



1

2

3

4

Document Number: DSP1035

Date: 2008-09-04

Version: 1.0.0

5 **Host LAN Network Port Profile**

6 **Document Type: Specification**

7 **Document Status: Final Standard**

8 **Document Language: E**

9 Copyright Notice

10 Copyright © 2008 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 for uses consistent with this purpose, provided that correct attribution is given. As DMTF
14 specifications may be revised from time to time, the particular version and release date should always be
15 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26 withdrawn or modified after publication, and shall be indemnified and held harmless by any party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>.

CONTENTS

33	Foreword	6
34	Introduction	7
35	1 Scope	9
36	2 Normative References.....	9
37	2.1 Approved References	9
38	2.2 References under Development	9
39	2.3 Other References.....	9
40	3 Terms and Definitions	9
41	4 Symbols and Abbreviated Terms	10
42	5 Synopsis	11
43	6 Description	12
44	7 Implementation Requirements	13
45	7.1 Representing a Network Port.....	13
46	7.2 Representing a Communication Endpoint	15
47	7.3 Managing Network Endpoints.....	17
48	7.4 Representing Multiple Ports Controlled from a Single Controller	18
49	8 Methods.....	20
50	8.1 CIM_NetworkPortConfigurationService.AddLANEndpoint().....	20
51	8.2 CIM_NetworkPort.RequestStateChange().....	22
52	8.3 CIM_LANEndpoint.RequestStateChange()	22
53	8.4 CIM_PortController.RequestStateChange()	23
54	8.5 Profile Conventions for Operations.....	24
55	8.6 CIM_ControlledBy.....	25
56	8.7 CIM_ElementCapabilities	25
57	8.8 CIM_EnabledLogicalElementCapabilities.....	25
58	8.9 CIM_HostedAccessPoint	25
59	8.10 CIM_HostedService	25
60	8.11 CIM_LANEndpoint	26
61	8.12 CIM_NetworkPort.....	27
62	8.13 CIM_NetworkPortConfigurationService	27
63	8.14 CIM_PortController	27
64	8.15 CIM_ServiceAffectsElement	28
65	8.16 CIM_SystemDevice	28
66	9 Use Cases	29
67	9.1 Object Diagrams	29
68	9.2 Querying MAC Address for an Interface.....	31
69	9.3 Determining Physical Connector for a Network Address.....	31
70	9.4 Determining If Physical Communication Is Possible	32
71	9.5 Correlating Controller and Port	32
72	9.6 Adding an Endpoint to the Port.....	32
73	9.7 Determining If ElementName Can Be Modified.....	33
74	9.8 Determining If State Management Is Supported	34
75	10 CIM Elements	34
76	10.1 CIM_ControlledBy.....	34
77	10.2 CIM_DeviceSAPImplementation	35
78	10.3 CIM_ElementCapabilities–LANEndpoint	35
79	10.4 CIM_ElementCapabilities–NetworkPort.....	35
80	10.5 CIM_ElementCapabilities–PortController	36
81	10.6 CIM_EnabledLogicalElementCapabilities–LANEndpoint	36
82	10.7 CIM_EnabledLogicalElementCapabilities–NetworkPort.....	36
83	10.8 CIM_EnabledLogicalElementCapabilities–PortController	37
84	10.9 CIM_HostedAccessPoint.....	37

85	10.10 CIM_HostedService	37
86	10.11 CIM_LANEndpoint	38
87	10.12 CIM_NetworkPort.....	38
88	10.13 CIM_NetworkPortConfigurationService	39
89	10.14 CIM_PhysicalConnector	39
90	10.15 CIM_PortController	40
91	10.16 CIM_Realizes.....	40
92	10.17 CIM_RegisteredProfile.....	41
93	10.18 CIM_ServiceAffectsElement	41
94	10.19 CIM_SystemDevice–CIM_NetworkPort.....	41
95	10.20 CIM_SystemDevice–CIM_PortController	42
96	ANNEX A (informative) Change Log.....	43
97	ANNEX B (informative) Acknowledgments	44

98

99 Figures

100	Figure 1 – Host LAN Network Port Profile: Class Diagram.....	12
101	Figure 2 – Registered Profile	29
102	Figure 3 – Single Interface.....	30
103	Figure 4 – Single Interface, Separate Card	30
104	Figure 5 – One Controller for Two Ports	31
105	Figure 6 – Endpoint Management Supported.....	32
106	Figure 7 – Second Endpoint Added.....	33
107		

108 Tables

109	Table 1 – Referenced Profiles	11
110	Table 2 – CIM_NetworkPortConfigurationService.AddLANEndpoint() Method: Return Code Values	21
111	Table 3 – CIM_NetworkPortConfigurationService.AddLANEndpoint() Method: Parameters	21
112	Table 4 – CIM_NetworkPort.RequestStateChange() Method: Return Code Values.....	22
113	Table 5 – CIM_NetworkPort.RequestStateChange() Method: Parameters	22
114	Table 6 – CIM_LANEndpoint.RequestStateChange() Method: Return Code Values.....	23
115	Table 7 – CIM_LANEndpoint.RequestStateChange() Method: Parameters	23
116	Table 8 – CIM_PortController.RequestStateChange() Method: Return Code Values.....	24
117	Table 9 – CIM_PortController.RequestStateChange() Method: Parameters	24
118	Table 10 – Operations: CIM_ControlledBy	25
119	Table 11 – Operations: CIM_ElementCapabilities	25
120	Table 12 – Operations: CIM_HostedService	25
121	Table 13 – Operations: CIM_LANEndpoint.....	26
122	Table 14 – Operations: CIM_NetworkPort	27
123	Table 15 – Operations: CIM_PortController.....	27
124	Table 16 – Operations: CIM_ServiceAffectsElement	28
125	Table 17 – Operations: CIM_SystemDevice.....	28
126	Table 18 – CIM Elements: Network Port Profile	34
127	Table 19 – Class: CIM_ControlledBy.....	34
128	Table 20 – Class: CIM_DeviceSAPImplementation	35

129 Table 21 – Class: CIM_ElementCapabilities–LANEndpoint 35

130 Table 22 – Class: CIM_ElementCapabilities–NetworkPort..... 35

131 Table 23 – Class: CIM_ElementCapabilities–PortController 36

132 Table 24 – Class: CIM_EnabledLogicalElementCapabilities–LANEndpoint 36

133 Table 25 – Class: CIM_EnabledLogicalElementCapabilities–NetworkPort..... 36

134 Table 26 – Class: CIM_EnabledLogicalElementCapabilities–PortController 37

135 Table 27 – Class: CIM_HostedAccessPoint 37

136 Table 28 – Class: CIM_HostedService 37

137 Table 29 – Class: CIM_LANEndpoint 38

138 Table 30 – Class: CIM_NetworkPort..... 38

139 Table 31 – Class: NetworkPortConfigurationService..... 39

140 Table 32 – Class: CIM_PhysicalConnector 39

141 Table 33 – Class: CIM_PortController 40

142 Table 34 – Class: CIM_Realizes..... 40

143 Table 35 – Class: CIM_RegisteredProfile..... 41

144 Table 36 – Class: CIM_ServiceAffectsElement 41

145 Table 37 – Class: CIM_SystemDevice 41

146 Table 38 – Class: CIM_SystemDevice 42

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

Introduction

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

160

161

Host LAN Network Port Profile

162 1 Scope

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

167 2 Normative References

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

171 2.1 Approved References

172 DMTF [DSP0200](#), *CIM Operations over HTTP 1.2.0*

173 DMTF [DSP0004](#), *CIM Infrastructure Specification 2.3.0*

174 DMTF [DSP1000](#), *Management Profile Specification Template 1.0.0*

175 DMTF [DSP1001](#), *Management Profile Specification Usage Guide 1.0.0*

176 2.2 References under Development

177 DMTF [DSP1011](#), *Physical Asset Profile*

178 DMTF [DSP1033](#), *Profile Registration Profile 1.0.0*

179 2.3 Other References

180 ISO/IEC Directives, Part 2, *Rules for the structure and drafting of International Standards*,
181 <http://isotc.iso.org>

182 *Unified Modeling Language (UML) from the Open Management Group (OMG)*, <http://www.uml.org>

183 3 Terms and Definitions

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

185 3.1

186 **can**

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

188 3.2

189 **cannot**

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

- 191 **3.3**
192 **conditional**
193 indicates requirements to be followed strictly in order to conform to the document when the specified
194 conditions are met
- 195 **3.4**
196 **mandatory**
197 indicates requirements to be followed strictly in order to conform to the document and from which no
198 deviation is permitted
- 199 **3.5**
200 **may**
201 indicates a course of action permissible within the limits of the document
- 202 **3.6**
203 **need not**
204 indicates a course of action permissible within the limits of the document
- 205 **3.7**
206 **optional**
207 indicates a course of action permissible within the limits of the document
- 208 **3.8**
209 **referencing profile**
210 indicates a profile that owns the definition of this class and can include a reference to this profile in its
211 "Related Profiles" table
- 212 **3.9**
213 **shall**
214 indicates requirements to be followed strictly in order to conform to the document and from which no
215 deviation is permitted
- 216 **3.10**
217 **shall not**
218 indicates requirements to be followed strictly in order to conform to the document and from which no
219 deviation is permitted
- 220 **3.11**
221 **should**
222 indicates that among several possibilities, one is recommended as particularly suitable, without
223 mentioning or excluding others, or that a certain course of action is preferred but not necessarily required
- 224 **3.12**
225 **should not**
226 indicates that a certain possibility or course of action is deprecated but not prohibited

227 **4 Symbols and Abbreviated Terms**

228 **Experimental Maturity Level**

- 229
230 Some of the content considered for inclusion in *Host LAN Network Port Profile* has yet to receive
231 sufficient review to satisfy the adoption requirements set forth by the Technical Committee within the
232 DMTF. This content is presented here as an aid to implementers who are interested in likely future
233 developments within this specification. The content marked experimental may change as implementation

234 experience is gained. There is a high likelihood that it will be included in an upcoming revision of the
 235 specification. Until that time, it is purely informational, and is clearly marked within the text.
 236 A sample of the typographical convention for experimental content is included here:

237 **EXPERIMENTAL**

238 Experimental content appears here.

239 **EXPERIMENTAL**

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

241 **4.1**

242 **DNS**

243 Domain Name System

244 **4.2**

245 **DHCP**

246 Dynamic Host Configuration Protocol

247 **4.3**

248 **LAN**

249 Local Area Network

250 **5 Synopsis**

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

252 **Version:** 1.0.0

253 **Organization:** DMTF

254 **CIM Schema version:** 2.18

255 **Central Class:** CIM_NetworkPort

256 **Scoping Class:** CIM_ComputerSystem

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

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

263 CIM_NetworkPort shall be the Central Class of this profile. The instance of CIM_NetworkPort shall be the
 264 Central Instance of this profile. CIM_ComputerSystem shall be the Scoping Class of this profile. The
 265 instance of CIM_ComputerSystem with which the Central Instance is associated through an instance of
 266 CIM_SystemDevice shall be the Scoping Instance of this profile.

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

268 **Table 1 – Referenced Profiles**

Profile Name	Organization	Version	Description
Profile Registration	DMTF	1.0.0	Mandatory
Physical Asset	DMTF	1.0.0	Optional. See section 7.1.6.

269 **6 Description**

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

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

- 273 • a specification of the network port and related hardware
- 274 • network interfaces active over the network port

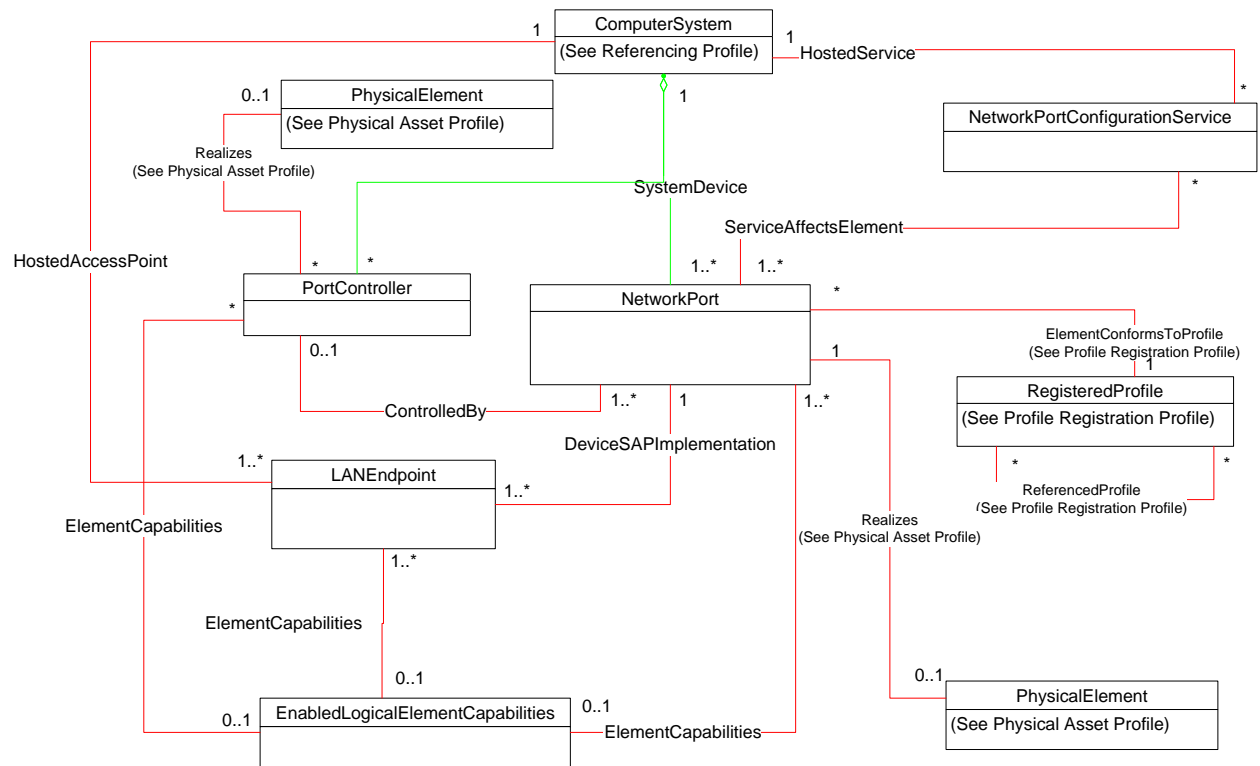
275 The following functionality is optional within the scope of this profile:

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

277 The following functionality is not covered in this profile:

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

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



288

289

Figure 1 – Host LAN Network Port Profile: Class Diagram

290 7 Implementation Requirements

291 This section details the requirements related to the arrangement of instances and properties of instances
292 for implementations of this profile.

293 7.1 Representing a Network Port

294 An instance of CIM_NetworkPort shall represent the network port.

295 7.1.1 CIM_NetworkPort.EnabledState—Enabled but Offline

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

299 7.1.2 Network Port State Management Is Supported—Conditional

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

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

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

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

310 7.1.2.1 CIM_EnabledLogicalElementCapabilities

311 When state management is supported, exactly one instance of CIM_EnabledLogicalElementCapabilities
312 shall be associated with the CIM_NetworkPort instance through an instance of the
313 CIM_ElementCapabilities association and it shall be subject to the conditions in this section.

314 7.1.2.1.1 CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported

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

317 7.1.2.2 CIM_NetworkPort.RequestedState

318 When the CIM_NetworkPort.RequestStateChange() method is successfully invoked, the value of the
319 RequestedState property shall be the value of the RequestedState parameter. If the method is not
320 successfully invoked, the value of the RequestedState property is indeterminate.

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

323 7.1.2.3 CIM_NetworkPort.EnabledState

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

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

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

329 **7.1.3 Network Port State Management Is Not Supported**

330 This section describes the CIM elements and behaviors that shall be implemented when management of
331 the Network Port state is not supported.

332 **7.1.3.1 CIM_EnabledLogicalElementCapabilities**

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

337 **7.1.3.1.1 CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported**

338 The CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported property shall not contain any
339 values.

340 **7.1.3.2 CIM_NetworkPort.RequestedState**

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

342 **7.1.3.3 CIM_NetworkPort.EnabledState**

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

345 **7.1.4 Modifying ElementName Is Supported—Conditional**

346 The CIM_NetworkPort.ElementName property may support being modified by the ModifyInstance
347 operation. See section 8.12.1.1. This behavior is conditional. This section describes the CIM elements
348 and behavior requirements when an implementation supports client modification of the
349 CIM_NetworkPort.ElementName property.

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

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

355 **7.1.4.1 CIM_EnabledLogicalElementCapabilities**

356 An instance of CIM_EnabledLogicalElementCapabilities shall be associated with the CIM_NetworkPort
357 instance through an instance of CIM_ElementCapabilities.

358 **7.1.4.1.1 CIM_EnabledLogicalElementCapabilities.ElementNameEditSupported**

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

361 **7.1.4.1.2 CIM_EnabledLogicalElement.MaxElementNameLen**

362 The MaxElementNameLen property shall be implemented.

363 **7.1.5 Modifying ElementName Is Not Supported**

364 This section describes the CIM elements and behaviors that shall be implemented when the
365 CIM_NetworkPort.ElementName does not support being modified by the ModifyInstance operation.

366 **7.1.5.1 CIM_EnabledLogicalElementCapabilities**

367 An instance of CIM_EnabledLogicalElementCapabilities may be associated with the CIM_NetworkPort
368 instance through an instance of CIM_ElementCapabilities.

369 **7.1.5.1.1 CIM_EnabledLogicalElementCapabilities.ElementNameEditSupported**

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

372 **7.1.5.1.2 CIM_EnabledLogicalElement.MaxElementNameLen**

373 The MaxElementNameLen property may be implemented. The MaxElementNameLen property is
374 irrelevant in this context.

375 **7.1.6 Representing the Physical Packaging**

376 Support for representing the physical packaging of the network device is optional. The physical packaging
377 may be modeled using one or more instances of CIM_PhysicalElement in accordance with the Physical
378 Asset Profile.

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

385 **7.2 Representing a Communication Endpoint**

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

388 **7.2.1 Endpoint Identified by Hardware MAC**

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

391 **7.2.2 Communication Endpoint Identified by Assigned MAC**

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

395 **7.2.3 Relationship between the Interface and Port**

396 For each instance of CIM_LANEndpoint, one instance of CIM_DeviceSAPImplementation shall associate
397 the CIM_LANEndpoint with the CIM_NetworkPort.

398 **7.2.4 Endpoint State Management Is Supported—Conditional**

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

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

404 **7.2.4.1 CIM_EnabledLogicalElementCapabilities**

405 When state management is supported, exactly one instance of CIM_EnabledLogicalElementCapabilities
406 shall be associated with the CIM_LANEndpoint instance through an instance of the
407 CIM_ElementCapabilities association.

408 **7.2.4.1.1 CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported**

409 The RequestedStatesSupported property may contain zero or more of the following values: 2 (Enabled),
410 3 (Disabled), or 11 (Reset).

411 **7.2.4.2 CIM_LANEndpoint.RequestedState**

412 When the CIM_LANEndpoint.RequestStateChange() method is successfully invoked, the value of the
413 RequestedState property shall be the value of the RequestedState parameter. If the method is not
414 successfully invoked, the value of the RequestedState property is indeterminate.

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

417 **7.2.4.3 CIM_LANEndpoint.EnabledState**

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

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

423 **7.2.5 Endpoint State Management Is Not Supported**

424 This section describes the CIM elements and behaviors that shall be implemented when management of
425 the endpoint state is not supported.

426 **7.2.5.1 CIM_EnabledLogicalElementCapabilities**

427 When state management is not supported, exactly one instance of
428 CIM_EnabledLogicalElementCapabilities may be associated with the CIM_LANEndpoint instance through
429 an instance of the CIM_ElementCapabilities association.

430 **7.2.5.1.1 CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported**

431 The CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported property shall not contain any
432 values.

433 **7.2.5.2 CIM_LANEndpoint.RequestedState**

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

435 7.2.5.3 CIM_LANEndpoint.EnabledState

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

438 7.2.6 Modifying ElementName Is Supported—Conditional

439 The CIM_LANEndpoint.ElementName property may support being modified by the ModifyInstance
440 operation. See section 8.11.2.2. This behavior is conditional. This section describes the CIM elements
441 and behavior requirements when an implementation supports client modification of the
442 CIM_LANEndpoint.ElementName property.

443 7.2.6.1 CIM_EnabledLogicalElementCapabilities

444 An instance of CIM_EnabledLogicalElementCapabilities shall be associated with the CIM_LANEndpoint
445 instance through an instance of CIM_ElementCapabilities.

446 7.2.6.1.1 CIM_EnabledLogicalElementCapabilities.ElementNameEditSupported

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

449 7.2.6.1.2 CIM_EnabledLogicalElement.MaxElementNameLen

450 The MaxElementNameLen property shall be implemented.

451 7.2.7 Modifying ElementName Is Not Supported

452 This section describes the CIM elements and behaviors that shall be implemented when the
453 CIM_LANEndpoint.ElementName does not support being modified by the ModifyInstance operation.

454 7.2.7.1 CIM_EnabledLogicalElementCapabilities

455 An instance of CIM_EnabledLogicalElementCapabilities may be associated with the CIM_LANEndpoint
456 instance through an instance of CIM_ElementCapabilities.

457 7.2.7.1.1 CIM_EnabledLogicalElementCapabilities.ElementNameEditSupported

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

460 7.2.7.1.2 CIM_EnabledLogicalElement.MaxElementNameLen

461 The MaxElementNameLen property may be implemented. The MaxElementNameLen property is
462 irrelevant in this context.

463 7.3 Managing Network Endpoints

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

465 EXPERIMENTAL

466 When an implementation supports the creation of network endpoints, there shall be an instance of
467 CIM_NetworkPortConfigurationService. An instance of CIM_ServiceAffectsElement is conditional. When
468 an instance of CIM_NetworkPortConfigurationService is instrumented, there shall be an instance of
469 CIM_ServiceAffectsElement that references the Central Instance and the
470 CIM_NetworkPortConfigurationService instance. The CIM_NetworkPortConfigurationService instance
471 shall be associated to an instance of CIM_ComputerSystem through an instance of CIM_HostedService.

472 A network endpoint can be created using the AddLANEndpoint() method of the
473 CIM_NetworkPortConfigurationService, as described in section 8.1.

474 **EXPERIMENTAL**

475 An implementation can remove a network endpoint by using the intrinsic DeleteInstance operation
476 defined in section 8.11.1.

477 **7.4 Representing Multiple Ports Controlled from a Single Controller**

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

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

485 **7.4.1 Modeling the Controller**

486 An instance of CIM_PortController shall represent the controller.

487 **7.4.2 Relationship between Controller and Port**

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

490 **7.4.3 Controller State Management Is Supported—Conditional**

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

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

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

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

501 **7.4.3.1 CIM_EnabledLogicalElementCapabilities**

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

505 **7.4.3.1.1 CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported**

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

508 7.4.3.2 CIM_PortController.RequestedState

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

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

514 7.4.3.3 CIM_PortController.EnabledState

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

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

520 7.4.4 Controller State Management Is Not Supported

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

523 7.4.4.1 CIM_EnabledLogicalElementCapabilities

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

527 7.4.4.1.1 CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported

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

530 7.4.4.2 CIM_PortController.RequestedState

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

532 7.4.4.3 CIM_PortController.EnabledState

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

535 7.4.5 Modifying ElementName Is Supported—Conditional

536 The CIM_PortController.ElementName property may support being modified by the ModifyInstance
537 operation. See section 8.14.1.1. This behavior is conditional. This section describes the CIM elements
538 and behavior requirements when an implementation supports client modification of the
539 CIM_PortController.ElementName property.

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

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

545 7.4.5.1 CIM_EnabledLogicalElementCapabilities

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

548 7.4.5.1.1 CIM_EnabledLogicalElementCapabilities.ElementNameEditSupported

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

551 7.4.5.1.2 CIM_EnabledLogicalElement.MaxElementNameLen

552 The MaxElementNameLen property shall be implemented.

553 7.4.6 Modifying ElementName Is Not Supported

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

556 7.4.6.1 CIM_EnabledLogicalElementCapabilities

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

559 7.4.6.1.1 CIM_EnabledLogicalElementCapabilities.ElementNameEditSupported

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

562 7.4.6.1.2 CIM_EnabledLogicalElement.MaxElementNameLen

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

565 8 Methods

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

568 EXPERIMENTAL**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
 585 the 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
- 593 • an instance of CIM_DeviceSAPImplementation 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
- 596 • 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 **EXPERIMENTAL**

609 8.2 CIM_NetworkPort.RequestStateChange()

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

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

616 No standard messages are defined.

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

619 **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

620 **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

621 8.2.1.1 CIM_NetworkPort.RequestStateChange() ConditionalSupport

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

627 8.3 CIM_LANEndpoint.RequestStateChange()

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

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

634 No standard messages are defined.

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

637 **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

638 **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

639 **8.3.1.1 CIM_LANEndpoint.RequestStateChange() Supported**

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

645 **8.4 CIM_PortController.RequestStateChange()**

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

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

652 No standard messages are defined.

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

655 **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

656 **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

657 **8.4.1.1 CIM_PortController.RequestStateChange() Supported**

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

663 **8.5 Profile Conventions for Operations**

664 Support for operations for each profile class (including associations) is specified in the following
665 subclauses. Each subclause includes either a statement "All operations are supported as described by
666 [DSP0200 version 1.2](#)" or a table listing all of the operations that are not supported by this profile or where
667 the profile requires behavior other than that described by [DSP0200](#).

668 The default list of operations is as follows:

- 669 • GetInstance
- 670 • Associators
- 671 • AssociatorNames
- 672 • References
- 673 • ReferenceNames
- 674 • EnumerateInstances
- 675 • EnumerateInstanceNames

676 A compliant implementation shall support all of the operations in the default list for each class, unless the
677 "Requirement" column states something other than *Mandatory*.

678 **8.6 CIM_ControlledBy**

679 Table 10 lists operations that either have special requirements beyond those from [DSP0200](#) or shall not
680 be supported. All other operations may be supported as defined in [DSP0200](#).

681 **Table 10 – Operations: CIM_ControlledBy**

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

682 **8.7 CIM_ElementCapabilities**

683 Table 11 lists operations that either have special requirements beyond those from [DSP0200](#) or shall not
684 be supported. All other operations may be supported as defined in [DSP0200](#).

685 **Table 11 – Operations: CIM_ElementCapabilities**

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

686 **8.8 CIM_EnabledLogicalElementCapabilities**

687 All operations in the default list in section 8.5 are supported as described by [DSP0200 version 1.2](#).

688 **8.9 CIM_HostedAccessPoint**

689 All operations in the default list in section 8.5 are supported as described by [DSP0200 version 1.2](#).

690 **EXPERIMENTAL**

691 **8.10 CIM_HostedService**

692 Table 12 lists operations that either have special requirements beyond those from [DSP0200](#) or shall not
693 be supported. All other operations may be supported as defined in [DSP0200](#).

694 **Table 12 – Operations: CIM_HostedService**

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

695 **EXPERIMENTAL**

696 8.11 CIM_LANEndpoint

697 Table 13 lists operations that either have special requirements beyond those from [DSP0200](#) or shall not
698 be supported. All other operations may be supported as defined in [DSP0200](#).

699 **Table 13 – Operations: CIM_LANEndpoint**

Operation	Requirement	Messages
DeleteInstance	Optional. See section 8.11.1.	None
ModifyInstance	Optional. See section 8.11.2.	None

700 8.11.1 CIM_LANEndpoint—DeleteInstance

701 An implementation may support the DeleteInstance operation for instances of CIM_LANEndpoint. When
702 the implementation supports the DeleteInstance operation, it may support the operation for some or all of
703 the CIM_LANEndpoint instances implemented. When the DeleteInstance operation is supported for an
704 instance of CIM_LANEndpoint, the implementation shall delete the instance of CIM_LANEndpoint and the
705 instances of CIM_DeviceSAPImplementation and CIM_HostedAccessPoint that reference the
706 CIM_LANEndpoint instance.

707 The implementation shall not support the DeleteInstance operation for the CIM_LANEndpoint instance
708 that is identified in section 7.2.1.

709 8.11.2 CIM_LANEndpoint—ModifyInstance

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

712 8.11.2.1 CIM_LANEndpoint.MACAddress

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

715 8.11.2.2 CIM_LANEndpoint.ElementName

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

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

727 8.12 CIM_NetworkPort

728 Table 14 lists operations that either have special requirements beyond those from [DSP0200](#) or shall not
729 be supported. All other operations may be supported as defined in [DSP0200](#).

730 **Table 14 – Operations: CIM_NetworkPort**

Operation	Requirement	Messages
ModifyInstance	Optional. See section 8.12.1.1.	None

731 8.12.1 CIM_NetworkPort—ModifyInstance Operation

732 This section details the specific requirements for the ModifyInstance operation that is applied to an
733 instance of CIM_NetworkPort.

734 8.12.1.1 CIM_NetworkPort.ElementName

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

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

746 EXPERIMENTAL

747 8.13 CIM_NetworkPortConfigurationService

748 All operations in the default list in section 8.5 are supported as described by [DSP0200 version 1.2](#).

749 EXPERIMENTAL

750 8.14 CIM_PortController

751 Table 10 lists operations that either have special requirements beyond those from [DSP0200](#) or shall not
752 be supported. All other operations may be supported as defined in [DSP0200](#).

753 **Table 15 – Operations: CIM_PortController**

Operation	Requirement	Messages
ModifyInstance	Optional. See section 8.14.1.1.	None

754 8.14.1 CIM_PortController—ModifyInstance Operation

755 This section details the specific requirements for the ModifyInstance operation that is applied to an
756 instance of CIM_PortController.

757 8.14.1.1 CIM_PortController.ElementName property

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

764 When an instance of CIM_EnabledLogicalElementCapabilities is not associated with the
 765 CIM_PortController instance, or the ElementNameEditSupported property of the
 766 CIM_EnabledLogicalElementCapabilities instance has a value of FALSE, the implementation shall not
 767 allow the ModifyInstance operation to change the value of the ElementName property of the
 768 CIM_PortController instance.

769 EXPERIMENTAL

770 8.15 CIM_ServiceAffectsElement

771 Table 16 lists operations that either have special requirements beyond those from [DSP0200](#) or shall not
 772 be supported. All other operations may be supported as defined in [DSP0200](#).

773 **Table 16 – Operations: CIM_ServiceAffectsElement**

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

774 EXPERIMENTAL

775 8.16 CIM_SystemDevice

776 Table 17 lists operations that either have special requirements beyond those from [DSP0200](#) or shall not
 777 be supported. All other operations may be supported as defined in [DSP0200](#).

778 **Table 17 – Operations: CIM_SystemDevice**

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

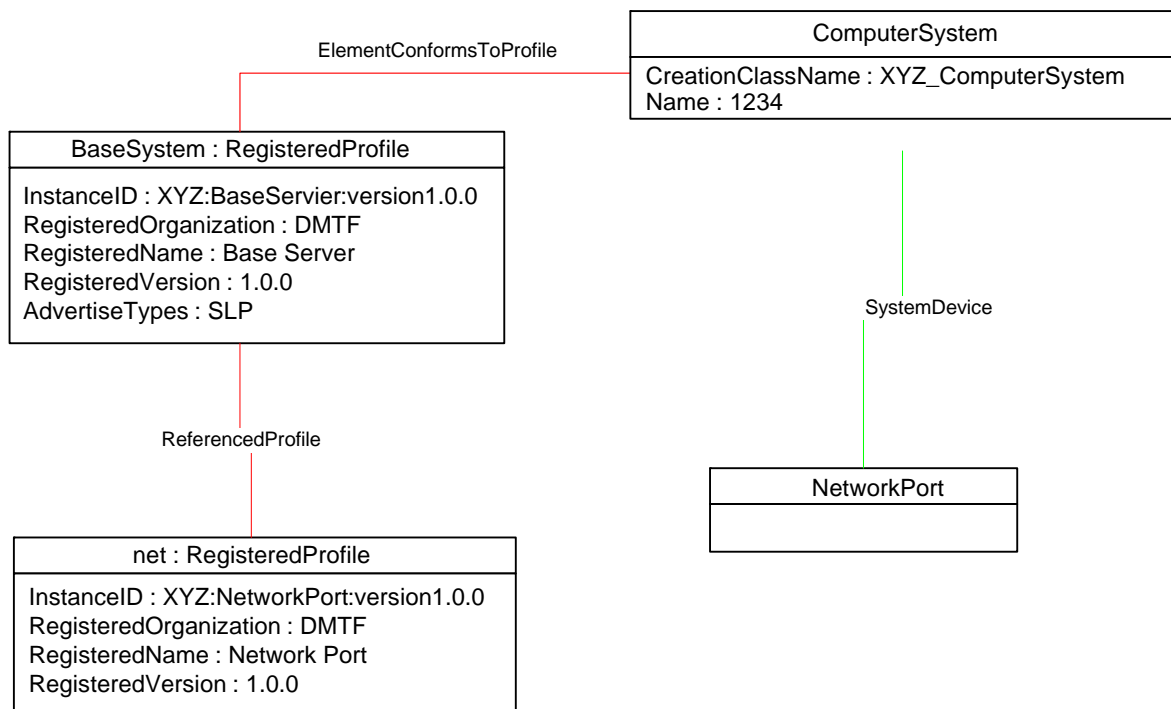
779 **9 Use Cases**

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

781 **9.1 Object Diagrams**

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

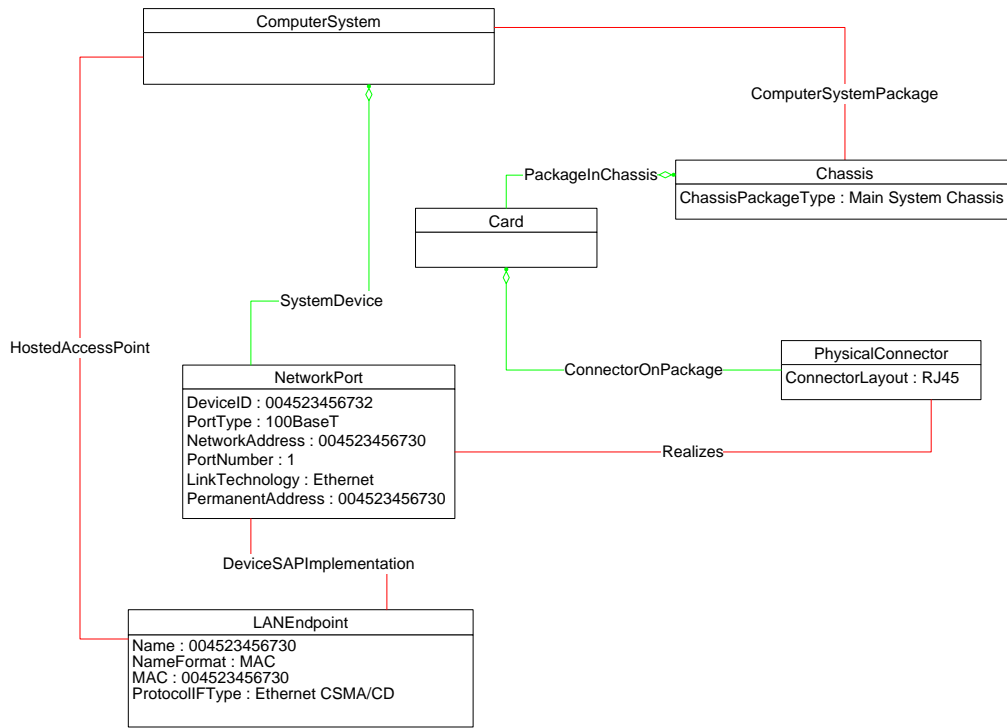
787 The CIM_NetworkPort instance is scoped to an instance of CIM_ComputerSystem. This instance of
 788 CIM_ComputerSystem is conformant with the DMTF *Base Server Profile*, version 1.0.0 as indicated by
 789 the CIM_ElementConformsToProfile association to the CIM_RegisteredProfile instance. The Scoping
 790 Instance in Figure 2 is the CIM_ComputerSystem instance. The Central Instance is the CIM_NetworkPort.
 791 The CIM_ReferencedProfile relationship between *BaseSystem* and *net* places the CIM_NetworkPort
 792 instance within the scope of *net*. Thus, the CIM_NetworkPort instance is conformant with the *Host LAN*
 793 *Network Port Profile*, version 1.0.0.



794

795 **Figure 2 – Registered Profile**

796 Figure 3 is a simple object diagram for a single network port with a single active network interface. The
 797 network port is represented by an instance of CIM_NetworkPort. The active interface is represented by an
 798 instance of CIM_LANEndpoint, which is associated with the CIM_NetworkPort instance through the
 799 CIM_DeviceSAPImplementation association. In the system modeled, the network port is reached through
 800 an RJ-45 connector located directly on the motherboard of the system. This connection is indicated by the
 801 CIM_Realizes association between the CIM_NetworkPort instance and the CIM_PhysicalConnector
 802 instance.



803

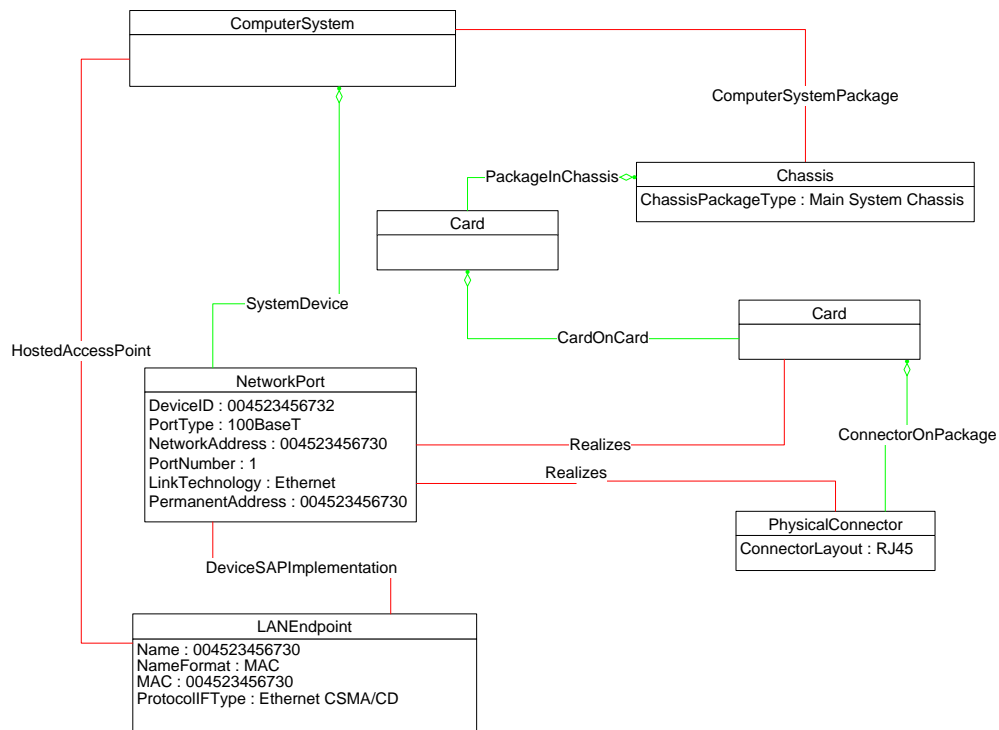
804

Figure 3 – Single Interface

805

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

806

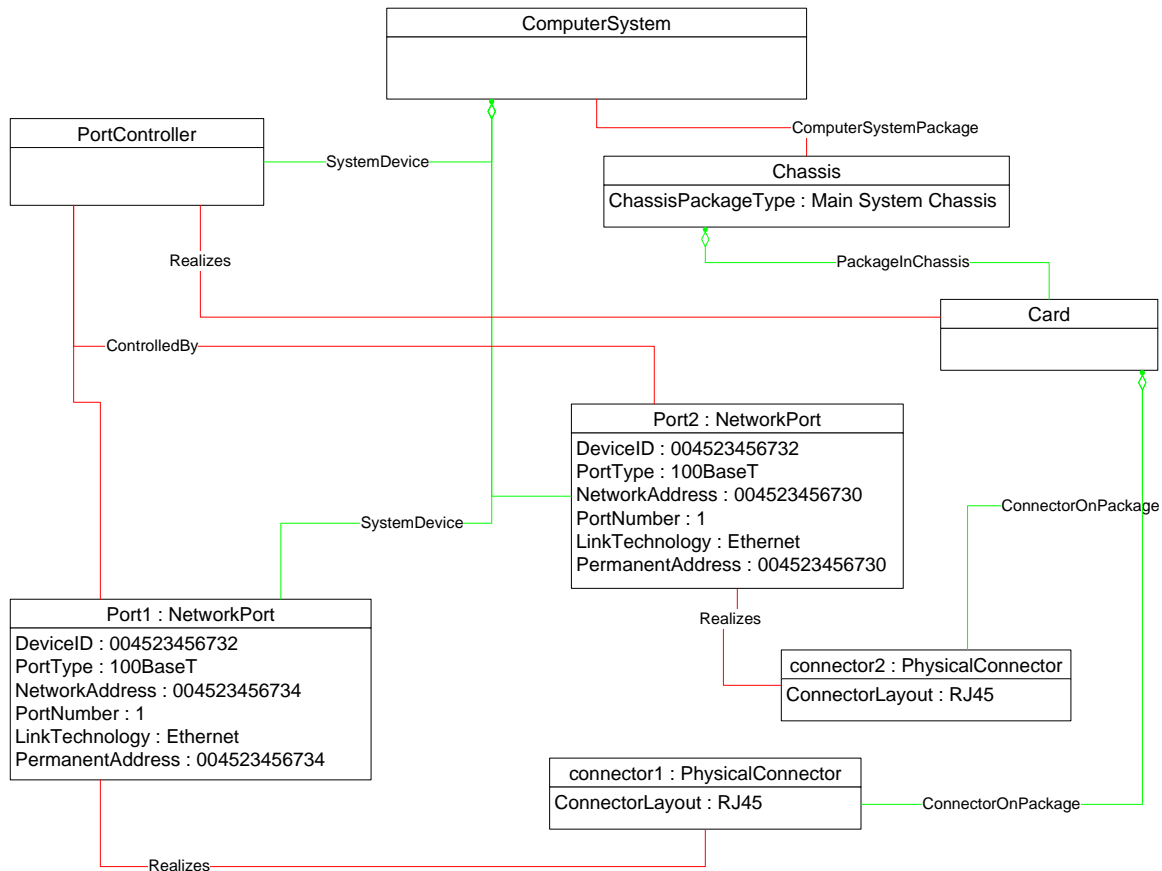


807

808

Figure 4 – Single Interface, Separate Card

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



814

815

Figure 5 – One Controller for Two Ports

816 **9.2 Querying MAC Address for an Interface**

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

- 818 1) Find all instances of CIM_LANEndpoint that are associated with the CIM_NetworkPort instance
- 819 through instances of CIM_DeviceSAPImplementation.
- 820 2) Query the MACAddress property of each instance of CIM_LANEndpoint.

821 **9.3 Determining Physical Connector for a Network Address**

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

- 824 1) Find the instance of CIM_NetworkPort that is associated with the CIM_LANEndpoint instance
- 825 through an instance of CIM_DeviceSAPImplementation.
- 826 2) Find the instance of CIM_PhysicalConnector that is associated with the CIM_NetworkPort
- 827 instance through an instance of CIM_Realizes.

828 **9.4 Determining If Physical Communication Is Possible**

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

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

832 **9.5 Correlating Controller and Port**

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

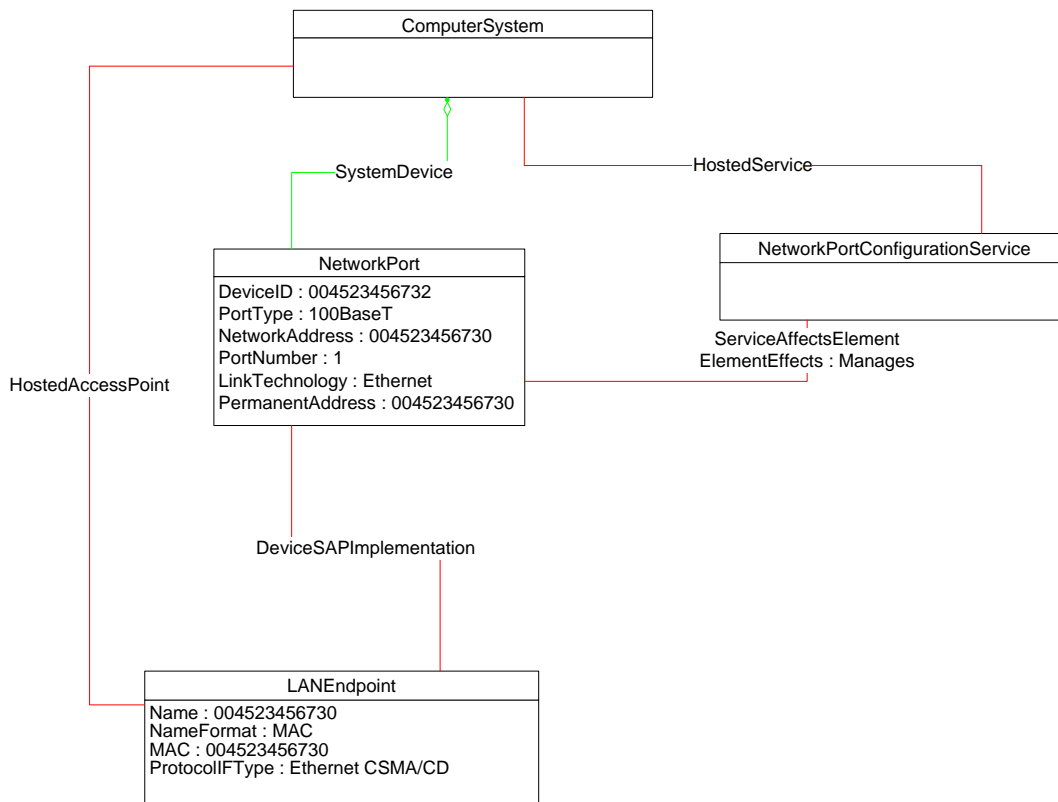
835 Find the instance of CIM_PortController that is associated with the CIM_NetworkPort instance
 836 through an instance of CIM_ControlledBy.

837 **EXPERIMENTAL**

838 **9.6 Adding an Endpoint to the Port**

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

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

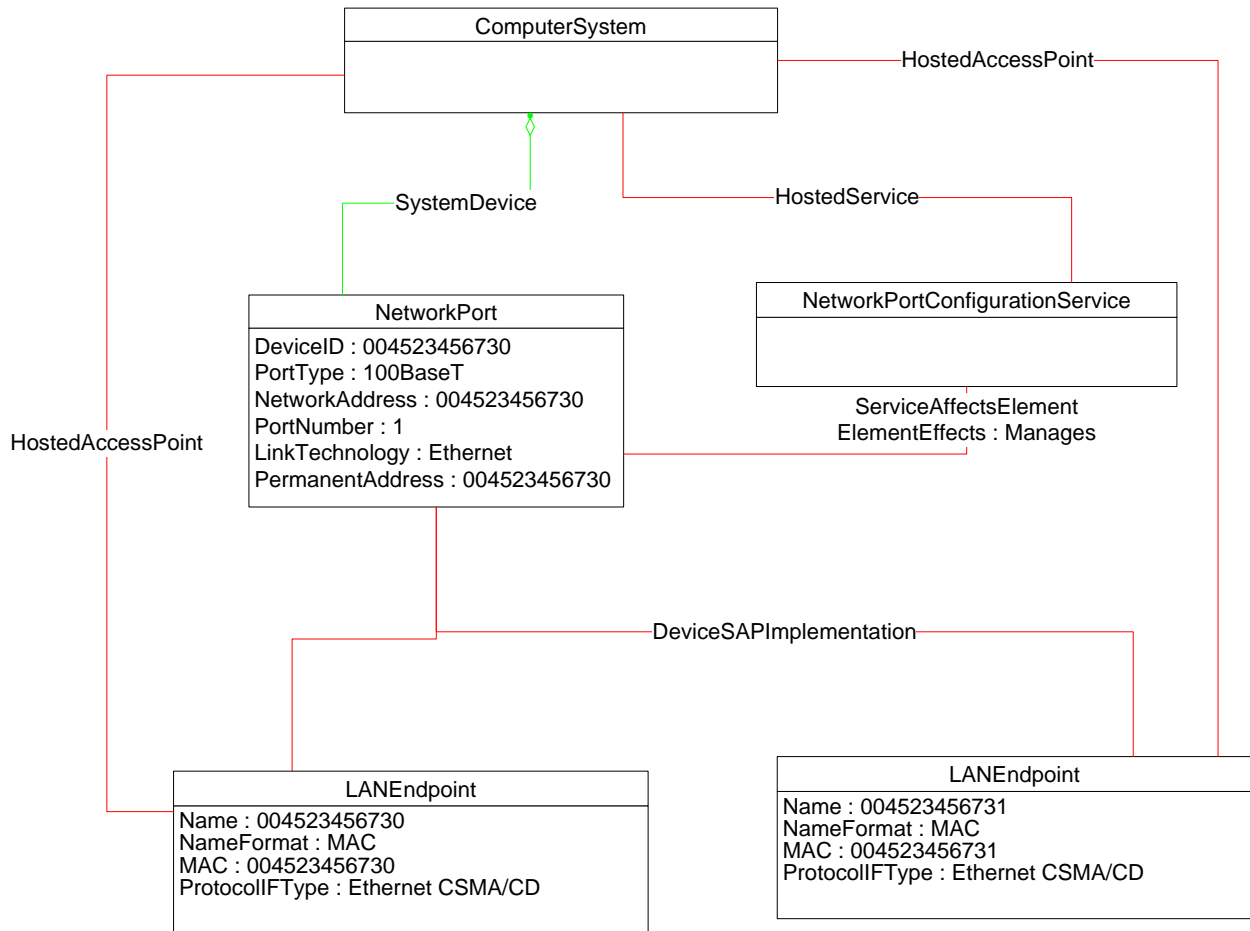


846

847

Figure 6 – Endpoint Management Supported

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



852

853

Figure 7 – Second Endpoint Added

854 **EXPERIMENTAL**

855 **9.7 Determining If ElementName Can Be Modified**

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

- 858 1) Find the CIM_EnabledLogicalElementCapabilities instance that is associated with the target
 859 instance.
- 860 2) Query the value of the ElementNameEditSupported property of the
 861 CIM_EnabledLogicalElementCapabilities instance. If the value is TRUE, the client can modify
 862 the ElementName property of the target instance.

863 9.8 Determining If State Management Is Supported

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

- 866 1) Find the CIM_EnabledLogicalElementCapabilities instance that is associated with the
867 CIM_LANEndpoint instance.
- 868 2) Query the value of the RequestedStatesSupported property. If at least one value is specified,
869 state management is supported.

870 10 CIM Elements

871 Table 18 shows the instances of CIM Elements for this profile. Instances of the CIM Elements shall be
872 implemented as described in Table 18. Sections 7 (“Implementation Requirements”) and 8 (“Methods”)
873 may impose additional requirements on these elements.

874 **Table 18 – CIM Elements: Network Port Profile**

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

875 10.1 CIM_ControlledBy

876 CIM_ControlledBy is used to associate an instance of CIM_NetworkPort with the instance of
877 CIM_PortController that controls the port, if the port controller is modeled.

878 **Table 19 – Class: CIM_ControlledBy**

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

879 **10.2 CIM_DeviceSAPImplementation**

880 CIM_DeviceSAPImplementation is used to associate the CIM_LANEndpoint instance with the
 881 CIM_NetworkPort instance that provides the network access.

882 **Table 20 – Class: CIM_DeviceSAPImplementation**

Properties	Requirement	Notes
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..*

883 **10.3 CIM_ElementCapabilities–LANEndpoint**

884 CIM_ElementCapabilities is used to associate an instance of CIM_EnabledLogicalElementCapabilities
 885 with an instance of CIM_LANEndpoint.

886 **Table 21 – Class: CIM_ElementCapabilities–LANEndpoint**

Properties	Requirement	Notes
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

887 **10.4 CIM_ElementCapabilities–NetworkPort**

888 CIM_ElementCapabilities is used to associate an instance of CIM_EnabledLogicalElementCapabilities
 889 with an instance of CIM_NetworkPort.

890 **Table 22 – Class: CIM_ElementCapabilities–NetworkPort**

Properties	Requirement	Notes
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

891 **10.5 CIM_ElementCapabilities–PortController**

892 CIM_ElementCapabilities is used to associate an instance of CIM_EnabledLogicalElementCapabilities
 893 with an instance of CIM_PortController.

894 **Table 23 – Class: CIM_ElementCapabilities–PortController**

Properties	Requirement	Notes
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

895 **10.6 CIM_EnabledLogicalElementCapabilities–LANEndpoint**

896 CIM_EnabledLogicalElementCapabilities is used to indicate support for managing the state of the network
 897 interface.

898 **Table 24 – Class: CIM_EnabledLogicalElementCapabilities–LANEndpoint**

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

899 **10.7 CIM_EnabledLogicalElementCapabilities–NetworkPort**

900 CIM_EnabledLogicalElementCapabilities is used to indicate support for managing the state of the network
 901 port.

902 **Table 25 – Class: CIM_EnabledLogicalElementCapabilities–NetworkPort**

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

903 **10.8 CIM_EnabledLogicalElementCapabilities–PortController**

904 CIM_EnabledLogicalElementCapabilities is used to indicate support for managing the state of the port
 905 controller.

906 **Table 26 – Class: CIM_EnabledLogicalElementCapabilities–PortController**

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

907 **10.9 CIM_HostedAccessPoint**

908 CIM_HostedAccessPoint is used to relate a CIM_LANEndpoint instance to its scoping
 909 CIM_ComputerSystem instance.

910 **Table 27 – Class: CIM_HostedAccessPoint**

Properties	Requirement	Notes
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..*

911 **EXPERIMENTAL**

912 **10.10 CIM_HostedService**

913 CIM_HostedService is used to associate the CIM_NetworkPortConfigurationService instance with the
 914 CIM_ComputerSystem instance to which it is scoped.

915 **Table 28 – Class: CIM_HostedService**

Properties	Requirement	Notes
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 *

916 **EXPERIMENTAL**

917 **10.11 CIM_LANEndpoint**

918 CIM_LANEndpoint represents a MAC address to which the network port will respond on the LAN.

919 **Table 29 – Class: CIM_LANEndpoint**

Properties and Methods	Requirement	Notes
SystemCreationClassName	Mandatory	None
CreationClassName	Mandatory	None
SystemName	Mandatory	None
Name	Mandatory	None
NameFormat	Mandatory	None
ProtocolIFType	Mandatory	None
MACAddress	Mandatory	This property shall be formatted as 12 unseparated case-insensitive hex digits. (pattern "[0123456789ABCDEFabcdef]{12}\$")
LANID	Optional	See section 8.1.
AliasAddresses	Optional	See section 8.1.
GroupAddresses	Optional	See section 8.1.
RequestedState	Mandatory	See sections 7.2.4.2 and 7.2.5.2.
EnabledState	Mandatory	See sections 7.2.5.3 and 7.2.4.3.
ElementName	Mandatory	See sections 7.2.6 and 7.2.7.
RequestStateChange()	Conditional	See section 8.3.

920 **10.12 CIM_NetworkPort**

921 CIM_NetworkPort represents the hardware and device aspects of a physical network interface.

922 **Table 30 – Class: CIM_NetworkPort**

Properties and Methods	Requirement	Notes
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 sections 7.1.2.3 and 7.1.3.3.
RequestedState	Mandatory	See sections 7.1.2.2 and 7.1.3.2.
ElementName	Mandatory	See sections 7.1.4 and 7.1.5.
RequestStateChange()	Conditional	See section 8.2.

923 **EXPERIMENTAL**

924 **10.13 CIM_NetworkPortConfigurationService**

925 CIM_NetworkPortConfigurationService represents the ability to add endpoints to the network port.

926 **Table 31 – Class: NetworkPortConfigurationService**

Properties and Methods	Requirement	Notes
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 section 8.1.

927 **EXPERIMENTAL**

928 **10.14 CIM_PhysicalConnector**

929 CIM_PhysicalConnector is used to represent the physical connector that connects the network port to the
 930 physical network. This class is defined by the *Physical Asset Profile*. The behavior specified in Table 32 is
 931 in addition to that specified by the *Physical Asset Profile*.

932 **Table 32 – Class: CIM_PhysicalConnector**

Properties	Requirement	Notes
ConnectorLayout	Mandatory	None

933 **10.15 CIM_PortController**

934 CIM_PortController represents a network controller.

935 **Table 33 – Class: CIM_PortController**

Properties	Requirement	Notes
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 section 7.4.3.3.
RequestedState	Mandatory	See section 7.4.3.2.
ElementName	Mandatory	See sections 7.4.5 and 7.4.6.
RequestStateChange()	Conditional	See section 8.4.

936 **10.16 CIM_Realizes**

937 The CIM_Realizes association is used to associate the CIM_NetworkPort with an instance of
 938 CIM_PhysicalConnector when an instance of CIM_PhysicalConnector is instrumented. This class is
 939 defined by the *Physical Asset Profile*. The behavior specified in Table 34 is in addition to that specified by
 940 the *Physical Asset Profile*.

941 **Table 34 – Class: CIM_Realizes**

Properties	Requirement	Notes
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..*

942 **10.17 CIM_RegisteredProfile**

943 CIM_RegisteredProfile identifies the *Host LAN Network Port Profile* in order for a client to determine
 944 whether an instance of CIM_LogicalModule is conformant with this profile. The CIM_RegisteredProfile
 945 class is defined by the *Profile Registration Profile*. With the exception of the mandatory values specified
 946 for the properties in Table 35, the behavior of the CIM_RegisteredProfile instance is in accordance with
 947 the constraints specified in the *Profile Registration Profile*.

948 **Table 35 – 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 "1.0.0".
OwningEntity	Mandatory	This property shall have a value of "DMTF".

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

952 **EXPERIMENTAL**

953 **10.18 CIM_ServiceAffectsElement**

954 CIM_ServiceAffectsElement is used to associate an instance of CIM_NetworkPortConfigurationService
 955 with an instance of CIM_NetworkPort that the service is able to configure.

956 **Table 36 – 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)

957 **EXPERIMENTAL**

958 **10.19 CIM_SystemDevice–CIM_NetworkPort**

959 CIM_SystemDevice is used to associate an instance of CIM_NetworkPort with the instance of
 960 CIM_ComputerSystem to which the CIM_NetworkPort is scoped.

961 **Table 37 – Class: CIM_SystemDevice**

Properties	Requirement	Notes
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..*

962 **10.20 CIM_SystemDevice–CIM_PortController**

963 CIM_SystemDevice is used to associate an instance of CIM_PortController with an instance of
964 CIM_ComputerSystem when CIM_PortController is implemented.

965 **Table 38 – Class: CIM_SystemDevice**

Properties	Requirement	Notes
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 *

966

967
968
969
970

ANNEX A
(informative)
Change Log

Version	Date	Description
1.0.0b	04/11/2006	Incorporated Company Review feedback.
1.0.0	06/03/2008	Final Standard

971
972
973
974

ANNEX B (informative)

Acknowledgments

975 The authors wish to acknowledge the following people.

976 **Editors:**

- 977 • Aaron Merkin – IBM
- 978 • Jeff Hilland – HP

979 **Contributors:**

- 980 • Jon Hass – Dell
- 981 • Khachatur Papanyan – Dell
- 982 • Enoch Suen – Dell
- 983 • Jeff Hilland – HP
- 984 • Christina Shaw – HP
- 985 • Aaron Merkin – IBM
- 986 • Perry Vincent – Intel
- 987 • John Leung – Intel
- 988