIM_HostedAccessPoint	26
56	8.11 CIM_HostedService	26
57	8.12 CIM_LANEndpoint	26
58	8.13 CIM_NetworkPort.....	27
59	8.14 CIM_NetworkPortConfigurationService	28
60	8.15 CIM_PortController	28
61	8.16 CIM_Realizes.....	29
62	8.17 CIM_ServiceAffectsElement	29
63	8.18 CIM_SystemDevice	29
64	9 Use Cases	30
65	9.1 Object Diagrams	30
66	9.2 Querying MAC Address for an Interface	33
67	9.3 Determining Physical Connector for a Network Address.....	33
68	9.4 Determining If Physical Communication Is Possible	34
69	9.5 Correlating Controller and Port	34
70	9.6 Adding an Endpoint to the Port.....	34
71	9.7 Determining If ElementName Can Be Modified.....	35
72	9.8 Determining If State Management Is Supported	36
73	10 CIM Elements.....	36
74	10.1 CIM_ControlledBy	36
75	10.2 CIM_DeviceSAPIImplementation	37
76	10.3 CIM_ElementCapabilities — LANEndpoint	37
77	10.4 CIM_ElementCapabilities — NetworkPort	37
78	10.5 CIM_ElementCapabilities — PortController	38
79	10.6 CIM_EnabledLogicalElementCapabilities — LANEndpoint.....	38
80	10.7 CIM_EnabledLogicalElementCapabilities — NetworkPort	38
81	10.8 CIM_EnabledLogicalElementCapabilities — PortController.....	39
82	10.9 CIM_HostedAccessPoint	39
83	10.10 CIM_HostedService	39

84	10.11 CIM_LANEndpoint	40
85	10.12 CIM_NetworkPort.....	40
86	10.13 CIM_NetworkPortConfigurationService.....	41
87	10.14 CIM_PhysicalConnector	41
88	10.15 CIM_PortController	42
89	10.16 CIM_Realizes.....	42
90	10.17 CIM_RegisteredProfile.....	43
91	10.18 CIM_ServiceAffectsElement	43
92	10.19 CIM_SystemDevice — CIM_NetworkPort	43
93	10.20 CIM_SystemDevice — CIM_PortController.....	44
94	ANNEX A (informative) Change Log.....	45
95		

96 Figures

97	Figure 1 – Host LAN Network Port Profile: Class Diagram.....	12
98	Figure 2 – Registered Profile	30
99	Figure 3 – Single Interface	31
100	Figure 4 – Single Interface, Separate Card	32
101	Figure 5 – One Controller for Two Ports	33
102	Figure 6 – Endpoint Management Supported	34
103	Figure 7 – Second Endpoint Added	35
104		

105 Tables

106	Table 1 – Referenced Profiles	11
107	Table 2 – CIM_NetworkPortConfigurationService.AddLANEndpoint() Method: Return Code Values	21
108	Table 3 – CIM_NetworkPortConfigurationService.AddLANEndpoint() Method: Parameters	21
109	Table 4 – CIM_NetworkPort.RequestStateChange() Method: Return Code Values	21
110	Table 5 – CIM_NetworkPort.RequestStateChange() Method: Parameters.....	22
111	Table 6 – CIM_LANEndpoint.RequestStateChange() Method: Return Code Values.....	23
112	Table 7 – CIM_LANEndpoint.RequestStateChange() Method: Parameters	23
113	Table 8 – CIM_PortController.RequestStateChange() Method: Return Code Values.....	24
114	Table 9 – CIM_PortController.RequestStateChange() Method: Parameters	24
115	Table 10 – Operations: CIM_ControlledBy	25
116	Table 11 – Operations: CIM_DeviceSAPIImplementation	25
117	Table 12 – Operations: CIM_ElementCapabilities	25
118	Table 13 – Operations: CIM_HostedAccessPoint	26
119	Table 14 – Operations: CIM_HostedService	26
120	Table 15 – Operations: CIM_LANEndpoint.....	26
121	Table 16 – Operations: CIM_NetworkPort.....	27
122	Table 17 – Operations: CIM_PortController.....	28
123	Table 18 – Operations: CIM_Realizes	29
124	Table 19 – Operations: CIM_ServiceAffectsElement	29
125	Table 20 – Operations: CIM_SystemDevice.....	29
126	Table 21 – CIM Elements: Network Port Profile	36
127	Table 22 – Class: CIM_ControlledBy.....	36

128	Table 23 – Class: CIM_DeviceSAPIImplementation	37
129	Table 24 – Class: CIM_ElementCapabilities — LANEndpoint.....	37
130	Table 25 – Class: CIM_ElementCapabilities — NetworkPort.....	37
131	Table 26 – Class: CIM_ElementCapabilities — PortController.....	38
132	Table 27 – Class: CIM_EnabledLogicalElementCapabilities — LANEndpoint.....	38
133	Table 28 – Class: CIM_EnabledLogicalElementCapabilities — NetworkPort	38
134	Table 29 – Class: CIM_EnabledLogicalElementCapabilities — PortController.....	39
135	Table 30 – Class: CIM_HostedAccessPoint	39
136	Table 31 – Class: CIM_HostedService	39
137	Table 32 – Class: CIM_LANEndpoint	40
138	Table 33 – Class: CIM_NetworkPort.....	40
139	Table 34 – Class: NetworkPortConfigurationService.....	41
140	Table 35 – Class: CIM_PhysicalConnector	41
141	Table 36 – Class: CIM_PortController	42
142	Table 37 – Class: CIM_Realizes.....	42
143	Table 38 – Class: CIM_RegisteredProfile.....	43
144	Table 39 – Class: CIM_ServiceAffectsElement	43
145	Table 40 – Class: CIM_SystemDevice	43
146	Table 41 – Class: CIM_SystemDevice	44
147		

148

Foreword

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

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

153 Acknowledgments

154 The authors wish to acknowledge the following people.

155 Editors:

156 • Hemal Shah – Broadcom

157 • Jeff Hilland – HP

158 • Aaron Merkin – IBM

159 Contributors:

160 • Hemal Shah – Broadcom

161 • Jon Hass – Dell

162 • Khachatur Papanyan – Dell

163 • Enoch Suen – Dell

164 • Jeff Hilland – HP

165 • Christina Shaw – HP

166 • Aaron Merkin – IBM

167 • Perry Vincent – Intel

168 • John Leung – Intel

169

Introduction

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

176

Host LAN Network Port Profile

177

1 Scope

178
179
180
181

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

182

2 Normative References

183
184
185

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

186
187

DMTF DSP0004, *CIM Infrastructure Specification 2.6*,
http://www.dmtf.org/standards/published_documents/DSP0004_2.6.pdf

188
189

DMTF DSP0200, *CIM Operations over HTTP 1.3*,
http://www.dmtf.org/standards/published_documents/DSP0200_1.3.pdf

190
191

DMTF DSP1001, *Management Profile Specification Usage Guide 1.0*,
http://www.dmtf.org/standards/published_documents/DSP1001_1.0.pdf

192
193

DMTF DSP1011, *Physical Asset Profile 1.0*,
http://www.dmtf.org/standards/published_documents/DSP1011_1.0.pdf

194
195

DMTF DSP1033, *Profile Registration Profile 1.0*,
http://www.dmtf.org/standards/published_documents/DSP1033_1.0.pdf

196
197

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

198

3 Terms and Definitions

199

For the purposes of this document, the following terms and definitions apply.

200

1.

201

can

202

used for statements of possibility and capability, whether material, physical, or causal

203

2.

204

cannot

205

used for statements of possibility and capability, whether material, physical, or causal

206

3.

207

conditional

208
209

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

210 **4.**
211 **mandatory**
212 indicates requirements to be followed strictly in order to conform to the document and from which no
213 deviation is permitted

214 **5.**
215 **may**
216 indicates a course of action permissible within the limits of the document

217 **6.**
218 **need not**
219 indicates a course of action permissible within the limits of the document

220 **7.**
221 **optional**
222 indicates a course of action permissible within the limits of the document

223 **8.**
224 **referencing profile**
225 indicates a profile that owns the definition of this class and can include a reference to this profile in its
226 "Related Profiles" table

227 **9.**
228 **shall**
229 indicates requirements to be followed strictly in order to conform to the document and from which no
230 deviation is permitted

231 **10.**
232 **shall not**
233 indicates requirements to be followed strictly in order to conform to the document and from which no
234 deviation is permitted

235 **11.**
236 **should**
237 indicates that among several possibilities, one is recommended as particularly suitable, without
238 mentioning or excluding others, or that a certain course of action is preferred but not necessarily required

239 **12.**
240 **should not**
241 indicates that a certain possibility or course of action is deprecated but not prohibited

242 **4 Symbols and Abbreviated Terms**

243 The following symbols and abbreviations are used in this document.

244 **4.1**
245 **DNS**
246 Domain Name System

247 **4.2**
248 **DHCP**
249 Dynamic Host Configuration Protocol

250 **4.3**
251 **LAN**
252 Local Area Network

253 **5 Synopsis**

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

255 **Version:** 1.0.2

256 **Organization:** DMTF

257 **CIM Schema version:** 2.22

258 **Central Class:** CIM_NetworkPort

259 **Scoping Class:** CIM_ComputerSystem

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

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

266 CIM_NetworkPort shall be the Central Class of this profile. The instance of CIM_NetworkPort shall be the
267 Central Instance of this profile. CIM_ComputerSystem shall be the Scoping Class of this profile. The
268 instance of CIM_ComputerSystem with which the Central Instance is associated through an instance of
269 CIM_SystemDevice shall be the Scoping Instance of this profile.

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

271 **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.

272 **6 Description**

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

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

- 276 • a specification of the network port and related hardware
277 • network interfaces active over the network port

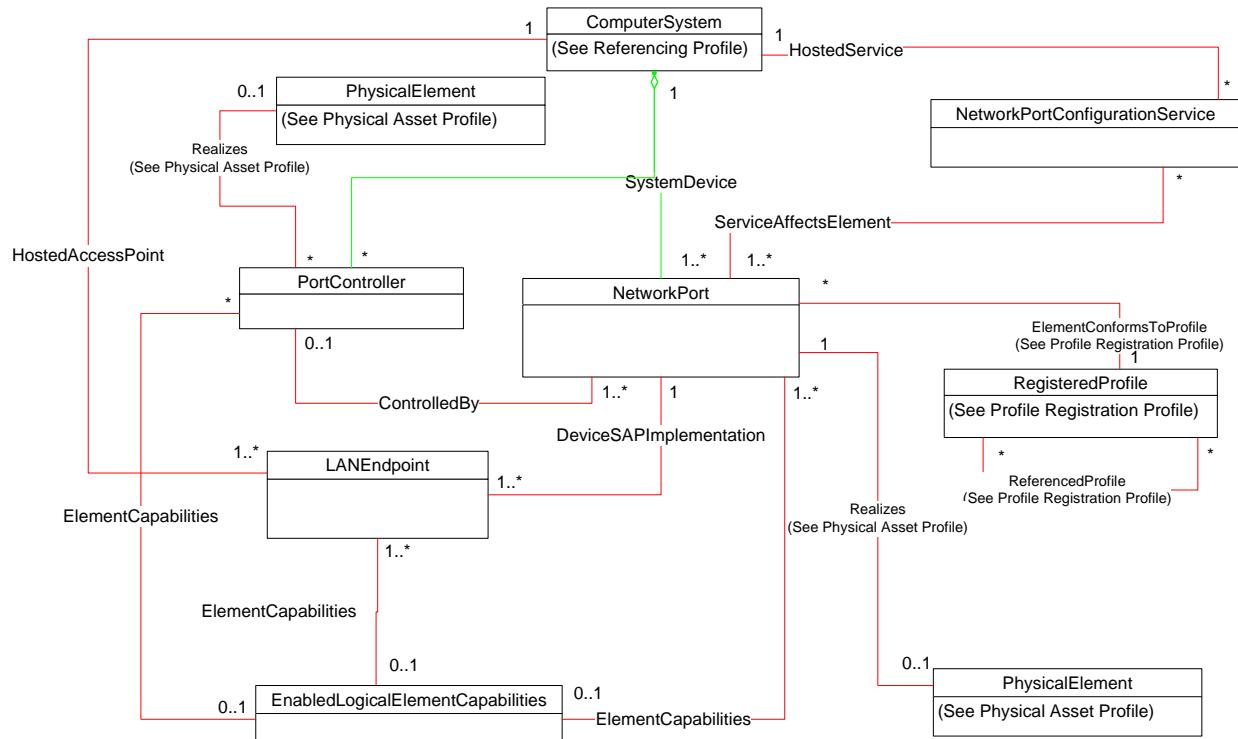
278 The following functionality is optional within the scope of this profile:

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

280 The following functionality is not covered in this profile:

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

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



291

292

Figure 1 – Host LAN Network Port Profile: Class Diagram

293 7 Implementation Requirements

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

296 7.1 Representing a Network Port

297 An instance of CIM_NetworkPort shall represent the network port.

298 7.1.1 CIM_NetworkPort.EnabledState — Enabled but Offline

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

302 **7.1.2 Network Port State Management Is Supported — Conditional**

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

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

308 **Conditional Determination:** A client can determine whether state management is supported as follows:

- 309 1) Find the CIM_EnabledLogicalElementCapabilities instance that is associated with the
310 CIM_NetworkPort instance.
- 311 2) Query the value of the RequestedStatesSupported property. If at least one value is specified,
312 state management is supported.

313 **7.1.2.1 CIM_EnabledLogicalElementCapabilities**

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

317 **7.1.2.1.1 CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported**

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

320 **7.1.2.2 CIM_NetworkPort.RequestedState**

321 When the CIM_NetworkPort.RequestStateChange() method is successfully invoked, the value of the
322 RequestedState property shall be the value of the RequestedState parameter. If the method is not
323 successfully invoked, the value of the RequestedState property is indeterminate.

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

326 **7.1.2.3 CIM_NetworkPort.EnabledState**

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

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

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

332 **7.1.3 Network Port State Management Is Not Supported**

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

335 **7.1.3.1 CIM_EnabledLogicalElementCapabilities**

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

340 **7.1.3.1.1 CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported**

341 The CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported property shall not contain any
342 values.

343 **7.1.3.2 CIM_NetworkPort.RequestedState**

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

345 **7.1.3.3 CIM_NetworkPort.EnabledState**

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

348 **7.1.4 Modifying ElementName Is Supported — Conditional**

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

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

- 354 1) Find the CIM_EnabledLogicalElementCapabilities instance that is associated with the
355 CIM_NetworkPort instance.
- 356 2) Query the value of the ElementNameEditSupported property of the instance. If the value is
357 TRUE, the client can modify the CIM_NetworkPort.ElementName property.

358 **7.1.4.1 CIM_EnabledLogicalElementCapabilities**

359 An instance of CIM_EnabledLogicalElementCapabilities shall be associated with the CIM_NetworkPort
360 instance through an instance of CIM_ElementCapabilities.

361 **7.1.4.1.1 CIM_EnabledLogicalElementCapabilities.ElementNameEditSupported**

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

364 **7.1.4.1.2 CIM_EnabledLogicalElement.MaxElementNameLen**

365 The MaxElementNameLen property shall be implemented.

366 **7.1.5 Modifying ElementName Is Not Supported**

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

369 **7.1.5.1 CIM_EnabledLogicalElementCapabilities**

370 An instance of CIM_EnabledLogicalElementCapabilities may be associated with the CIM_NetworkPort
371 instance through an instance of CIM_ElementCapabilities.

372 **7.1.5.1.1 CIM_EnabledLogicalElementCapabilities.ElementNameEditSupported**

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

375 **7.1.5.1.2 CIM_EnabledLogicalElement.MaxElementNameLen**

376 The MaxElementNameLen property may be implemented. The MaxElementNameLen property is
377 irrelevant in this context.

378 **7.1.6 Representing the Physical Packaging**

379 Support for representing the physical packaging of the network device is optional. The physical packaging
380 may be modeled using one or more instances of CIM_PhysicalElement in accordance with the [Physical
381 Asset Profile](#).

382 In addition, an implementation may use an instance of CIM_PhysicalConnector to represent the physical
383 connector. When an implementation instruments an instance of CIM_PhysicalConnector to represent the
384 physical connector of the network device for connecting to the network, the instance of
385 CIM_PhysicalConnector shall be compliant with the [Physical Asset Profile](#). Instrumentation of the
386 CIM_Realizes class is conditional. If a corresponding instance of CIM_PhysicalConnector is instantiated,
387 it shall be associated to the corresponding CIM_NetworkPort via a CIM_Realizes instance.

388 **7.2 Representing a Communication Endpoint**

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

391 **7.2.1 Endpoint Identified by Hardware MAC**

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

394 **7.2.2 Communication Endpoint Identified by Assigned MAC**

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

398 **7.2.3 Relationship between the Interface and Port**

399 For each instance of CIM_LANEndpoint, one instance of CIM_DeviceSAPIImplementation shall associate
400 the CIM_LANEndpoint with the CIM_NetworkPort.

401 **7.2.4 Endpoint State Management Is Supported — Conditional**

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

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

407 **7.2.4.1 CIM_EnabledLogicalElementCapabilities**

408 When state management is supported, exactly one instance of CIM_EnabledLogicalElementCapabilities
409 shall be associated with the CIM_LANEndpoint instance through an instance of the
410 CIM_ElementCapabilities association.

411 **7.2.4.1.1 CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported**

412 The RequestedStatesSupported property may contain zero or more of the following values: 2 (Enabled),
413 3 (Disabled), or 11 (Reset).

414 **7.2.4.2 CIM_LANEndpoint.RequestedState**

415 When the CIM_LANEndpoint.RequestStateChange() method is successfully invoked, the value of the RequestedState property shall be the value of the RequestedState parameter. If the method is not
416 successfully invoked, the value of the RequestedState property is indeterminate.
417

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

420 **7.2.4.3 CIM_LANEndpoint.EnabledState**

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

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

426 **7.2.5 Endpoint State Management Is Not Supported**

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

429 **7.2.5.1 CIM_EnabledLogicalElementCapabilities**

430 When state management is not supported, exactly one instance of
431 CIM_EnabledLogicalElementCapabilities may be associated with the CIM_LANEndpoint instance through
432 an instance of the CIM_ElementCapabilities association.

433 **7.2.5.1.1 CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported**

434 The CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported property shall not contain any
435 values.

436 **7.2.5.2 CIM_LANEndpoint.RequestedState**

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

438 **7.2.5.3 CIM_LANEndpoint.EnabledState**

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

441 **7.2.6 Modifying ElementName Is Supported — Conditional**

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

446 **7.2.6.1 CIM_EnabledLogicalElementCapabilities**

447 An instance of CIM_EnabledLogicalElementCapabilities shall be associated with the CIM_LANEndpoint
448 instance through an instance of CIM_ElementCapabilities.

449 **7.2.6.1.1 CIM_EnabledLogicalElementCapabilities.ElementNameEditSupported**

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

452 **7.2.6.1.2 CIM_EnabledLogicalElement.MaxElementNameLen**

453 The MaxElementNameLen property shall be implemented.

454 **7.2.7 Modifying ElementName Is Not Supported**

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

457 **7.2.7.1 CIM_EnabledLogicalElementCapabilities**

458 An instance of CIM_EnabledLogicalElementCapabilities may be associated with the CIM_LANEndpoint
459 instance through an instance of CIM_ElementCapabilities.

460 **7.2.7.1.1 CIM_EnabledLogicalElementCapabilities.ElementNameEditSupported**

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

463 **7.2.7.1.2 CIM_EnabledLogicalElement.MaxElementNameLen**

464 The MaxElementNameLen property may be implemented. The MaxElementNameLen property is
465 irrelevant in this context.

466 **7.3 Managing Network Endpoints**

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

468 When an implementation supports the creation of network endpoints, there shall be an instance of
469 CIM_NetworkPortConfigurationService. An instance of CIM_ServiceAffectsElement is conditional. When
470 an instance of CIM_NetworkPortConfigurationService is instrumented, there shall be an instance of
471 CIM_ServiceAffectsElement that references the Central Instance and the
472 CIM_NetworkPortConfigurationService instance. The CIM_NetworkPortConfigurationService instance
473 shall be associated to an instance of CIM_ComputerSystem through an instance of CIM_HostedService.
474 A network endpoint can be created using the AddLANEndpoint() method of the
475 CIM_NetworkPortConfigurationService, as described in 8.1.

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

478 **7.4 Representing Multiple Ports Controlled from a Single Controller**

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

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

486 **7.4.1 Modeling the Controller**

487 An instance of CIM_PortController shall represent the controller.

488 **7.4.2 Relationship between Controller and Port**

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

491 7.4.3 Controller State Management Is Supported — Conditional

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

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

497 **Conditional Determination:** A client can determine whether state management is supported as follows:

- 498 1) Find the CIM_EnabledLogicalElementCapabilities instance that is associated with the
499 CIM_PortController instance.
- 500 2) Query the value of the RequestedStatesSupported property. If at least one value is specified,
501 state management is supported.

502 7.4.3.1 CIM_EnabledLogicalElementCapabilities

503 When state management is supported, exactly one instance of CIM_EnabledLogicalElementCapabilities
504 shall be associated with the CIM_PortController instance through an instance of the
505 CIM_ElementCapabilities association.

506 7.4.3.1.1 CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported

507 The RequestedStatesSupported property may contain zero or more of the following values: 2 (Enabled),
508 3 (Disabled), or 11 (Reset).

509 7.4.3.2 CIM_PortController.RequestedState

510 When the CIM_PortController.RequestStateChange() method is successfully invoked, the value of the
511 RequestedState property shall be the value of the RequestedState parameter. If the method is not
512 successfully invoked, the value of the RequestedState property is indeterminate.

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

515 7.4.3.3 CIM_PortController.EnabledState

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

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

521 7.4.4 Controller State Management Is Not Supported

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

524 7.4.4.1 CIM_EnabledLogicalElementCapabilities

525 When state management is not supported, exactly one instance of
526 CIM_EnabledLogicalElementCapabilities may be associated with the CIM_PortController instance
527 through an instance of the CIM_ElementCapabilities association.

528 **7.4.4.1.1 CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported**

529 The CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported property shall not contain any
530 values.

531 **7.4.4.2 CIM_PortController.RequestedState**

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

533 **7.4.4.3 CIM_PortController.EnabledState**

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

536 **7.4.5 Modifying ElementName Is Supported—Conditional**

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

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

- 542 1) Find the CIM_EnabledLogicalElementCapabilities instance that is associated with the
543 CIM_PortController instance.
- 544 2) Query the value of the ElementNameEditSupported property of the instance. If the value is
545 TRUE, the client can modify the CIM_PortController.ElementName property.

546 **7.4.5.1 CIM_EnabledLogicalElementCapabilities**

547 An instance of CIM_EnabledLogicalElementCapabilities shall be associated with the CIM_PortController
548 instance through an instance of CIM_ElementCapabilities.

549 **7.4.5.1.1 CIM_EnabledLogicalElementCapabilities.ElementNameEditSupported**

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

552 **7.4.5.1.2 CIM_EnabledLogicalElement.MaxElementNameLen**

553 The MaxElementNameLen property shall be implemented.

554 **7.4.6 Modifying ElementName Is Not Supported**

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

557 **7.4.6.1 CIM_EnabledLogicalElementCapabilities**

558 An instance of CIM_EnabledLogicalElementCapabilities may be associated with the CIM_PortController
559 instance through an instance of CIM_ElementCapabilities.

560 **7.4.6.1.1 CIM_EnabledLogicalElementCapabilities.ElementNameEditSupported**

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

563 **7.4.6.1.2 CIM_EnabledLogicalElement.MaxElementNameLen**

564 The MaxElementNameLen property may be implemented. The MaxElementNameLen property is
565 irrelevant in this context.

566 **8 Methods**

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

569 **8.1 CIM_NetworkPortConfigurationService.AddLANEndpoint()**

570 The AddLANEndpoint() method is used to create a new endpoint on a network port. This method shall be
571 supported when the CIM_NetworkPortConfigurationService is instrumented. When this method is
572 invoked, the implementation shall attempt to create a new instance of CIM_LANEndpoint. The
573 MACAddress property of the CIM_LANEndpoint instance shall have the value of the Address parameter
574 of the method invocation.

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

579 When the AliasAddresses parameter is specified in the method invocation, the AliasAddresses property
580 of the CIM_LANEndpoint instance shall have the value of the AliasAddresses parameter. When the
581 AliasAddresses parameter is not specified in the method invocation, the AliasAddresses property of the
582 CIM_LANEndpoint instance shall have a value of NULL.

583 When the GroupAddresses parameter is specified in the method invocation, the GroupAddresses
584 property of the CIM_LANEndpoint instance shall have the value of the GroupAddresses parameter. When the
585 GroupAddresses parameter is not specified in the method invocation, the GroupAddresses property
586 of the CIM_LANEndpoint instance shall have a value of NULL.

587 Before creating the instance of CIM_LANEndpoint, the implementation shall verify that the communication
588 endpoint represented by the resultant CIM_LANEndpoint instance is valid for the CIM_NetworkPort
589 instance that is identified by the Port parameter of the method invocation. If the resultant
590 CIM_LANEndpoint represents a valid endpoint for the identified CIM_NetworkPort instance, the
591 implementation shall create the following instances:

- 592
 - the instance of CIM_LANEndpoint described in the preceding paragraph
 - an instance of CIM_DeviceSAPIImplementation that references the newly created instance of
594 CIM_LANEndpoint and the instance of CIM_NetworkPort that is identified by the Port parameter
595 of the method invocation
 - an instance of CIM_HostedAccessPoint that references the CIM_LANEndpoint and references
597 the instance of CIM_ComputerSystem with which the instance of CIM_NetworkPort that the Port
598 parameter identified is associated through the CIM_SystemDevice association

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

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

605 No standard messages are defined.

606 **Table 2 – CIM_NetworkPortConfigurationService.AddLANEndpoint() Method: Return Code Values**

Value	Description
0	Request was successfully executed.
2	Error occurred

607 **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

608 **8.2 CIM_NetworkPort.RequestStateChange()**

609 Invocation of the RequestStateChange() method changes the element's state to the value specified in the
 610 RequestedState parameter. The 2 (Enabled) and 3 (Disabled) values of the RequestedState parameter
 611 shall correspond to enabling or disabling the network interface that the CIM_NetworkPort instance
 612 represents. A value of 11 (Reset) for the RequestedState parameter shall be equivalent to disabling and
 613 then enabling the network interface that the CIM_NetworkPort instance represents.

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

615 No standard messages are defined.

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

618 **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

619

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

620 **8.2.1.1 CIM_NetworkPort.RequestStateChange() ConditionalSupport**

621 When an instance of CIM_EnabledLogicalElementCapabilities is associated with the CIM_NetworkPort
 622 instance and the CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported property contains
 623 at least one value, the CIM_NetworkPort.RequestStateChange() method shall be implemented and
 624 supported. The CIM_NetworkPort.RequestStateChange() method shall not return a value of 1 (Not
 625 Supported).

626 **8.3 CIM_LANEndpoint.RequestStateChange()**

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

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

633 No standard messages are defined.

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

636

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

637

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

638 **8.3.1.1 CIM_LANEndpoint.RequestStateChange() Supported**

639 When an instance of CIM_EnabledLogicalElementCapabilities is associated with the CIM_LANEndpoint
 640 instance and the CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported property contains
 641 at least one value, the CIM_LANEndpoint.RequestStateChange() method shall be implemented and
 642 supported. The CIM_LANEndpoint.RequestStateChange() method shall not return a value of 1 (Not
 643 Supported).

644 **8.4 CIM_PortController.RequestStateChange()**

645 Invocation of the RequestStateChange() method changes the element's state to the value specified in the
 646 RequestedState parameter. The 2 (Enabled) and 3 (Disabled) values of the RequestedState parameter
 647 shall correspond to enabling or disabling the controller that the CIM_PortController instance represents. A
 648 value of 11 (Reset) for the RequestedState parameter shall be equivalent to disabling and then enabling
 649 the controller that the CIM_PortController instance represents.

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

651 No standard messages are defined.

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

654

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

655

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

656 8.4.1.1 CIM_PortController.RequestStateChange() Supported

657 When an instance of CIM_EnabledLogicalElementCapabilities is associated with the CIM_PortController
 658 instance and the CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported property contains
 659 at least one value, the CIM_PortController.RequestStateChange() method shall be implemented and
 660 supported. The CIM_PortController.RequestStateChange() method shall not return a value of 1 (Not
 661 Supported).

662 8.5 Profile Conventions for Operations

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

665 The default list of operations is as follows:

- 666 • GetInstance
- 667 • Associators
- 668 • AssociatorNames
- 669 • References
- 670 • ReferenceNames
- 671 • EnumerateInstances
- 672 • EnumerateInstanceNames

673 **8.6 CIM_ControlledBy**

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

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

678 **Table 10 – Operations: CIM_ControlledBy**

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

679 **8.7 CIM_DeviceSAPImplementation**

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

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

684 **Table 11 – Operations: CIM_DeviceSAPImplementation**

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

685 **8.8 CIM_ElementCapabilities**

686 Table 12 lists implementation requirements for operations. If implemented, these operations shall be
 687 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 12, all operations
 688 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 12 – Operations: CIM_ElementCapabilities**

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

691 **8.9 CIM_EnabledLogicalElementCapabilities**

692 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 **8.10 CIM_HostedAccessPoint**

695 Table 13, all operations in the default list in 8.5 shall be implemented as defined in [DSP0200](#).

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

697 **Table 13 – Operations: CIM_HostedAccessPoint**

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

698 **8.11 CIM_HostedService**

699 Table 14, all operations in the default list in 8.5 shall be implemented as defined in [DSP0200](#).

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

701 **Table 14 – Operations: CIM_HostedService**

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

702 **8.12 CIM_LANEndpoint**

703 Table 15 lists implementation requirements for operations. If implemented, these operations shall be
 704 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 15, all operations
 705 in the default list in 8.5 shall be implemented as defined in [DSP0200](#).

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

707 **Table 15 – Operations: CIM_LANEndpoint**

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

708 **8.12.1 CIM_LANEndpoint — DeleteInstance**

709 An implementation may support the DeleteInstance operation for instances of CIM_LANEndpoint. When
 710 the implementation supports the DeleteInstance operation, it may support the operation for some or all of
 711 the CIM_LANEndpoint instances implemented. When the DeleteInstance operation is supported for an
 712 instance of CIM_LANEndpoint, the implementation shall delete the instance of CIM_LANEndpoint and the
 713 instances of CIM_DeviceSAPIImplementation and CIM_HostedAccessPoint that reference the
 714 CIM_LANEndpoint instance.

715 The implementation shall not support the DeleteInstance operation for the CIM_LANEndpoint instance
 716 that is identified in 7.2.1.

717 **8.12.2 CIM_LANEndpoint — ModifyInstance**

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

720 **8.12.2.1 CIM_LANEndpoint.MACAddress**

721 The ModifyInstance operation shall not modify the MACAddress property of a CIM_LANEndpoint
 722 instance.

723 **8.12.2.2 CIM_LANEndpoint.ElementName**

724 When an instance of CIM_EnabledLogicalElementCapabilities is associated with the CIM_LANEndpoint
 725 instance and the CIM_EnabledLogicalElementCapabilities.ElementNameEditSupported property has a
 726 value of TRUE, the implementation shall allow the ModifyInstance operation to change the value of the
 727 ElementName property of the CIM_LANEndpoint instance. The ModifyInstance operation shall enforce
 728 the length restriction specified in the MaxElementNameLen property of the
 729 CIM_EnabledLogicalElementCapabilities instance.

730 When an instance of CIM_EnabledLogicalElementCapabilities is not associated with the
 731 CIM_LANEndpoint instance, or the ElementNameEditSupported property of the
 732 CIM_EnabledLogicalElementCapabilities instance has a value of FALSE, the implementation shall not
 733 allow the ModifyInstance operation to change the value of the ElementName property of the
 734 CIM_LANEndpoint instance.

735 **8.13 CIM_NetworkPort**

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

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

740 **Table 16 – Operations: CIM_NetworkPort**

Operation	Requirement	Messages
ModifyInstance	Optional. See 8.13.1.1.	None

741 **8.13.1 CIM_NetworkPort — ModifyInstance Operation**

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

744 **8.13.1.1 CIM_NetworkPort.ElementName**

745 When an instance of CIM_EnabledLogicalElementCapabilities is associated with the CIM_NetworkPort
 746 instance and the CIM_EnabledLogicalElementCapabilities.ElementNameEditSupported property has a
 747 value of TRUE, the implementation shall allow the ModifyInstance operation to change the value of the
 748 ElementName property of the CIM_NetworkPort instance. The ModifyInstance operation shall enforce the
 749 length restriction specified in the MaxElementNameLen property of the
 750 CIM_EnabledLogicalElementCapabilities instance.

751 When an instance of CIM_EnabledLogicalElementCapabilities is not associated with the
 752 CIM_NetworkPort instance, or the ElementNameEditSupported property of the

753 CIM_EnabledLogicalElementCapabilities instance has a value of FALSE, the implementation shall not
754 allow the ModifyInstance operation to change the value of the ElementName property of the
755 CIM_NetworkPort instance.

756 **8.14 CIM_NetworkPortConfigurationService**

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

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

759 **8.15 CIM_PortController**

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

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

764 **Table 17 – Operations: CIM_PortController**

Operation	Requirement	Messages
ModifyInstance	Optional. See 8.15.1.1.	None

765 **8.15.1 CIM_PortController — ModifyInstance Operation**

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

768 **8.15.1.1 CIM_PortController.ElementName Property**

769 When an instance of CIM_EnabledLogicalElementCapabilities is associated with the CIM_PortController
770 instance and the CIM_EnabledLogicalElementCapabilities.ElementNameEditSupported property has a
771 value of TRUE, the implementation shall allow the ModifyInstance operation to change the value of the
772 ElementName property of the CIM_PortController instance. The ModifyInstance operation shall enforce
773 the length restriction specified in the MaxElementNameLen property of the
774 CIM_EnabledLogicalElementCapabilities instance.

775 When an instance of CIM_EnabledLogicalElementCapabilities is not associated with the
776 CIM_PortController instance, or the ElementNameEditSupported property of the
777 CIM_EnabledLogicalElementCapabilities instance has a value of FALSE, the implementation shall not
778 allow the ModifyInstance operation to change the value of the ElementName property of the
779 CIM_PortController instance.

780 **8.16 CIM_Realizes**

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 18 – Operations: CIM_Realizes**

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

786

787 **8.17 CIM_ServiceAffectsElement**

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

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

792 **Table 19 – Operations: CIM_ServiceAffectsElement**

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

793 **8.18 CIM_SystemDevice**

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

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

798 **Table 20 – Operations: CIM_SystemDevice**

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

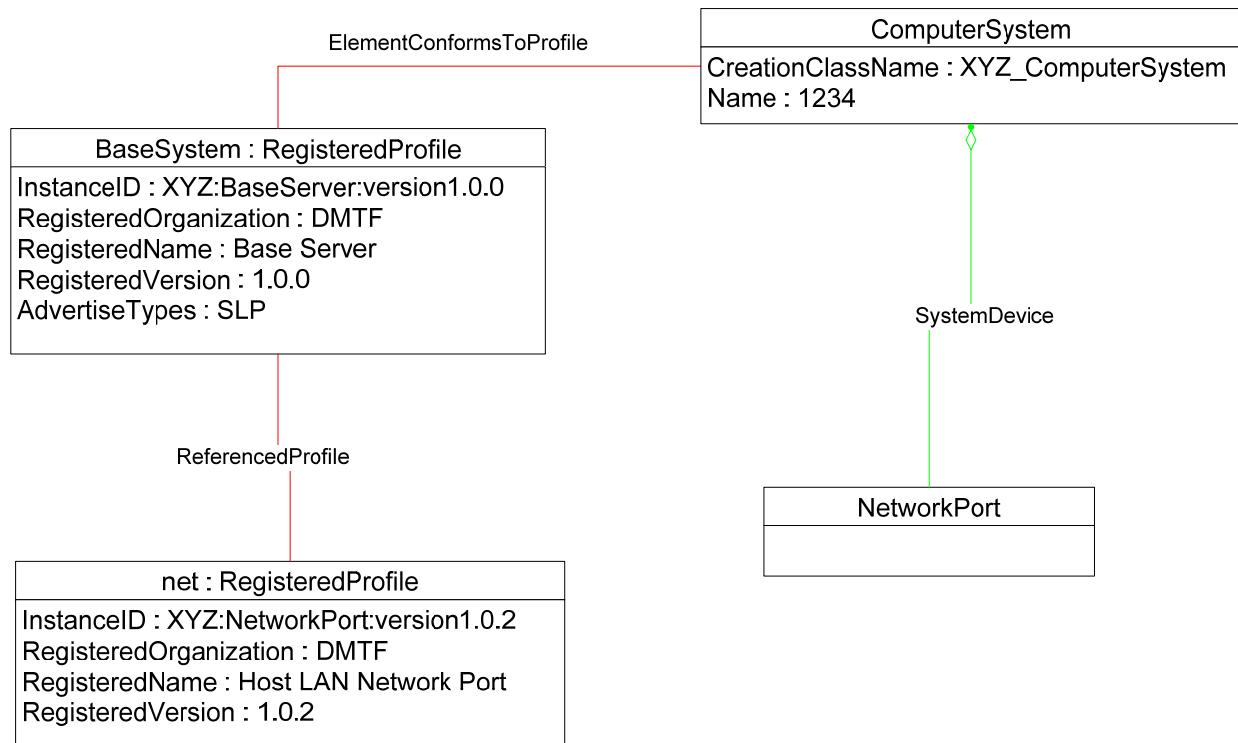
799 9 Use Cases

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

801 9.1 Object Diagrams

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

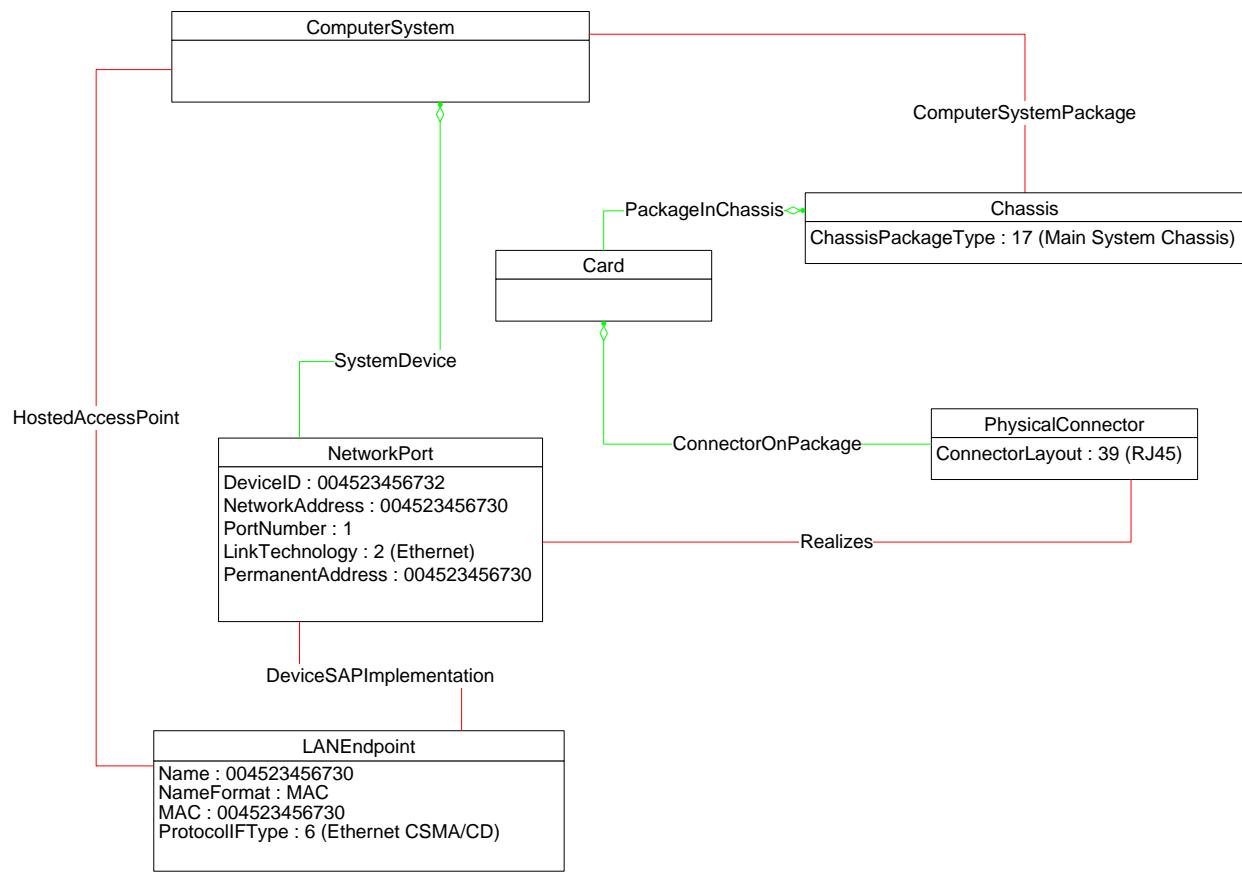
807 The CIM_NetworkPort instance is scoped to an instance of CIM_ComputerSystem. This instance of
 808 CIM_ComputerSystem is conformant with the DMTF *Base Server Profile*, version 1.0.0 as indicated by
 809 the CIM_ElementConformsToProfile association to the CIM_RegisteredProfile instance. The Scoping
 810 Instance in Figure 2 is the CIM_ComputerSystem instance. The Central Instance is the CIM_NetworkPort.
 811 The CIM_ReferencedProfile relationship between *BaseSystem* and *net* places the CIM_NetworkPort
 812 instance within the scope of *net*. Thus, the CIM_NetworkPort instance is conformant with the *Host LAN*
 813 *Network Port Profile*, version 1.0.0.



815 **Figure 2 – Registered Profile**

816 Figure 3 is a simple object diagram for a single network port with a single active network interface. The
 817 network port is represented by an instance of CIM_NetworkPort. The active interface is represented by an
 818 instance of CIM_LANEndpoint, which is associated with the CIM_NetworkPort instance through the
 819 CIM_DeviceSAPIImplementation association. In the system modeled, the network port is reached through
 820 an RJ-45 connector located directly on the motherboard of the system. This connection is indicated by the

821 CIM_Realizes association between the CIM_NetworkPort instance and the CIM_PhysicalConnector instance.

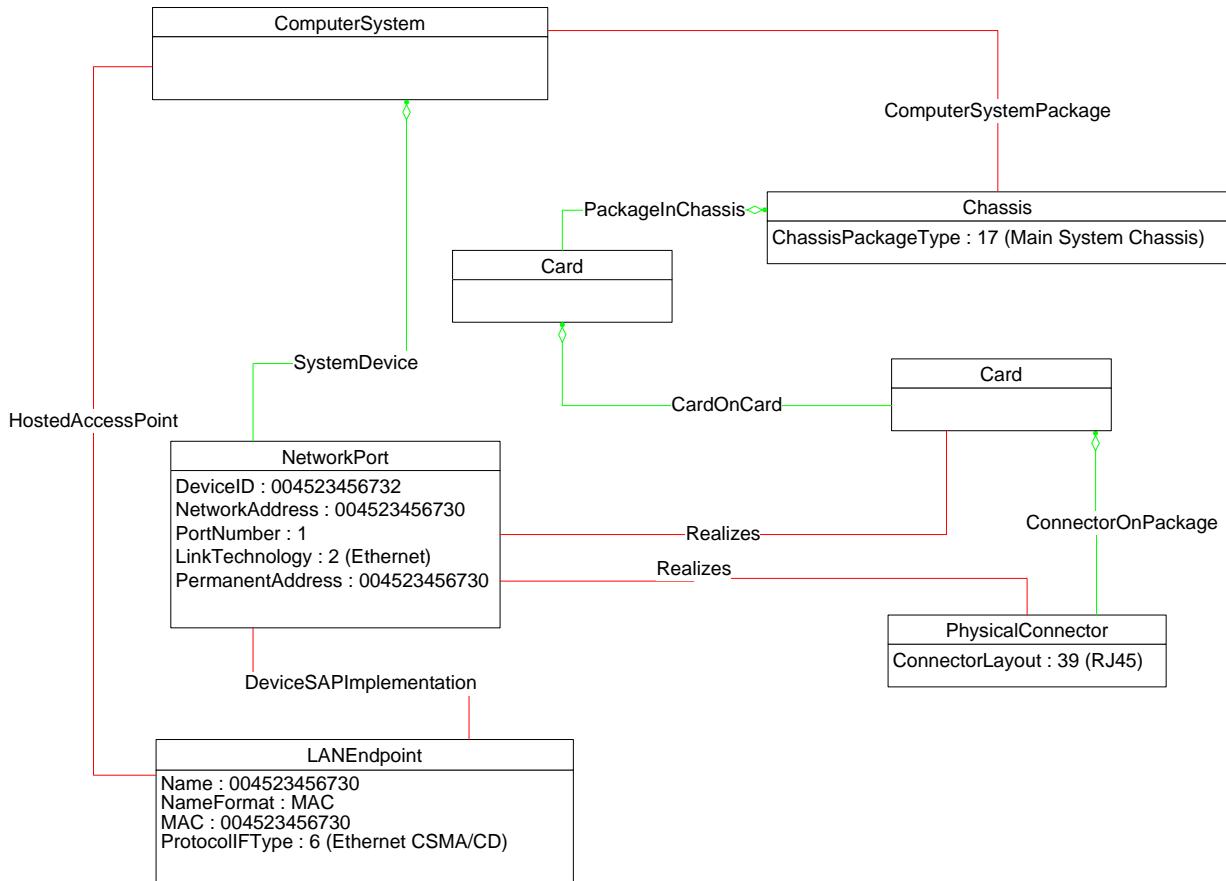


823

824

Figure 3 – Single Interface

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

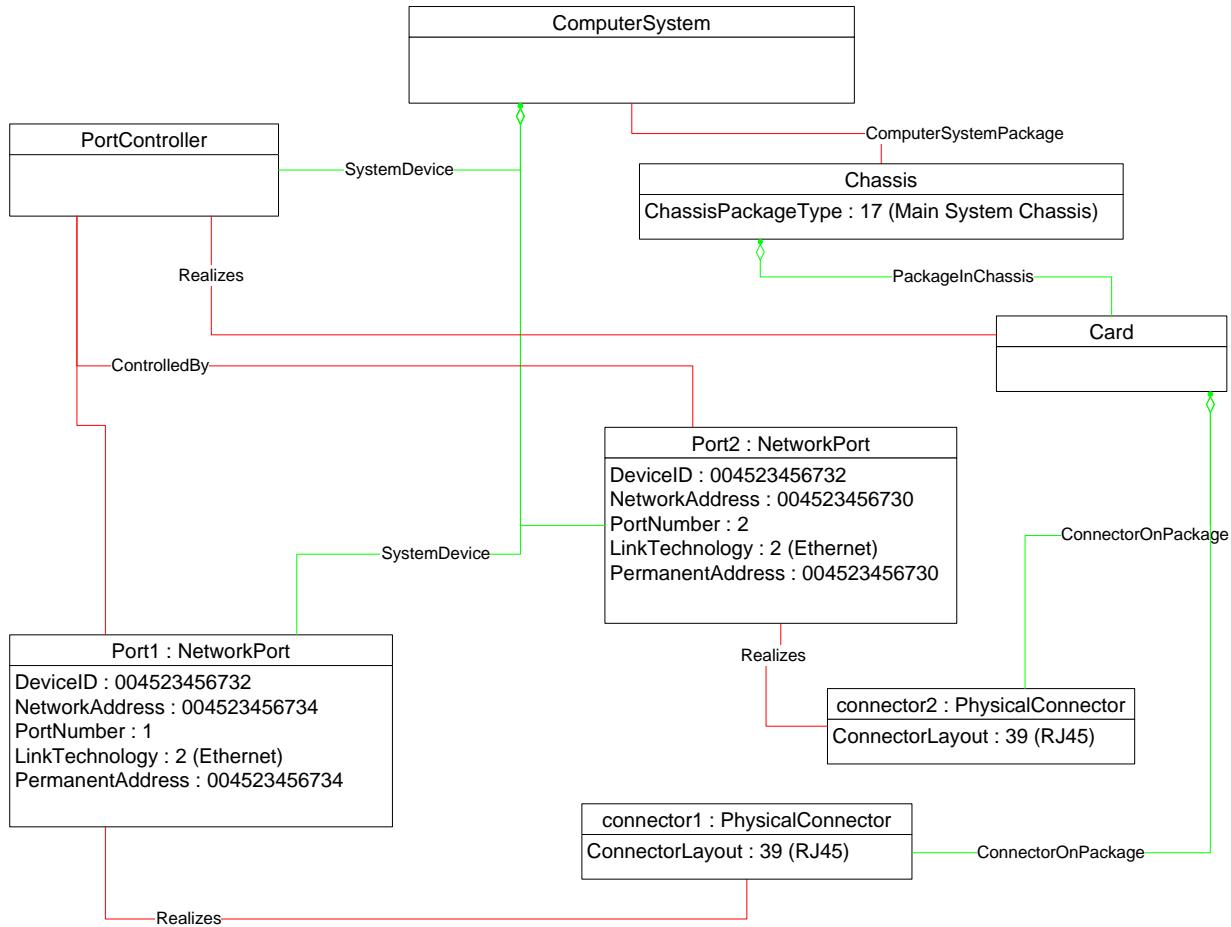


827

828

Figure 4 – Single Interface, Separate Card

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



834

835

Figure 5 – One Controller for Two Ports

836 9.2 Querying MAC Address for an Interface

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

- 838 1) Find all instances of CIM_LANEndpoint that are associated with the CIM_NetworkPort instance through instances of CIM_DeviceSAPIImplementation.
- 839 2) Query the MACAddress property of each instance of CIM_LANEndpoint.

841 9.3 Determining Physical Connector for a Network Address

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

- 844 1) Find the instance of CIM_NetworkPort that is associated with the CIM_LANEndpoint instance through an instance of CIM_DeviceSAPIImplementation.

846 2) Find the instance of CIM_PhysicalConnector that is associated with the CIM_NetworkPort
 847 instance through an instance of CIM_Realizes.

848 9.4 Determining If Physical Communication Is Possible

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

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

852 9.5 Correlating Controller and Port

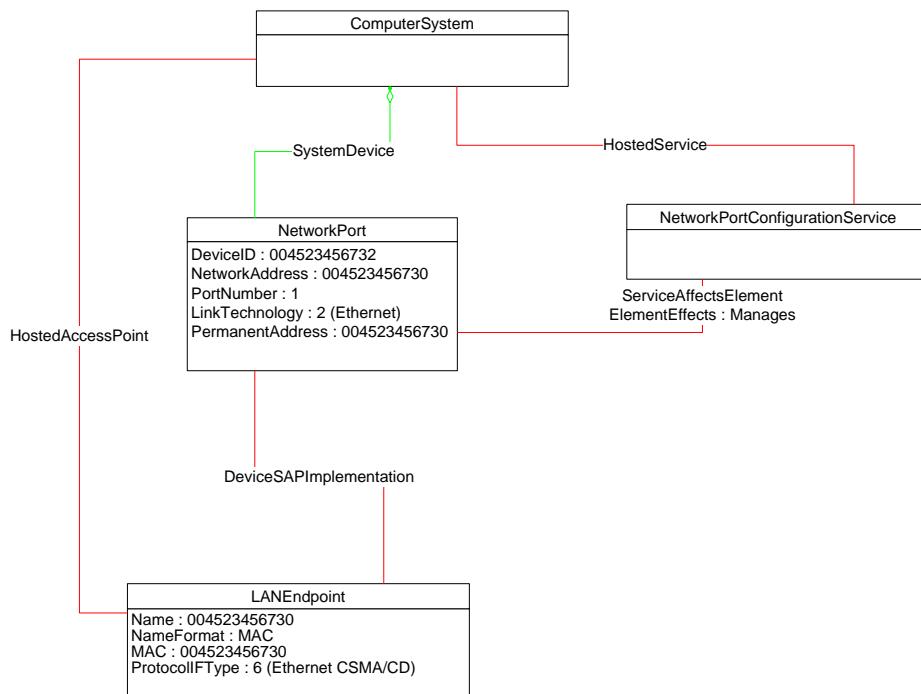
853 Multiple network ports may be controlled by a single controller. A client can determine which controller
 854 controls a network port as follows:

855 Find the instance of CIM_PortController that is associated with the CIM_NetworkPort instance
 856 through an instance of CIM_ControlledBy.

857 9.6 Adding an Endpoint to the Port

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

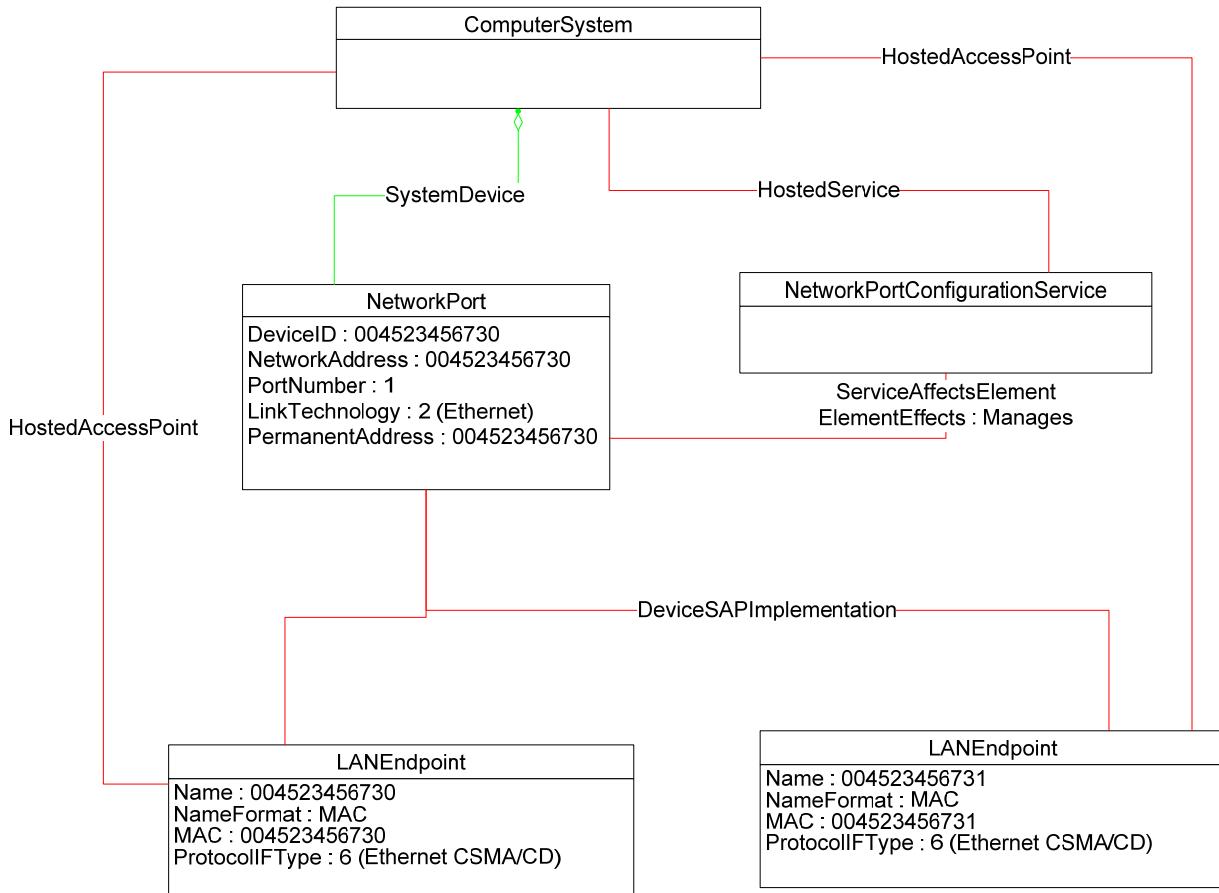
863 Figure 6 illustrates an example of a single endpoint associated with the network port. The endpoint
 864 corresponds to the real physical address burned into the network port.



865

866 **Figure 6 – Endpoint Management Supported**

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



871

872

Figure 7 – Second Endpoint Added

873 9.7 Determining If ElementName Can Be Modified

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

- 876 1) Find the CIM_EnabledLogicalElementCapabilities instance that is associated with the target
 877 instance.
- 878 2) Query the value of the ElementNameEditSupported property of the
 879 CIM_EnabledLogicalElementCapabilities instance. If the value is TRUE, the client can modify
 880 the ElementName property of the target instance.

881 **9.8 Determining If State Management Is Supported**

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

- 884 1) Find the CIM_EnabledLogicalElementCapabilities instance that is associated with the
 885 CIM_LANEndpoint instance.
- 886 2) Query the value of the RequestedStatesSupported property. If at least one value is specified,
 887 state management is supported.

888 **10 CIM Elements**

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

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

Element Name	Requirement	Description
Classes		
CIM_ControlledBy	Optional	See 10.1.
CIM_DeviceSAPIImplementation	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		

893 **10.1 CIM_ControlledBy**

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

897 **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..*

898 **10.2 CIM_DeviceSAPIImplementation**

899 CIM_DeviceSAPIImplementation is used to associate the CIM_LANEndpoint instance with the
 900 CIM_NetworkPort instance that provides the network access. Table 23 provides information about the
 901 properties of CIM_DeviceSAPIImplementation.

902 **Table 23 – Class: CIM_DeviceSAPIImplementation**

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..*

903 **10.3 CIM_ElementCapabilities — LANEndpoint**

904 CIM_ElementCapabilities is used to associate an instance of CIM_EnabledLogicalElementCapabilities
 905 with an instance of CIM_LANEndpoint. Table 24 provides information about the properties of
 906 CIM_ElementCapabilities in this context.

907 **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

908 **10.4 CIM_ElementCapabilities — NetworkPort**

909 CIM_ElementCapabilities is used to associate an instance of CIM_EnabledLogicalElementCapabilities
 910 with an instance of CIM_NetworkPort. Table 25 provides information about the properties of
 911 CIM_ElementCapabilities in this context.

912 **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

913 **10.5 CIM_ElementCapabilities — PortController**

914 CIM_ElementCapabilities is used to associate an instance of CIM_EnabledLogicalElementCapabilities
 915 with an instance of CIM_PortController. Table 26 provides information about the properties of
 916 CIM_ElementCapabilities in this context.

917 **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

918 **10.6 CIM_EnabledLogicalElementCapabilities — LANEndpoint**

919 CIM_EnabledLogicalElementCapabilities is used to indicate support for managing the state of the network
 920 interface. Table 27 provides information about the properties of CIM_EnabledLogicalElementCapabilities
 921 in this context.

922 **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.

923 **10.7 CIM_EnabledLogicalElementCapabilities — NetworkPort**

924 CIM_EnabledLogicalElementCapabilities is used to indicate support for managing the state of the network
 925 port. Table 28 provides information about the properties of CIM_EnabledLogicalElementCapabilities in
 926 this context.

927 **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.

928 **10.8 CIM_EnabledLogicalElementCapabilities — PortController**

929 CIM_EnabledLogicalElementCapabilities is used to indicate support for managing the state of the port
 930 controller. Table 29 provides information about the properties of CIM_EnabledLogicalElementCapabilities
 931 in this context.

932 **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.

933 **10.9 CIM_HostedAccessPoint**

934 CIM_HostedAccessPoint is used to relate a CIM_LANEndpoint instance to its scoping
 935 CIM_ComputerSystem instance. Table 30 provides information about the properties of
 936 CIM_HostedAccessPoint.

937 **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..*

938 **10.10 CIM_HostedService**

939 CIM_HostedService is used to associate the CIM_NetworkPortConfigurationService instance with the
 940 CIM_ComputerSystem instance to which it is scoped. Table 31 provides information about the properties
 941 of CIM_HostedService.

942 **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 *

943 **10.11 CIM_LANEndpoint**

944 CIM_LANEndpoint represents a MAC address to which the network port will respond on the LAN. Table
945 32 provides information about the properties of CIM_LANEndpoint.

946 **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
ProtocolIFTType	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.

947 **10.12 CIM_NetworkPort**

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

950 **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.

951 **10.13 CIM_NetworkPortConfigurationService**

952 CIM_NetworkPortConfigurationService represents the ability to add endpoints to the network port. Table
 953 34 provides information about the properties of CIM_NetworkPortConfigurationService.

954 **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.

955 **10.14 CIM_PhysicalConnector**

956 CIM_PhysicalConnector is used to represent the physical connector that connects the network port to the
 957 physical network. This class is defined by the [Physical Asset Profile](#). The behavior specified in Table 35 is
 958 in addition to that specified by the [Physical Asset Profile](#).

959 **Table 35 – Class: CIM_PhysicalConnector**

Properties	Requirement	Description
ConnectorLayout	Mandatory	None

960 **10.15 CIM_PortController**

961 CIM_PortController represents a network controller. Table 36 provides information about the properties of
 962 CIM_PortController.

963 **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.

964 **10.16 CIM_Realizes**

965 The CIM_Realizes association is used to associate the CIM_NetworkPort with an instance of
 966 CIM_PhysicalConnector when an instance of CIM_PhysicalConnector is instrumented. This class is
 967 defined by the [Physical Asset Profile](#). The behavior specified in Table 37 is in addition to that specified by
 968 the [Physical Asset Profile](#).

969 **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..*

970 **10.17 CIM_RegisteredProfile**

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

976 **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 "1.0.2".
RegisteredOrganization	Mandatory	This property shall have a value of "DMTF".

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

980 **10.18 CIM_ServiceAffectsElement**

981 CIM_ServiceAffectsElement is used to associate an instance of CIM_NetworkPortConfigurationService
 982 with an instance of CIM_NetworkPort that the service is able to configure. Table 39 provides information
 983 about the properties of CIM_ServiceAffectsElement.

984 **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)

985 **10.19 CIM_SystemDevice — CIM_NetworkPort**

986 CIM_SystemDevice is used to associate an instance of CIM_NetworkPort with the instance of
 987 CIM_ComputerSystem to which the CIM_NetworkPort is scoped. Table 40 provides information about the
 988 properties of CIM_SystemDevice.

989 **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..*

990 10.20 CIM_SystemDevice — CIM_PortController

991 CIM_SystemDevice is used to associate an instance of CIM_PortController with an instance of
992 CIM_ComputerSystem when CIM_PortController is implemented. Table 41 provides information about the
993 properties of CIM_SystemDevice.

994 **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 *

995

996
997
998
999

ANNEX A (informative)

Change Log

Version	Date	Description
1.0.0	2008-06-03	Final Standard
1.0.1	2010-09-15	Final Standard formatted for DMTF Standard release
1.0.2	2011-06-30	Errata version. Added operations tables in Section 8 for CIM_DeviceSAPIImplementation and CIM_Realizes.

1000