



1

2

3

4

Document Number: DSP1036

Date: 2010-10-21

Version: 1.1.0

5 **IP Interface Profile**

6 **Document Type: Specification**

7 **Document Status: DMTF Standard**

8 **Document Language: en-US**

9

10 Copyright Notice

11 Copyright © 2008, 2010 Distributed Management Task Force, Inc. (DMTF). All rights reserved.

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

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26 withdrawn or modified after publication, and shall be indemnified and held harmless by any party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

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

32

CONTENTS

34	Foreword	6
35	Introduction	7
36	1 Scope	9
37	2 Normative References.....	9
38	3 Terms and Definitions	9
39	4 Symbols and Abbreviated Terms	11
40	5 Synopsis.....	11
41	6 Description	12
42	6.1 Pending and Alternate Configuration Management.....	13
43	7 Implementation.....	13
44	7.1 Basic IP Configuration	13
45	7.2 DHCP Client Is Supported	17
46	7.3 DNS Client Is Supported.....	17
47	7.4 Managing Alternate Configurations — Optional	17
48	7.5 Applying an Alternate Configuration	20
49	7.6 Relationship with a Network Interface	22
50	8 Methods.....	22
51	8.1 CIM_IPProtocolEndpoint.RequestStateChange()	23
52	8.2 CIM_IPConfigurationService.ApplySettingToIPProtocolEndpoint()	23
53	8.3 Profile Conventions for Operations.....	24
54	8.4 CIM_BindsToLANEndpoint.....	25
55	8.5 CIM_ElementSettingData	25
56	8.6 CIM_HostedAccessPoint	26
57	8.7 CIM_HostedService	26
58	8.8 CIM_IPAssignmentSettingData	27
59	8.9 CIM_IPConfigurationService	27
60	8.10 CIM_IPProtocolEndpoint	27
61	8.11 CIM_OrderedComponent	28
62	8.12 CIM_RemoteAccessAvailableToElement.....	28
63	8.13 CIM_RemoteServiceAccessPoint.....	28
64	8.14 CIM_ServiceAffectsElement	29
65	8.15 CIM_StaticIPAssignmentSettingData	29
66	8.16 CIM_SystemDevice	29
67	9 Use Cases	30
68	9.1 Miscellaneous Object Diagrams	30
69	9.2 Determine Supported Configuration Methods	43
70	9.3 Determine Gateway Address.....	43
71	9.4 Determine Method Used for Current Configuration	43
72	9.5 Determine Whether DHCP Then Static Is Supported.....	43
73	9.6 View Default Configuration	43
74	9.7 Configure the Interface to Use DHCP.....	44
75	9.8 Establish a Static IP Configuration for an Interface	44
76	9.9 Apply a Pending Configuration — Synchronously.....	44
77	9.10 Apply a Pending Configuration — Upon Restart	45
78	9.11 Determine Whether DNS Configuration Was DHCP Assigned	45
79	9.12 Determine Whether ElementName Can Be Modified	45
80	9.13 Determine Whether State Management Is Supported.....	45
81	10 CIM Elements.....	46
82	10.1 CIM_BindsToLANEndpoint.....	46
83	10.2 CIM_ElementCapabilities	47
84	10.3 CIM_ElementSettingData — CIM_IPAssignmentSettingData Reference.....	47
85	10.4 CIM_ElementSettingData — CIM_StaticIPAssignmentSettingData Reference.....	47

86	10.5	CIM_EnabledLogicalElementCapabilities.....	48
87	10.6	CIM_HostedAccessPoint — CIM_RemoteServiceAccessPoint Reference	48
88	10.7	CIM_HostedAccessPoint — CIM_IPProtocolEndpoint Reference	49
89	10.8	CIM_HostedService	49
90	10.9	CIM_IPAssignmentSettingData	49
91	10.10	CIM_IPConfigurationService	50
92	10.11	CIM_IPProtocolEndpoint	50
93	10.12	CIM_OrderedComponent	50
94	10.13	CIM_RegisteredProfile.....	51
95	10.14	CIM_RemoteAccessAvailableToElement.....	51
96	10.15	CIM_RemoteServiceAccessPoint.....	52
97	10.16	CIM_ServiceAffectsElement.....	52
98	10.17	CIM_StaticIPAssignmentSettingData	52
99		ANNEX A (informative) Change Log.....	54

100

101 Figures

102	Figure 1 – IP Interface Profile: Class Diagram.....	12
103	Figure 2 – Registered Profile	30
104	Figure 3 – Basic Configuration — IPv4.....	31
105	Figure 4 – Basic Configuration — IPv6.....	32
106	Figure 5 – Basic Configuration — IPv4 and IPv6	33
107	Figure 6 – Static Current and Pending Configuration	34
108	Figure 7 – Static and DHCP Pending Configurations	35
109	Figure 8 – DHCP Timed Out to a Static Configuration	36
110	Figure 9 – Service Processor and Server Share an NIC	37
111	Figure 10 – Configuration Choices	38
112	Figure 11 – DHCP Assigned Partial DNS	39
113	Figure 12 – DHCP with DNS Statically Configured	40
114	Figure 13 – Static without DNS Configuration — One.....	41
115	Figure 14 – Static without DNS Configuration — Two.....	42
116	Figure 15 – Static without DNS Configuration — Three	42

117

118 Tables

119	Table 1 – Referenced Profiles	11
120	Table 2 – CIM_IPProtocolEndpoint.RequestStateChange() Method: Return Code Values.....	23
121	Table 3 – CIM_IPProtocolEndpoint.RequestStateChange() Method: Parameters.....	23
122	Table 4 – CIM_IPConfigurationService.ApplySettingToIPProtocolEndpoint() Method: Return Code	
123	Values	24
124	Table 5 – CIM_IPConfigurationService.ApplySettingToIPProtocolEndpoint() Method: Parameters.....	24
125	Table 6 – Operations: CIM_BindsToLANEndpoint	25
126	Table 7 – Operations: CIM_ElementSettingData.....	25
127	Table 8 – Operations: CIM_HostedAccessPoint	26
128	Table 9 – Operations: CIM_HostedService	27
129	Table 10 – Operations: CIM_IPProtocolEndpoint.....	27
130	Table 11 – Operations: CIM_OrderedComponent.....	28
131	Table 12 – Operations: CIM_RemoteAccessAvailableToElement	28

132 Table 13 – Operations: CIM_ServiceAffectsElement 29

133 Table 14 – Operations: CIM_StaticIPAssignmentSettingData..... 29

134 Table 15 – Operations: CIM_SystemDevice..... 29

135 Table 16 – CIM Elements: IP Interface Profile..... 46

136 Table 17 – Class: CIM_BindsToLANEndpoint..... 46

137 Table 18 – Class: CIM_ElementCapabilities..... 47

138 Table 19 – Class: CIM_ElementSettingData — CIM_IPAssignmentSettingData..... 47

139 Table 20 – Class: CIM_ElementSettingData — CIM_StaticIPAssignmentSettingData..... 47

140 Table 21 – Class: CIM_EnabledLogicalElementCapabilities..... 48

141 Table 22 – Class: CIM_HostedAccessPoint — CIM_RemoteServiceAccessPoint 48

142 Table 23 – Class: CIM_HostedAccessPoint — CIM_IPProtocolEndpoint..... 49

143 Table 24 – Class: CIM_HostedService 49

144 Table 25 – Class: CIM_IPAssignmentSettingData 49

145 Table 26 – Class: CIM_IPConfigurationService..... 50

146 Table 27 – Class: CIM_IPProtocolEndpoint..... 50

147 Table 28 – Class: CIM_OrderedComponent..... 51

148 Table 29 – Class: CIM_RegisteredProfile 51

149 Table 30 – Class: CIM_RemoteAccessAvailableToElement 51

150 Table 31 – Class: CIM_RemoteServiceAccessPoint..... 52

151 Table 32 – Class: CIM_ServiceAffectsElement 52

152 Table 33 – Class: CIM_StaticIPAssignmentSettingData 52

153

154

Foreword

155 The *IP Interface Profile* (DSP1036) was prepared by the Server Management Working Group and the
156 Physical Platform Profiles Working Group of the DMTF.

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

159 Acknowledgments

160 The authors wish to acknowledge the following people.

161 Editors:

- 162 • Hemal Shah – Broadcom
- 163 • Jeff Hilland – HP
- 164 • Aaron Merkin – IBM

165 Contributors:

- 166 • Hemal Shah – Broadcom
- 167 • RadhaKrishna Dasari – Dell
- 168 • Jon Hass – Dell
- 169 • Khachatur Papanyan – Dell
- 170 • Enoch Suen – Dell
- 171 • Jeff Hilland – HP
- 172 • Christina Shaw – HP
- 173 • Aaron Merkin – IBM
- 174 • Perry Vincent – Intel
- 175 • John Leung – Intel

176

177

Introduction

178 The information in this specification should be sufficient for a provider or consumer of this data to identify
179 unambiguously the classes, properties, methods, and values that shall be instantiated and manipulated to
180 represent and manage an IP interface and its associated configuration information. The target audience
181 for this specification is implementers who are writing CIM-based providers or consumers of management
182 interfaces that represent the component described in this document.

183

IP Interface Profile

184 1 Scope

185 The *IP Interface Profile* extends the management capability of referencing profiles by adding the
186 capability to represent an IP interface of a managed system. This profile includes a specification of the IP
187 interface, its associated IP configuration, optional support for managing pending configurations, optional
188 support for the relationship with a DNS client, and optional support for the relationship with a DHCP client.

189 2 Normative References

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

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

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

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

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

201 DMTF DSP1035, *Host LAN Network Port Profile 1.0*,
202 http://www.dmtf.org/standards/published_documents/DSP1035_1.0.pdf

203 DMTF DSP1037, *DHCP Client Profile 1.0*,
204 http://www.dmtf.org/standards/published_documents/DSP1037_1.0.pdf

205 DMTF DSP1038, *DNS Client Profile 1.0*,
206 http://www.dmtf.org/standards/published_documents/DSP1038_1.0.pdf

207 IETF, RFC 1208, *A Glossary of Networking Terms*, March 1991, <http://www.ietf.org/rfc/rfc1208.txt>

208 IETF, RFC 2131, *Dynamic Host Configuration Protocol*, March 1997, <http://www.ietf.org/rfc/rfc2131.txt>

209 IETF, RFC 4291, *IP Version 6 Addressing Architecture*, February 2006, <http://www.ietf.org/rfc/rfc4291.txt>

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

212 3 Terms and Definitions

213 For the purposes of this document, the terms and definitions in [DSP1033](#) and [DSP1001](#) as well as the
214 following definitions apply.

215 3.1

216 can

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

- 218 **3.2**
219 **cannot**
220 used for statements of possibility and capability, whether material, physical, or causal
- 221 **3.3**
222 **conditional**
223 indicates requirements to be followed strictly to conform to the document when the specified conditions
224 are met
- 225 **3.4**
226 **mandatory**
227 indicates requirements to be followed strictly to conform to the document and from which no deviation is
228 permitted
- 229 **3.5**
230 **may**
231 indicates a course of action permissible within the limits of the document
- 232 **3.6**
233 **need not**
234 indicates a course of action permissible within the limits of the document
- 235 **3.7**
236 **optional**
237 indicates a course of action permissible within the limits of the document
- 238 **3.8**
239 **pending configuration**
240 the configuration that will be applied to an IP interface the next time the interface accepts a configuration
- 241 **3.9**
242 **referencing profile**
243 indicates a profile that owns the definition of this class and can include a reference to this profile in its
244 "Referenced Profiles" table
- 245 **3.10**
246 **shall**
247 indicates requirements to be followed strictly to conform to the document and from which no deviation is
248 permitted
- 249 **3.11**
250 **shall not**
251 indicates requirements to be followed strictly to conform to the document and from which no deviation is
252 permitted
- 253 **3.12**
254 **should**
255 indicates that among several possibilities, one is recommended as particularly suitable, without
256 mentioning or excluding others, or that a certain course of action is preferred but not necessarily required
- 257 **3.13**
258 **should not**
259 indicates that a certain possibility or course of action is deprecated but not prohibited

260 **3.14**
 261 **unspecified**
 262 indicates that this profile does not define any constraints for the referenced CIM element or operation

263 4 Symbols and Abbreviated Terms

264 The following abbreviations are used in this document.

265 **4.1**
 266 **DHCP**
 267 Dynamic Host Configuration Protocol

268 **4.2**
 269 **DNS**
 270 Domain Name System

271 **4.3**
 272 **IP**
 273 Internet Protocol

274 5 Synopsis

275 **Profile Name:** IP Interface

276 **Version:** 1.1.0

277 **Organization:** DMTF

278 **CIM Schema Version:** 2.26

279 **Central Class:** CIM_IPProtocolEndpoint

280 **Scoping Class:** CIM_ComputerSystem

281 The *IP Interface Profile* extends the management capability of referencing profiles by adding the
 282 capability to represent an IP interface of a managed system. This profile includes a specification of the IP
 283 interface, its associated IP configuration, optional support for managing pending configurations, optional
 284 support for the relationship with a DNS client, and optional support for the relationship with a DHCP client.

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

286

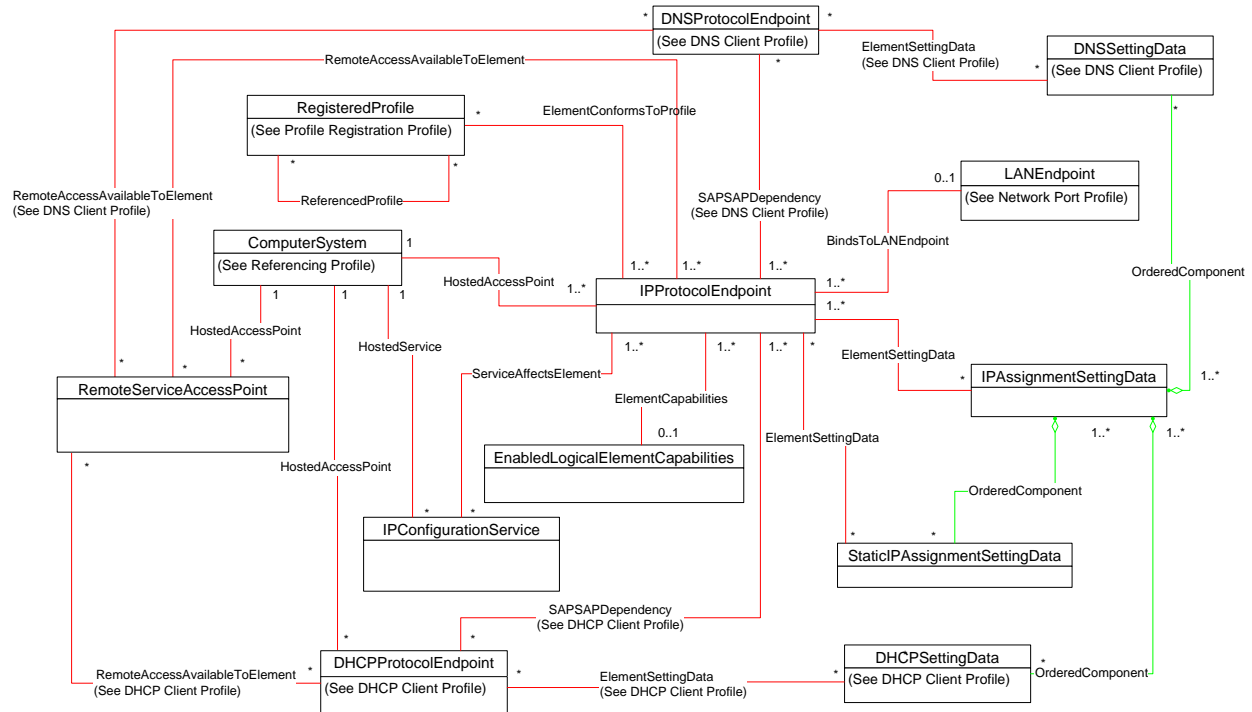
Table 1 – Referenced Profiles

Profile Name	Organization	Version	Requirement	Description
Profile Registration	DMTF	1.0	Mandatory	None
DNS Client	DMTF	1.0	Optional	See 7.3.
DHCP Client	DMTF	1.0	Optional	See 7.2.
Host LAN Network Port	DMTF	1.0	Optional	See 7.6.

287 6 Description

288 The *IP Interface Profile* describes an IP interface and associated IP configuration information in a
289 managed system.

290 Figure 1 represents the class schema for the *IP Interface Profile*. For simplicity, the CIM_ prefix has been
291 removed from the names of the classes. Note that this class diagram is meant to be used in conjunction
292 with the class diagrams from the *DHCP Client Profile* ([DSP1037](#)) and the *DNS Client Profile* ([DSP1038](#)).



293

294

Figure 1 – IP Interface Profile: Class Diagram

295 The *IP Interface Profile* extends the management capability of referencing profiles by adding the
296 capability to represent an IP interface in a managed system. Functionality within the scope of this profile
297 includes:

- 298 • IPv4 interface (optionally associated with a network interface)
- 299 • optional relationship with a DNS client
- 300 • optional relationship with a DHCP client
- 301 • current and pending configurations

302 Functionality explicitly excluded from the scope of this profile includes:

- 303 • modeling of the network gateway
- 304 • modeling of TCP/UDP ports

305 Any representation of network elements is purely from the perspective of the IP interface. That is, no
306 provisions are made for the modeling of network resources for the purposes of managing those
307 resources.

308 This profile represents the current configuration of an IP interface, associated configurations that could be
309 applied, the DNS client, and the DHCP client. Support for the DNS and DHCP clients is not required. In
310 general, the various subclasses of CIM_ProtocolEndpoint reflect the current configuration and status of
311 their respective elements.

312 Functionality provided by other systems (Gateway, DHCP server, and DNS server) is modeled from the
313 IP interface view and is therefore represented by instances of CIM_RemoteServiceAccessPoint.

314 **6.1 Pending and Alternate Configuration Management**

315 Pending configurations, which are associated with the IP interface and could be applied in the future, are
316 represented by instances of CIM_IPAssignmentSettingData and its subclasses. Each pending
317 configuration can include multiple settings that will be applied to the different elements of the endpoint
318 configuration. Settings for a particular element of the configuration are represented with the appropriate
319 subclass of CIM_IPAssignmentSettingData and aggregated into one or more instances of
320 CIM_IPAssignmentSettingData that represent the configuration.

321 The management of DNS and DHCP clients as part of an alternate configuration is handled differently for
322 the two clients. DHCP and static IP configuration management are generally treated as alternatives to
323 each other. For the basic configuration of an IP interface, the information is assigned either statically or
324 through DHCP. DNS configuration occurs differently. When DNS and static configuration occur together,
325 there is no overlap. Thus the DNS settings that are part of the configuration are applied to the DNS client.
326 When DHCP and DNS settings are used together, portions of the DNS configuration can potentially be
327 assigned through DHCP.

328 The intended usage model for alternate configurations is that an implementation presents a finite set of
329 alternate configurations. It is expected that an alternate configuration will be instrumented for each unique
330 ordering of static and DHCP assignment supported by the implementation. An alternate configuration can
331 also be provided for each unique configuration persisted (either in the instrumentation layer or underlying
332 modeled component). DNS configuration is presented as an optional aspect of each unique alternate
333 configuration with which DNS usage is supported.

334 **7 Implementation**

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

337 **7.1 Basic IP Configuration**

338 The basic configuration of the IP interface consists of the IP address, subnet mask, and default gateway.

339 **7.1.1 CIM_IPProtocolEndpoint**

340 An instance of CIM_IPProtocolEndpoint shall represent the IP interface. The properties of the instance of
341 CIM_IPProtocolEndpoint shall reflect the current configuration of an IP interface.

342 **7.1.1.1 CIM_IPProtocolEndpoint.AddressOrigin**

343 The AddressOrigin property indicates the configuration method that resulted in the configuration being
344 assigned to the CIM_IPProtocolEndpoint.

345 **7.1.1.1.1 AddressOrigin — Static**

346 A value of 3 (Static) shall indicate that the configuration was assigned statically. The AddressOrigin
347 property shall have a value of 3 (Static) when the configuration is the result of an instance of
348 CIM_StaticIPAssignmentSettingData being successfully applied. Clause 7.5.3.3 explains what it means
349 for settings to be successfully applied.

350 7.1.1.1.2 AddressOrigin — DHCP

351 A value of 4 (DHCP) shall indicate that the configuration was obtained through an associated DHCP
352 client. The AddressOrigin property shall have a value of 4 (DHCP) when the configuration is the result of
353 an instance of CIM_DHCPSettingData being successfully applied.

354 7.1.1.2 CIM_IPProtocolEndpoint.ProtocolIFType

355 The ProtocolIFType property shall indicate the current IP address type.

356 If the value is 4096 (IPv4) the IPv4Address and SubnetMask properties shall be implemented.

357 The value of CIM_IPProtocolEndpoint.ProtocolIFType shall be 4096, 4097, or 4098.

358 If the value is 4097 (Ipv6) the IPv6Address and IPv6SubnetPrefixLength properties shall be implemented
359 and IPv6AddressType may be implemented.

360 If the value is 4098 (Ipv4/Ipv6) the IPv6Address and IPv6SubnetPrefixLength properties shall be
361 implemented and IPv6AddressType may be implemented. If IPv6AddressType is implemented, the
362 IPv6AddressType shall be 7 (Embedded IPv4 Address).

363 7.1.1.3 CIM_IPProtocolEndpoint.IPv4Address

364 If the value of CIM_IPProtocolEndpoint.ProtocolIFType is 4096 (IPv4), the IPv4Address property shall
365 indicate the current IPv4 address assigned to this IP endpoint. The value of the property shall be
366 specified in dotted decimal notation as defined in IETF [RFC 1208](#). A value of 0.0.0.0 shall indicate that a
367 valid IP address is not assigned to this IP endpoint.

368 7.1.1.4 CIM_IPProtocolEndpoint.SubnetMask

369 If the value of CIM_IPProtocolEndpoint.ProtocolIFType is 4096 (IPv4), the SubnetMask property shall be
370 specified using dotted decimal notation as defined in IETF [RFC 1208](#). A value of 0.0.0.0 shall indicate
371 that a valid subnet mask is not assigned to this IP endpoint.

372 7.1.1.5 CIM_IPProtocolEndpoint.IPv6Address

373 If the value of CIM_IPProtocolEndpoint.ProtocolIFType is 4097 (IPv6) or 4098 (IPv4/IPv6), the
374 IPv6Address property shall indicate the current IPv6 address assigned to this IP endpoint. The value of
375 the property shall be specified in the notation specified in IETF [RFC 4291](#), section 2.2.

376 7.1.2 IP Interface State Management Is Supported — Conditional

377 When management of the state of an IP interface is supported, exactly one instance of
378 CIM_EnabledLogicalElementCapabilities shall be associated with the CIM_IPProtocolEndpoint instance
379 through an instance of CIM_ElementCapabilities. The existence of the CIM_ElementCapabilities instance
380 is conditional on the existence of the CIM_EnabledLogicalElementCapabilities instance.

381 Support for managing the state of the IP interface is optional behavior. This clause describes the CIM
382 elements and behaviors that shall be implemented when this behavior is supported.

383 7.1.2.1 CIM_EnabledLogicalElementCapabilities

384 The instance of CIM_EnabledLogicalElementCapabilities is used to advertise the state management
385 supported for the IP interface.

386 7.1.2.1.1 CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported

387 The RequestedStatesSupported property may contain zero or more of the following values: 2 (Enabled),
388 3 (Disabled), or 11 (Reset).

389 **7.1.2.2 CIM_IPProtocolEndpoint.RequestedState**

390 When the CIM_IPProtocolEndpoint.RequestStateChange() method is successfully invoked, the value of
391 the RequestedState property shall be the value of the RequestedState parameter. If the method is not
392 successfully invoked, the value of the RequestedState property is indeterminate.

393 The CIM_IPProtocolEndpoint.RequestedState property shall have one of the values specified in the
394 CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported property or a value of 5 (No
395 Change).

396 **7.1.2.3 CIM_IPProtocolEndpoint.EnabledState**

397 When the RequestedState parameter has a value of 2 (Enabled) or 3 (Disabled) and the
398 CIM_IPProtocolEndpoint.RequestStateChange() method completes successfully, the value of the
399 EnabledState property shall equal the value of the CIM_IPProtocolEndpoint.RequestedState property.

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

401 The EnabledState property shall have one of the following values: 2 (Enabled), 3 (Disabled), or 6
402 (Enabled but Offline).

403 **7.1.3 IP Interface State Management Is Not Supported**

404 This clause describes the CIM elements and behaviors that shall be implemented when management of
405 the IP Interface state is not supported.

406 **7.1.3.1 CIM_EnabledLogicalElementCapabilities**

407 When state management is not supported, exactly one instance of
408 CIM_EnabledLogicalElementCapabilities may be associated with the CIM_IPProtocolEndpoint instance
409 through an instance of CIM_ElementCapabilities.

410 **7.1.3.1.1 CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported**

411 The CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported property shall not contain any
412 values.

413 **7.1.3.2 CIM_IPProtocolEndpoint.RequestedState**

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

415 **7.1.3.3 CIM_IPProtocolEndpoint.EnabledState**

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

418 **7.1.4 Modifying ElementName Is Supported — Conditional**

419 The CIM_IPProtocolEndpoint.ElementName property may support being modified by the ModifyInstance
420 operation. See 8.10.1.1.

421 This behavior is conditional. This clause describes the CIM elements and behavior requirements when an
422 implementation supports client modification of the CIM_IPProtocolEndpoint.ElementName property.

423 **7.1.4.1 CIM_EnabledLogicalElementCapabilities**

424 An instance of CIM_EnabledLogicalElementCapabilities shall be associated with the
425 CIM_IPProtocolEndpoint instance through an instance of CIM_ElementCapabilities.

426 7.1.4.1.1 CIM_EnabledLogicalElementCapabilities.ElementNameEditSupported

427 The ElementNameEditSupported property shall have a value of TRUE.

428 7.1.4.1.2 CIM_EnabledLogicalElementCapabilities.MaxElementNameLen

429 The MaxElementNameLen property shall be implemented.

430 7.1.5 Modifying ElementName Is Not Supported

431 This clause describes the CIM elements and behaviors that shall be implemented when the
432 CIM_IPProtocolEndpoint.ElementName property does not support being modified by the ModifyInstance
433 operation.

434 7.1.5.1 CIM_EnabledLogicalElementCapabilities

435 An instance of CIM_EnabledLogicalElementCapabilities may be associated with the
436 CIM_IPProtocolEndpoint instance through an instance of CIM_ElementCapabilities.

437 7.1.5.1.1 CIM_EnabledLogicalElementCapabilities.ElementNameEditSupported

438 The ElementNameEditSupported property shall have a value of FALSE.

439 7.1.5.1.2 CIM_EnabledLogicalElementCapabilities.MaxElementNameLen

440 The MaxElementNameLen property may be implemented. The MaxElementNameLen property is
441 irrelevant in this context.

442 7.1.6 Default Gateway

443 An IP interface can be configured with the address of a network gateway. Modeling of the default gateway
444 is optional. When the IP interface is configured with the address of a default gateway, an instance of
445 CIM_RemoteServiceAccessPoint shall represent the default gateway. The instance of
446 CIM_RemoteServiceAccessPoint shall be associated with the instance of CIM_IPProtocolEndpoint
447 through an instance of CIM_RemoteAccessAvailableToElement. An instance of
448 CIM_RemoteServiceAccessPoint may represent the default gateway even when a valid default gateway
449 has not been configured for the IP interface. It can be more convenient for an implementation to always
450 instantiate the instance of CIM_RemoteServiceAccessPoint even if a default gateway has not been
451 assigned to the IP interface rather than conditionally provide the relevant instances. For IPv4, this will
452 result in a single instance of CIM_RemoteServiceAccessPoint associated with the instance of
453 CIM_IPProtocolEndpoint.

454 For IPv6 or IPv4/IPv6 there may be one or more instances of CIM_RemoteServiceAccessPoint
455 associated with the instance of CIM_IPProtocolEndpoint, because there may be more than one default
456 gateway. In this case, the use of CIM_RemoteAccessAvailableToElement.OrderOfAccess can be used to
457 represent the list of default gateways in priority order.

458 7.1.6.1 CIM_RemoteServiceAccessPoint.AccessInfo

459 If the associated value of CIM_IPProtocolEndpoint.ProtocolIFType = 4096 (IPv4), then the value of the
460 AccessInfo property shall be the IPv4 address of the default gateway. The value shall be specified in
461 dotted decimal notation as defined in IETF [RFC 1208](#). A value of 0.0.0.0 shall indicate that a default
462 gateway has not been assigned to the associated IP interface.

463 If the associated value of CIM_IPProtocolEndpoint.ProtocolIFType = 4097 (IPv6), then the value of the
464 AccessInfo property shall be the IPv6 address of the default gateway. The value shall be specified in the
465 IPv6 notation as defined in IETF [RFC 4291](#). An unspecified address, which has the value of "::/128", shall
466 indicate that a default gateway has not been assigned to the associated IP interface.

467 If the associated value of CIM_IPProtocolEndpoint.ProtocolIFType = 4098 (IPv4/IPv6), then the value of
468 the AccessInfo property shall be the IPv6 address of the default gateway. The value shall be specified in
469 the IPv6 notation as defined in IETF [RFC 4291](#). An Unspecified Address, which has the value of “::/128”,
470 shall indicate that a default gateway has not been assigned to the associated IP interface.

471 **7.1.6.2 CIM_RemoteAccessAvailableToElement.Antecedent**

472 The value of the Antecedent reference shall be the instance of CIM_RemoteServiceAccessPoint.
473 Cardinality *.

474 **7.1.6.3 CIM_RemoteAccessAvailableToElement.Dependent**

475 The value of the Dependent reference shall be the instance of CIM_IPProtocolEndpoint. Cardinality *.

476 **7.1.6.4 CIM_RemoteAccessAvailableToElement.OrderOfAccess**

477 If the associated value of CIM_IPProtocolEndpoint.ProtocolIFType = 4096 (IPv4), then the
478 OrderOfAccess property shall have a value of 0 (Zero).

479 **7.2 DHCP Client Is Supported**

480 When a DHCP client is supported for the IP interface, the [DHCP Client Profile](#) shall be supported. This
481 behavior is optional.

482 **7.3 DNS Client Is Supported**

483 When a DNS client is supported for the IP interface, the [DNS Client Profile](#) shall be supported. This
484 behavior is optional.

485 **7.4 Managing Alternate Configurations — Optional**

486 Implementations may support the management of alternate or pending configurations for an IP interface.
487 When an implementation supports the management of alternate configurations, the following behavior
488 shall be supported.

489 **7.4.1 Configuration Management Is Supported**

490 The CIM_IPConfigurationService class provides management of alternate configurations and support for
491 configuring additional interfaces. When an implementation supports management of alternate
492 configurations, exactly one instance of CIM_IPConfigurationService shall be associated with the Central
493 Instance of the profile through an instance of CIM_ServiceAffectsElement. The existence of the
494 CIM_ServiceAffectsElement association is conditional on the existence of the
495 CIM_IPConfigurationService instance.

496 The CIM_IPConfigurationService instance shall be associated with a CIM_ComputerSystem instance
497 through an instance of CIM_HostedService. The existence of the CIM_HostedService association is
498 conditional on the existence of the CIM_IPConfigurationService instance.

499 **7.4.2 Representing an Alternate Configuration Using CIM_IPAssignmentSettingData**

500 Each instance of CIM_IPAssignmentSettingData shall represent a possible configuration for an IP
501 interface. The detailed settings for the IP interface shall be contained in the instances of subclasses of
502 CIM_IPAssignmentSettingData, which are associated with the instance of CIM_IPAssignmentSettingData
503 through instances of CIM_OrderedComponent.

504 The existence of one or more instances of CIM_IPAssignmentSettingData is conditional on the existence
505 of the CIM_IPConfigurationService instance. The existence of one or more instances of

506 CIM_ElementSettingData is conditional on the existence of one or more instances of
507 CIM_IPAssignmentSettingData.

508 **7.4.2.1 Associating an Alternate Configuration with an IP Interface**

509 The instance of CIM_IPAssignmentSettingData shall be associated with the instance of
510 CIM_IPProtocolEndpoint through an instance of CIM_ElementSettingData.

511 **7.4.2.1.1 CIM_ElementSettingData.IsCurrent**

512 When an instance of CIM_ElementSettingData associates an instance of CIM_IPAssignmentSettingData
513 with an instance of CIM_IPProtocolEndpoint, the CIM_ElementSettingData.IsCurrent property shall have
514 a value of 1 (Is Current) when the configuration represented by the referenced instance of
515 CIM_IPAssignmentSettingData is the last configuration applied to the IP interface represented by the
516 referenced instance of CIM_IPProtocolEndpoint.

517 When an instance of CIM_ElementSettingData associates an instance of CIM_IPAssignmentSettingData
518 with an instance of CIM_IPProtocolEndpoint, the CIM_ElementSettingData.IsCurrent property shall have
519 a value of 2 (Is Not Current) when the configuration represented by the referenced instance of
520 CIM_IPAssignmentSettingData is not the last configuration applied to the IP interface represented by the
521 referenced instance of CIM_IPProtocolEndpoint.

522 **7.4.3 Associating Settings Using CIM_OrderedComponent**

523 Instances of the subclasses of CIM_IPAssignmentSettingData contain the details of the IP configuration.
524 The CIM_OrderedComponent association aggregates these instances into instances of
525 CIM_IPAssignmentSettingData. An instance of CIM_IPAssignmentSettingData will have one or more
526 instances of its subclasses associated with it through an instance of CIM_OrderedComponent. An
527 instance of a subclass of CIM_IPAssignmentSettingData will be associated with one or more instances of
528 CIM_IPAssignmentSettingData.

529 **7.4.3.1 CIM_OrderedComponent.GroupComponent**

530 An instance of CIM_IPAssignmentSettingData shall be the value of the GroupComponent property of an
531 instance of CIM_OrderedComponent. Cardinality 1..*

532 **7.4.3.2 CIM_OrderedComponent.PartComponent**

533 An instance of a subclass of CIM_IPAssignmentSettingData shall be the value of the PartComponent
534 property of an instance of CIM_OrderedComponent. Cardinality *

535 **7.4.3.3 Interpretation of CIM_OrderedComponent.AssignedSequence**

536 The relative value of the CIM_OrderedComponent.AssignedSequence property shall indicate the order in
537 which aggregated instances of subclasses of CIM_IPAssignmentSettingData are applied to their
538 associated CIM_ProtocolEndpoint instances.

539 **7.4.3.3.1 Use of 0 (Zero)**

540 When the CIM_OrderedComponent.AssignedSequence property has a value of 0 (zero), the instance of
541 CIM_SettingData referenced by the CIM_OrderedComponent.PartComponent property shall not be
542 applied when the configuration represented by the CIM_IPAssignmentSettingData instance that is the
543 value of the CIM_OrderedComponent.GroupComponent property is applied. The
544 CIM_OrderedComponent.AssignedSequence property may have the value 0 (zero) when the instance of
545 CIM_OrderedComponent references an instance of CIM_DNSSettingData or
546 CIM_DNSGeneralSettingData. The CIM_OrderedComponent.AssignedSequence property shall not have
547 the value 0 (zero) when the instance of CIM_OrderedComponent does not reference an instance of
548 CIM_DNSSettingData or CIM_DNSGeneralSettingData.

549 7.4.3.3.2 Discreteness

550 Two instances of CIM_OrderedComponent that reference the same instance of
551 CIM_IPAssignmentSettingData shall not have the same value for their AssignedSequence properties
552 unless the value is 0 (zero).

553 7.4.4 Alternate Static Configuration

554 When an implementation supports the manual assignment of an IP configuration to the IP endpoint, an
555 instance of CIM_StaticIPAssignmentSettingData shall be associated with the CIM_IPProtocolEndpoint
556 through an instance of CIM_ElementSettingData. This instance of CIM_StaticIPAssignmentSettingData
557 shall be associated with at least one instance of CIM_IPAssignmentSettingData through an instance of
558 CIM_OrderedComponent. When the aggregating IP configuration has been applied to the IP interface
559 and the IP interface is using the settings contained in the instance of
560 CIM_StaticIPAssignmentSettingData, the IsCurrent property of the CIM_ElementSettingData instance
561 has the value 1 (Is Current). Otherwise, the CIM_ElementSettingData.IsCurrent property shall have the
562 value 2 (Is Not Current).

563 7.4.5 Alternate DHCP Configuration

564 When an alternate configuration includes the configuration of the DHCP client, the implementation will
565 follow the rules for representing a pending DHCP configuration defined in the [DHCP Client Profile](#).

566 7.4.6 DNS Client Alternate Configuration

567 When an alternate configuration includes the configuration of the DNS client, the implementation will
568 follow the rules for representing a pending DNS configuration defined in the [DNS Client Profile](#).

569 7.4.7 Relationship between DHCP and DNS Configuration

570 Some settings of the DNS configuration might be provided by the DHCP server.

571 An instance of CIM_IPAssignmentSettingData can have associated with it an instance of
572 CIM_DHCPSettingData and an instance of CIM_DNSSettingData. It is necessary to be able to
573 differentiate between a configuration in which the manual DNS settings take precedence and one in
574 which the DHCP assigned values take precedence. The DNS configuration is assigned according to the
575 principle of last applied. That is, within a given configuration, the last value applied for a property takes
576 precedence.

577 7.4.7.1 Relationship between DHCP Options and the DNS Configuration

578 This clause details the requirements for the relationship between DHCP options and CIM elements that
579 model the DNS configuration. For the requirements expressed in this clause, the following definitions
580 apply:

581 DHCPPE – the instance of CIM_DHCPProtocolEndpoint that represents the DHCP client for an IP
582 interface

583 DNSPE – the instance of CIM_DNSProtocolEndpoint that represents the DNS client that is associated
584 through an instance of CIM_SAPSAPDependency with the same instance of CIM_IPProtocolEndpoint
585 with which the DHCPPE is associated through an instance of CIM_SAPSAPDependency

586 DNS Pending – the instance of CIM_DNSSettingData that is associated through an instance of
587 CIM_OrderedComponent with the instance of CIM_IPAssignmentSettingData that is being applied to the
588 CIM_IPProtocolEndpoint instance

589 DHCP Pending – the instance of CIM_DHCPSettingData that is associated through an instance of
590 CIM_OrderedComponent with the instance of CIM_IPAssignmentSettingData that is being applied to the
591 CIM_IPProtocolEndpoint instance

592 The following requirements shall be met when the [DHCP Client Profile](#) and the [DNS Client Profile](#) are
593 implemented:

- 594 • When the OptionsReceived property of the DHCPPE instance and the DHCPOptionsToUse
595 property of the DNSPE instance both contain the value 8 (Domain Name Server), the DNS
596 Servers instrumented in accordance with the "DNS Server Representation" section of the [DNS
597 Client Profile](#) shall identify the DNS server addresses specified by the DHCP server as the data
598 for the Domain Name Server DHCP option.
- 599 • When the OptionsReceived property of the DHCPPE instance and the DHCPOptionsToUse
600 property of the DNSPE instance both contain the value 14 (Host Name), the value of the
601 Hostname property of the DNSPE instance shall be the hostname specified by the DHCP server
602 as the data for the Host Name DHCP option.
- 603 • When the OptionsReceived property of the DHCPPE instance and the DHCPOptionsToUse
604 property of the DNSPE instance both contain the value 17 (Domain Name), the value of the
605 DomainName property of the DNSPE instance shall be the domain name specified by the DHCP
606 server as the data for the Domain Name DHCP option.

607 When the RequestedHostname property of the DNS Pending instance has a non-null value and the
608 RequestedOptions or RequiredOptions property of the DHCP Pending instance contains the value 14
609 (Host Name), the DHCP client shall use the value of the RequestedHostname property as the data for the
610 Host Name DHCP option.

611 **7.4.8 Representing a Pending Configuration**

612 When an implementation supports alternate configurations, exactly one instance of
613 CIM_IPAssignmentSettingData shall be associated with the Central Instance through an instance of
614 CIM_ElementSettingData whose IsNext property has the value 1 (Is Next) .

615 Exactly one instance of CIM_IPAssignmentSettingData may be associated with the Central Instance
616 through an instance of CIM_ElementSettingData whose IsNext property has the value 3 (Is Next For
617 Single Use).

618 If an instance of CIM_IPAssignmentSettingData is associated with the Central Instance through an
619 instance of CIM_ElementSettingData whose IsNext property has the value 3 (Is Next For Single Use), this
620 instance of CIM_IPAssignmentSettingData shall represent the pending configuration. If no instance of
621 CIM_IPAssignmentSettingData is associated with the Central Instance through an instance of
622 CIM_ElementSettingData whose IsNext property has the value 3 (Is Next For Single Use), the instance of
623 CIM_IPAssignmentSettingData that is associated with the Central Instance through an instance of
624 CIM_ElementSettingData whose IsNext property has the value 1 (Is Next) shall represent the pending
625 configuration.

626 **7.5 Applying an Alternate Configuration**

627 Two methods exist for applying an alternate configuration to an IP interface. The first method allows a
628 client to explicitly select an alternate configuration to apply to an IP interface. A client can use the
629 CIM_IPConfigurationService.ApplySettingToIPProtocolEndpoint() method described in 8.1.1.1 to apply a
630 specific alternate configuration to the IP interface. The second method implicitly applies the pending
631 configuration to the IP interface when the IP interface transitions through a state transition or into a state
632 such that it will accept the pending configuration.

633 **7.5.1 Applying the Pending Configuration upon Transition to Enabled**

634 When the value of the EnabledState property of the CIM_IPProtocolEndpoint instance has a value other
635 than 2 (Enabled) and the value of the EnabledState property transitions to 2 (Enabled), the
636 implementation shall apply the pending configuration.

637 **7.5.2 Determining the Target CIM_ProtocolEndpoint Instance**

638 An instance of CIM_IPAssignmentSettingData or its subclasses may be associated with more than one
639 instance of a subclass of CIM_ProtocolEndpoint through instances of CIM_ElementSettingData.
640 Instances of subclasses of CIM_IPAssignmentSettingData may be aggregated into one or more instances
641 of CIM_IPAssignmentSettingData where the aggregating CIM_IPAssignmentSettingData instances are
642 associated with different instances of CIM_IPProtocolEndpoint. This is allowed as a convenience for
643 instrumentation to reduce the number of instances required when multiple IP interfaces share a common
644 configuration.

645 The following rules unambiguously identify the instance of a subclass of CIM_ProtocolEndpoint that will
646 have an instance of a subclass of CIM_SettingData applied to it when a pending configuration is applied
647 to an instance of CIM_IPProtocolEndpoint. Note that the DNS and DHCP related classes are owned by
648 the [DNS Client Profile](#) and [DHCP Client Profile](#), respectively. The algorithm for determining their use is
649 provided here because it is part of the behavior of applying a configuration.

650 When a pending IP configuration is applied, each instance of CIM_StaticIPAssignmentSettingData that is
651 associated with the CIM_IPAssignmentSettingData instance through an instance of
652 CIM_OrderedComponent shall be applied to the CIM_IPProtocolEndpoint instance that is identified as
653 follows:

- 654 1) The CIM_IPProtocolEndpoint instance shall be associated with the
655 CIM_StaticIPAssignmentSettingData instance through an instance of CIM_ElementSettingData.
- 656 2) The CIM_IPProtocolEndpoint instance shall be the CIM_IPProtocolEndpoint instance to which
657 the aggregating CIM_IPAssignmentSettingData is being applied.

658 When a pending IP configuration is applied, each instance of CIM_DHCPSettingData that is associated
659 with the CIM_IPAssignmentSettingData instance through an instance of CIM_OrderedComponent shall
660 be applied to the CIM_DHCPProtocolEndpoint instance that is identified as follows:

- 661 1) The CIM_DHCPProtocolEndpoint instance shall be associated with the CIM_DHCPSettingData
662 instance through an instance of CIM_ElementSettingData.
- 663 2) The CIM_DHCPProtocolEndpoint instance shall be associated through an instance of
664 CIM_SAPSAPDependency with the CIM_IPProtocolEndpoint instance to which the aggregating
665 CIM_IPAssignmentSettingData is being applied.

666 When a pending IP configuration is applied, each instance of CIM_DNSSettingData that is associated
667 with the CIM_IPAssignmentSettingData instance through an instance of CIM_OrderedComponent shall
668 be applied to the CIM_DNSProtocolEndpoint instance that is identified as follows:

- 669 1) The CIM_DNSProtocolEndpoint instance shall be associated with the CIM_DNSSettingData
670 instance through an instance of CIM_ElementSettingData.
- 671 2) The CIM_DNSProtocolEndpoint instance shall be associated through an instance of
672 CIM_SAPSAPDependency with the CIM_IPProtocolEndpoint instance to which the aggregating
673 CIM_IPAssignmentSettingData is being applied.

674 **7.5.3 Applying Static IP Settings**

675 When an instance of CIM_StaticIPAssignmentSettingData is applied to the CIM_IPProtocolEndpoint
676 instance, the values of the properties of the CIM_IPProtocolEndpoint instance shall be the values of the
677 properties of the CIM_StaticIPAssignmentSettingData instance.

678 **7.5.3.1 CIM_StaticIPAssignmentSettingData.GatewayIPv4Address**

679 If the associated value of CIM_IPProtocolEndpoint.ProtocolIFType = 4096 (IPv4), then the value of the
680 AccessInfo property of the CIM_RemoteServiceAccessPoint that represents the default gateway shall be
681 the value of the CIM_StaticIPAssignmentSettingData.GatewayIPv4Address property.

682 If the associated value of CIM_IPProtocolEndpoint.ProtocolIFType = 4098 (IPv4/IPv6), then the value of
683 the AccessInfo property of the CIM_RemoteServiceAccessPoint that represents the default IPv4 gateway
684 shall be the value of the CIM_StaticIPAssignmentSettingData.GatewayIPv4Address property.

685 **7.5.3.2 CIM_StaticIPAssignmentSettingData.GatewayIPv6Address**

686 If the associated value of CIM_IPProtocolEndpoint.ProtocolIFType = 4097 (IPv6), then the value of the
687 AccessInfo property of the CIM_RemoteServiceAccessPoint that represents the default IPv6 gateway
688 shall be the value of the CIM_StaticIPAssignmentSettingData.GatewayIPv6Address property.

689 **7.5.3.3 Successful Application of Settings**

690 An instance of CIM_StaticIPAssignmentSettingData shall be considered successfully applied when the
691 properties of the associated instance of CIM_IPProtocolEndpoint to which the instance of
692 CIM_StaticIPAssignmentSettingData has been applied have the values of the relevant properties of the
693 CIM_StaticIPAssignmentSettingData instance.

694 **7.5.4 Applying DHCP Settings**

695 When a pending configuration includes the configuration of the DHCP client, the DHCP configuration is
696 applied as defined in the [DHCP Client Profile](#).

697 **7.5.5 Applying DNS Settings**

698 When a pending configuration includes DNS client configuration, the DNS configuration is applied as
699 defined in the [DNS Client Profile](#). When the AssignedSequence property of the CIM_OrderedComponent
700 association that references an instance of CIM_DNSSettingData or CIM_DNSGeneralSettingData has a
701 non-zero value, the referenced instance of CIM_DNSSettingData or CIM_DNSGeneralSettingData shall
702 be applied, regardless of whether the application of a preceding CIM_SettingData instance was
703 successful.

704 **7.5.6 Resolving Overlapped Settings**

705 When more than one instance of CIM_StaticIPAssignmentSettingData or CIM_DHCPSettingData is
706 associated with the same instance of CIM_IPAssignmentSettingData, each CIM_SettingData instance
707 shall be applied in order (as described in 7.4.3.3) until the implementation determines that the resultant
708 configuration is valid. The amount of time an implementation waits after applying an instance of
709 CIM_SettingData before deciding whether the resultant configuration is valid is implementation specific
710 and outside the scope of this specification. The criterion for determining whether a configuration that is
711 represented by a specific CIM_SettingData instance is valid is implementation specific and outside the
712 scope of this specification.

713 **7.6 Relationship with a Network Interface**

714 An IP interface is generally bound to an underlying network interface. The underlying network interface
715 might participate in a LAN and be modeled using the [Host LAN Network Port Profile](#) or a specialization
716 thereof. When the underlying network interface is modeled with instrumentation compliant with the [Host
717 LAN Network Port Profile](#), an instance of CIM_BindsToLANEndpoint shall associate the Central Instance
718 of this profile with an instance of CIM_LANEndpoint that is compliant with the [Host LAN Network Port
719 Profile](#).

720 **8 Methods**

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

723 **8.1 CIM_IPProtocolEndpoint.RequestStateChange()**

724 Invocation of the RequestStateChange() method changes the element’s state to the value specified in the
 725 RequestedState parameter. The 2 (Enabled) and 3 (Disabled) values of the RequestedState parameter
 726 shall correspond to enabling or disabling the IP network interface, respectively. A value of 11 (Reset)
 727 shall correspond to disabling and then enabling the IP interface.

728 Detailed requirements of the RequestStateChange() method are specified in Table 2 and Table 3.

729 No standard messages are defined.

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

732 **Table 2 – CIM_IPProtocolEndpoint.RequestStateChange() Method: Return Code Values**

Value	Description
0	Request was successfully executed.
1	Method is unsupported.
2	Error occurred
4096	Job started: REF returned to started CIM_ConcreteJob

733 **Table 3 – CIM_IPProtocolEndpoint.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

734 **8.1.1.1 CIM_IPProtocolEndpoint.RequestStateChange() — Conditional Support**

735 When an instance of CIM_EnabledLogicalElementCapabilities is associated with the
 736 CIM_IPProtocolEndpoint instance and the
 737 CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported property contains at least one
 738 value, the CIM_IPProtocolEndpoint.RequestStateChange() method shall be implemented and supported.
 739 The CIM_IPProtocolEndpoint.RequestStateChange() method shall not return a value of 1 (Not
 740 Supported).

741 **8.2 CIM_IPConfigurationService.ApplySettingToIPProtocolEndpoint()**

742 The CIM_IPConfigurationService.ApplySettingToIPProtocolEndpoint() method is used to apply a
 743 configuration, as represented by an aggregating instance of CIM_IPAssignmentSettingData, to an IP
 744 interface, as represented by an instance of CIM_IPProtocolEndpoint. Implementation of this method is
 745 optional.

746 Detailed requirements of the ApplySettingToIPProtocolEndpoint() method are specified in Table 4 and
747 Table 5.

748 No standard messages are defined.

749 **Table 4 – CIM_IPConfigurationService.ApplySettingToIPProtocolEndpoint() Method: Return Code**
750 **Values**

Value	Description
0	Request was successfully executed.
1	Unsupported
2	Unknown/unspecified error
4	Failed
0x1000	Input parameters have been validated and a job started to apply the configuration.

751 **Table 5 – CIM_IPConfigurationService.ApplySettingToIPProtocolEndpoint() Method: Parameters**

Qualifiers	Name	Type	Description/Values
IN, REQ	Configuration	CIM_IPAssignmentSettingData REF	The settings to apply
IN, REQ	Endpoint	CIM_IPProtocolEndpoint REF	CIM_IPProtocolEndpoint to configure
OUT	Job	CIM_ConcreteJob REF	Returned if job started

752 The CIM_IPConfigurationService.ApplySettingToIPProtocolEndpoint() method shall be implemented as
753 follows:

- 754 • The implementation shall validate that an instance of CIM_ServiceAffectsElement references
755 the CIM_IPConfigurationService instance and the CIM_IPProtocolEndpoint instance that is
756 identified by the Endpoint parameter to the method. If the association does not exist, the return
757 code of the method shall be 4 (Failed).
- 758 • The implementation shall validate that an instance of CIM_ElementSettingData associates the
759 instance of CIM_IPProtocolEndpoint that is identified by the Endpoint parameter with the
760 instance of CIM_IPAssignmentSettingData that is identified by the Configuration parameter. If
761 the association does not exist, the return code of the method shall be 4 (Failed).

762 When the parameters have been validated and the method is applying the settings, the method shall
763 apply the settings as documented in 7.5 and its subclasses.

764 8.3 Profile Conventions for Operations

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

767 The default list of operations is as follows:

- 768 • GetInstance
- 769 • EnumerateInstances
- 770 • EnumerateInstanceNames
- 771 • Associators
- 772 • AssociatorNames

- 773 • References
- 774 • ReferenceNames

775 **8.4 CIM_BindsToLANEndpoint**

776 Table 6 lists implementation requirements for operations. If implemented, these operations shall be
 777 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 6, all operations in
 778 the default list in 8.3 shall be implemented as defined in [DSP0200](#).

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

780 **Table 6 – Operations: CIM_BindsToLANEndpoint**

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

781 **8.5 CIM_ElementSettingData**

782 Table 7 lists implementation requirements for operations. If implemented, these operations shall be
 783 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 7, all operations in
 784 the default list in 8.3 shall be implemented as defined in [DSP0200](#).

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

786 **Table 7 – Operations: CIM_ElementSettingData**

Operation	Requirement	Messages
ModifyInstance	Conditional. See 8.5.1.	None
Associators	Unspecified	None
AssociatorNames	Unspecified	None
References	Unspecified	None
ReferenceNames	Unspecified	None

787 **8.5.1 CIM_ElementSettingData — ModifyInstance**

788 The behavior of the ModifyInstance operation varies depending on the property of the association
 789 modified and the instances that are referenced by the association instance.

790 **8.5.1.1 CIM_ElementSettingData Referencing CIM_IPAssignmentSettingData**

791 When an instance of CIM_ElementSettingData associates an instance of CIM_IPAssignmentSettingData
 792 with an instance of CIM_IPProtocolEndpoint, the following rules shall govern the behavior of the
 793 ModifyInstance operation:

- 794 • The ModifyInstance operation shall not allow the IsDefault property to be modified.
- 795 • The ModifyInstance operation shall not allow the IsCurrent property to be modified.
- 796 • When the ModifyInstance operation is used to set the IsNext property to a value of 1 (Is Next),
 797 the ModifyInstance operation shall implement the following behavior:

- 798 1) The ModifyInstance operation shall find all other instances of CIM_ElementSettingData
 799 that associate an instance of CIM_IPAssignmentSettingData with the instance of
 800 CIM_IPProtocolEndpoint referenced by the target instance of CIM_ElementSettingData
 801 where the IsNext property has a value of 1 (Is Next).
- 802 2) For each instance of CIM_ElementSettingData found, the ModifyInstance operation shall
 803 modify the value of its IsNext property to have a value of 2 (Is Not Next).
- 804 • When the IsNext property has a value of 1 (Is Next), the ModifyInstance operation shall not be
 805 supported.
 - 806 • When the ModifyInstance operation is used to set the IsNext property to a value of 3 (Is Next for
 807 Single Use), the ModifyInstance operation shall implement the following behavior:
 - 808 1) The ModifyInstance operation shall find all other instances of CIM_ElementSettingData
 809 that associate an instance of CIM_IPAssignmentSettingData with the instance of
 810 CIM_IPProtocolEndpoint referenced by the target instance of CIM_ElementSettingData
 811 where the IsNext property has a value of 3 (Is Next For Single Use).
 - 812 2) For each instance of CIM_ElementSettingData found, the ModifyInstance operation shall
 813 modify the value of its IsNext property to have a value of 2 (Is Not Next).

814 8.5.1.2 CIM_ElementSettingData Referencing CIM_StaticIPAssignmentSettingData

815 When an instance of CIM_ElementSettingData associates an instance of
 816 CIM_StaticIPAssignmentSettingData with an instance of CIM_IPProtocolEndpoint, the ModifyInstance
 817 operation shall not be supported.

818 8.6 CIM_HostedAccessPoint

819 Table 8 lists implementation requirements for operations. If implemented, these operations shall be
 820 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 8, all operations in
 821 the default list in 8.3 shall be implemented as defined in [DSP0200](#).

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

823 **Table 8 – Operations: CIM_HostedAccessPoint**

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

824 8.7 CIM_HostedService

825 Table 9 lists implementation requirements for operations. If implemented, these operations shall be
 826 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 9, all operations in
 827 the default list in 8.3 shall be implemented as defined in [DSP0200](#).

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

829

830

Table 9 – Operations: CIM_HostedService

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

831 **8.8 CIM_IPAssignmentSettingData**

832 All operations in the default list in 8.3 shall be implemented as defined in [DSP0200](#).

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

834 **8.9 CIM_IPConfigurationService**

835 All operations in the default list in 8.3 shall be implemented as defined in [DSP0200](#).

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

837 **8.10 CIM_IPProtocolEndpoint**

838 Table 10 lists implementation requirements for operations. If implemented, these operations shall be
 839 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 10, all operations
 840 in the default list in 8.3 shall be implemented as defined in [DSP0200](#).

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

842 **Table 10 – Operations: CIM_IPProtocolEndpoint**

Operation	Requirement	Messages
ModifyInstance	Conditional. See 8.10.1.	None

843 **8.10.1 CIM_IPProtocolEndpoint — ModifyInstance Operation**

844 This clause details the specific requirements for the ModifyInstance operation applied to an instance of
 845 CIM_IPProtocolEndpoint.

846 **8.10.1.1 CIM_IPProtocolEndpoint.ElementName Property**

847 When an instance of CIM_EnabledLogicalElementCapabilities is associated with the
 848 CIM_IPProtocolEndpoint instance and the
 849 CIM_EnabledLogicalElementCapabilities.ElementNameEditSupported property has a value of TRUE, the
 850 implementation shall allow the ModifyInstance operation to change the value of the ElementName
 851 property of the CIM_IPProtocolEndpoint instance. The ModifyInstance operation shall enforce the length
 852 restriction specified in the MaxElementNameLen property of the instance of
 853 CIM_EnabledLogicalElementCapabilities.

854 When no instance of CIM_EnabledLogicalElementCapabilities is associated with the
 855 CIM_IPProtocolEndpoint instance, or the ElementNameEditSupported property of the
 856 CIM_EnabledLogicalElementCapabilities instance has a value of FALSE, the implementation shall not
 857 allow the ModifyInstance operation to change the value of the ElementName property of the
 858 CIM_IPProtocolEndpoint instance.

859 8.11 CIM_OrderedComponent

860 Table 11 lists implementation requirements for operations. If implemented, these operations shall be
 861 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 11, all operations
 862 in the default list in 8.3 shall be implemented as defined in [DSP0200](#).

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

864 **Table 11 – Operations: CIM_OrderedComponent**

Operation	Requirement	Messages
ModifyInstance	Optional. See 8.11.1.	None
Associators	Unspecified	None
AssociatorNames	Unspecified	None
References	Unspecified	None
ReferenceNames	Unspecified	None

865 8.11.1 CIM_OrderedComponent — ModifyInstance

866 The ModifyInstance operation may be supported for CIM_OrderedComponent. When an instance of
 867 CIM_OrderedComponent references an instance of CIM_DNSSettingData or an instance of
 868 CIM_DNSGeneralSettingData, the AssignedSequence property may be modified. When an instance of
 869 CIM_OrderedComponent references an instance of CIM_StaticIPAssignmentSettingData or an instance
 870 of CIM_DHCPSettingData, the AssignedSequence property shall not be modified.

871 8.12 CIM_RemoteAccessAvailableToElement

872 Table 12 lists implementation requirements for operations. If implemented, these operations shall be
 873 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 12, all operations
 874 in the default list in 8.3 shall be implemented as defined in [DSP0200](#).

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

876 **Table 12 – Operations: CIM_RemoteAccessAvailableToElement**

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

877 8.13 CIM_RemoteServiceAccessPoint

878 All operations in the default list in 8.3 shall be implemented as defined in [DSP0200](#).

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

880 8.14 CIM_ServiceAffectsElement

881 Table 13 lists implementation requirements for operations. If implemented, these operations shall be
 882 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 13, all operations
 883 in the default list in 8.3 shall be implemented as defined in [DSP0200](#).

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

885 **Table 13 – Operations: CIM_ServiceAffectsElement**

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

886 8.15 CIM_StaticIPAssignmentSettingData

887 Table 14 lists implementation requirements for operations. If implemented, these operations shall be
 888 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 14, all operations
 889 in the default list in 8.3 shall be implemented as defined in [DSP0200](#).

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

891 **Table 14 – Operations: CIM_StaticIPAssignmentSettingData**

Operation	Requirement	Messages
ModifyInstance	Optional	None

892 8.16 CIM_SystemDevice

893 Table 15 lists implementation requirements for operations. If implemented, these operations shall be
 894 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 15, all operations
 895 in the default list in 8.3 shall be implemented as defined in [DSP0200](#).

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

897 **Table 15 – Operations: CIM_SystemDevice**

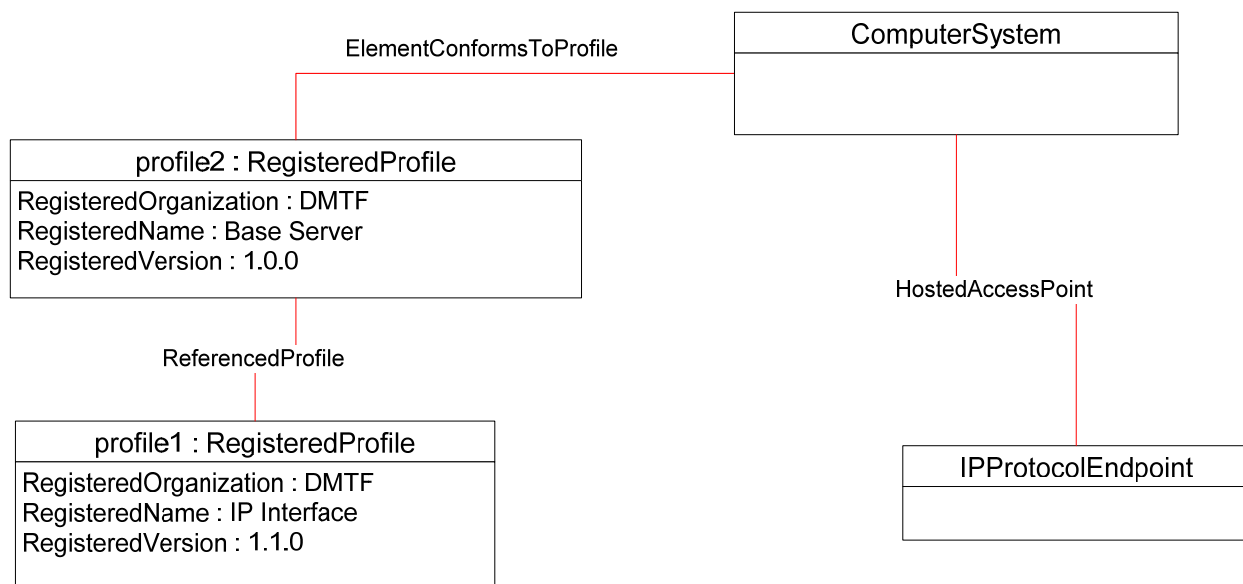
Operation	Requirement	Messages
Associators	Unspecified	None
AssociatorNames	Unspecified	None
References	Unspecified	None
ReferenceNames	Unspecified	None

9 Use Cases

This clause contains object diagrams and use cases for the *IP Interface Profile*.

9.1 Miscellaneous Object Diagrams

The object diagram in Figure 2 shows one possible method for advertising profile conformance. The instances of `CIM_RegisteredProfile` are used to identify the version of the *IP Interface Profile* with which an instance of `CIM_IPProtocolEndpoint` and its associated instances are conformant. An instance of `CIM_RegisteredProfile` exists for each profile that is instrumented in the system. One instance of `CIM_RegisteredProfile` identifies the "DMTF Base Server Profile version 1.0.0". The other instance identifies the "DMTF IP Interface Profile version 1.0.0". The `CIM_IPProtocolEndpoint` instance is scoped to an instance of `CIM_ComputerSystem`. This instance of `CIM_ComputerSystem` is conformant with the DMTF *Base Server Profile* version 1.0.0 as indicated by the `CIM_ElementConformsToProfile` association to the `CIM_RegisteredProfile` instance.



910

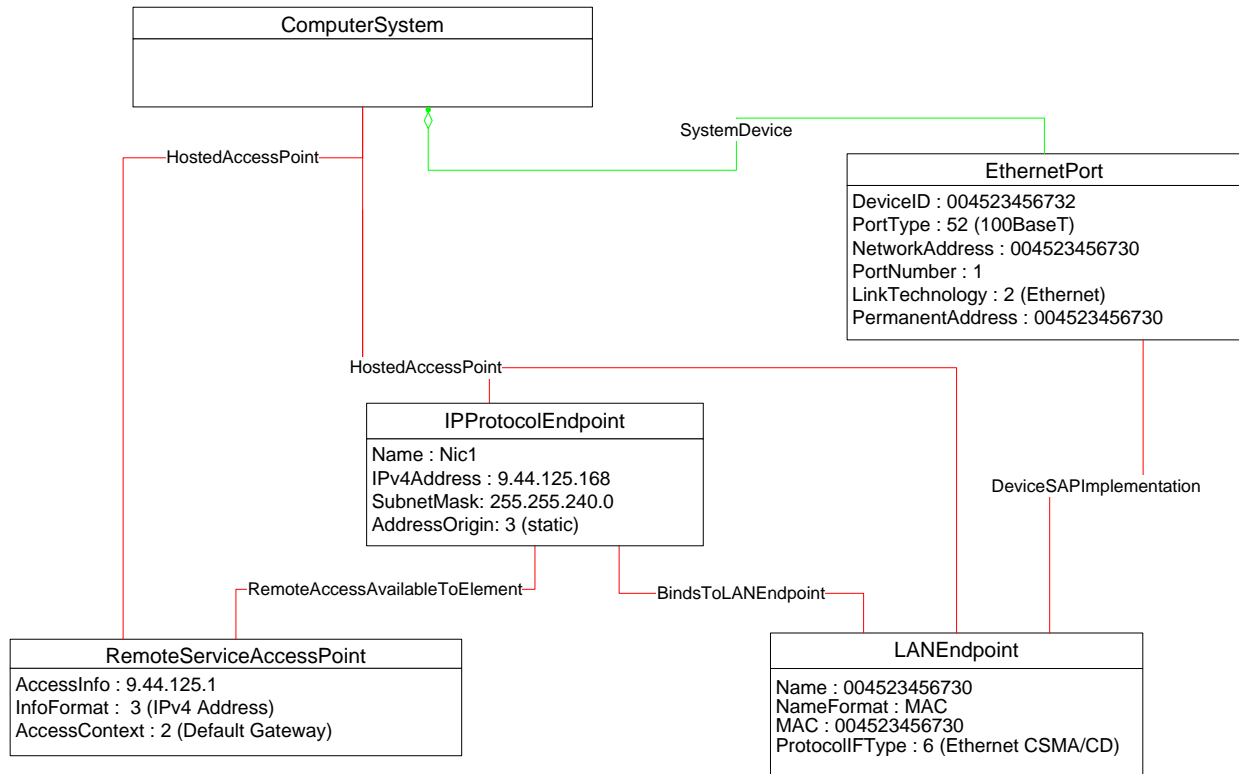
911

Figure 2 – Registered Profile

The object diagram shown in Figure 3 contains the basic elements used to model the current configuration of an IP interface when the `CIM_IPProtocolEndpoint.ProtocolIFType` is 4096 (Ipv4). The IP interface is bound to an Ethernet NIC, as illustrated by the `CIM_BindsToLANEndpoint` association between the `CIM_IPProtocolEndpoint` instance and the `CIM_LANEndpoint` instance. The `AddressOrigin` property of the `CIM_IPProtocolEndpoint` has a value of "static", indicating that the configuration was statically assigned. In this diagram, the *Ethernet Port Profile* and *IP Interface Profile* have been implemented.

The default gateway used by the IP interface is represented by the instance of `CIM_RemoteServiceAccessPoint` that is associated with the `CIM_IPProtocolEndpoint` instance through an instance of `CIM_RemoteAccessAvailableToElement`.

921



922

923

Figure 3 – Basic Configuration — IPv4

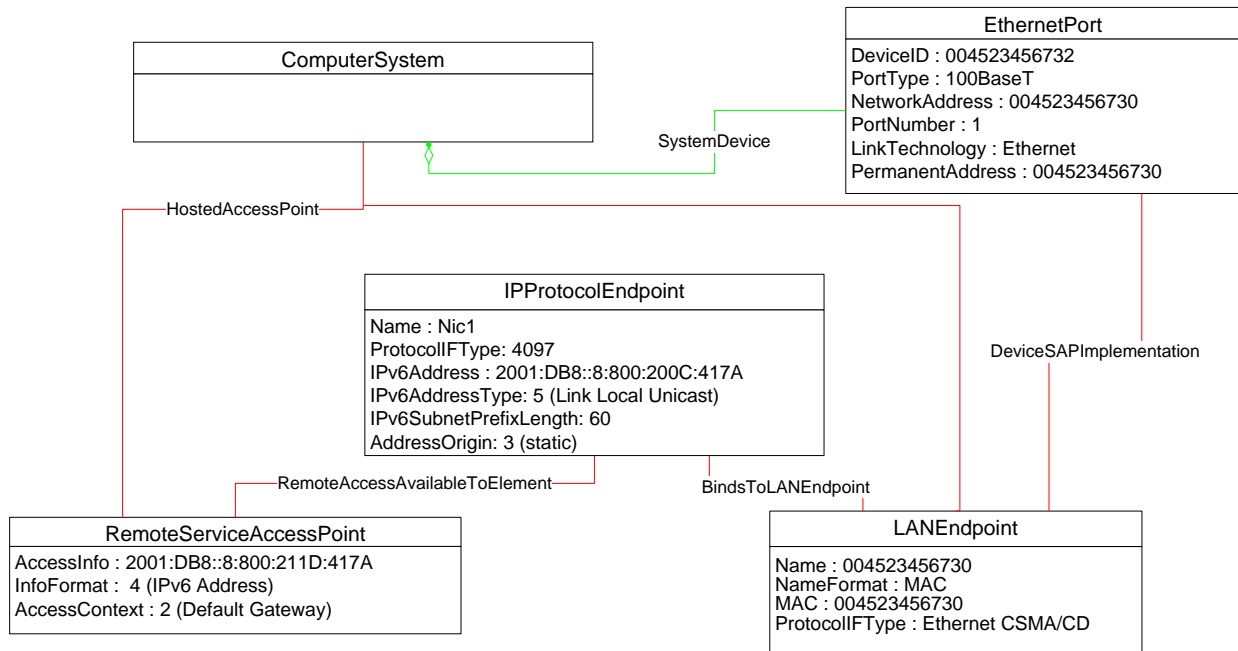
924

The object diagram shown in Figure 4 contains the basic elements used to model the current configuration of an IP interface when the CIM_IPProtocolEndpoint.ProtocolIFType is 4097 (IPv6). Note the similarities between this figure and the previous diagram. In this diagram, the *Ethernet Port Profile* and *IP Interface Profile* have been implemented.

926

927

928



929

930

Figure 4 – Basic Configuration — IPv6

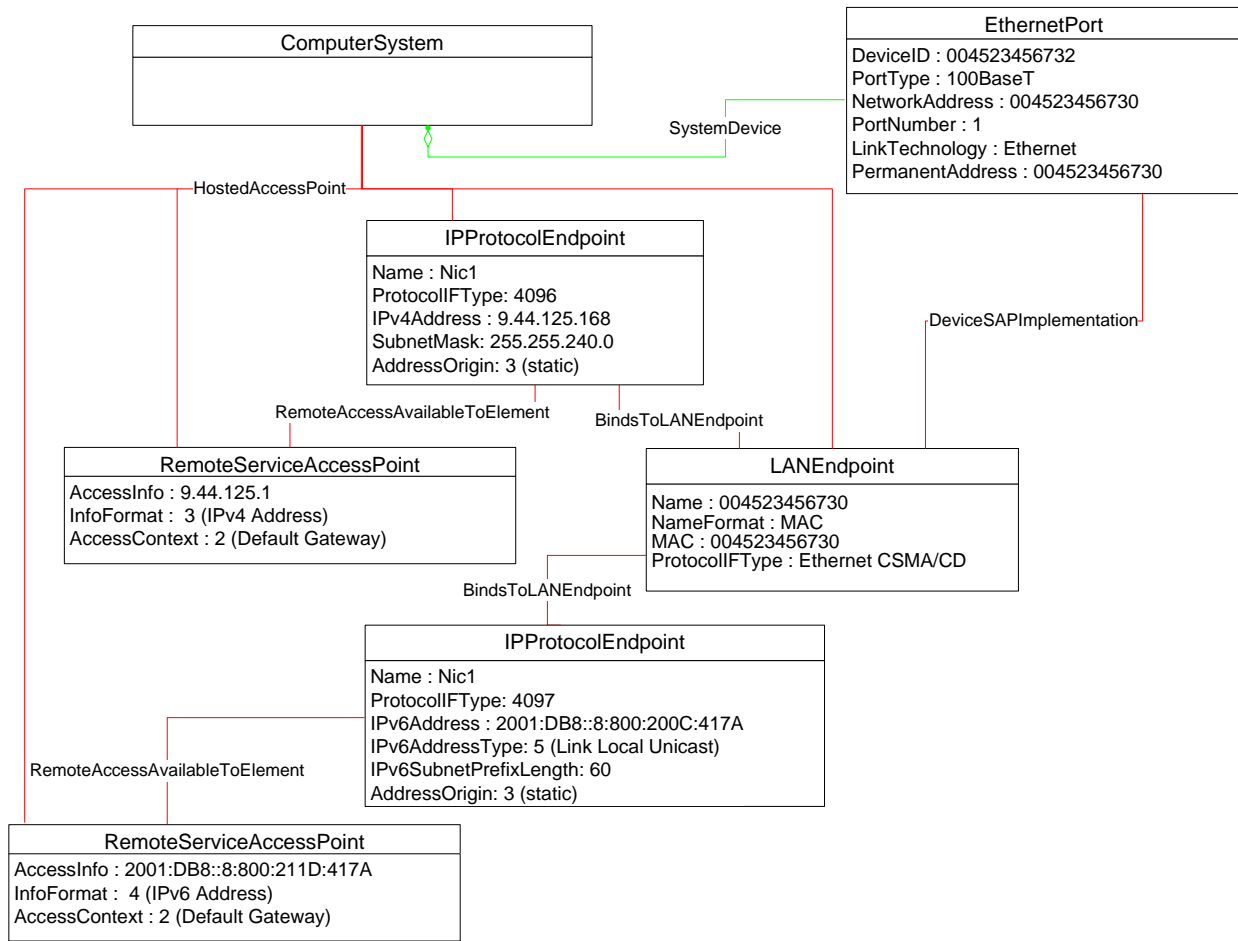
931

932

933

934

The object diagram shown in Figure 5 contains the basic elements used to model the current configuration of two IP interfaces on a single EthernetPort — one that has an IPv4 address and one that has an IPv6 address. In this diagram, the *Ethernet Port Profile* and *IP Interface Profile* have been implemented.



935

936

Figure 5 – Basic Configuration — IPv4 and IPv6

937

Figure 6 illustrates the elements and properties of an IP interface that supports static configuration. The IP interface currently has a single, alternate configuration associated with it. The optional IP configuration management behavior is depicted in this object diagram. Note that the pending configuration has been modified after it was applied to the CIM_IPProtocolEndpoint. Hence the values for properties of CIM_IPProtocolEndpoint do not align with the values of properties of the CIM_StaticIPAssignmentSettingData instance.

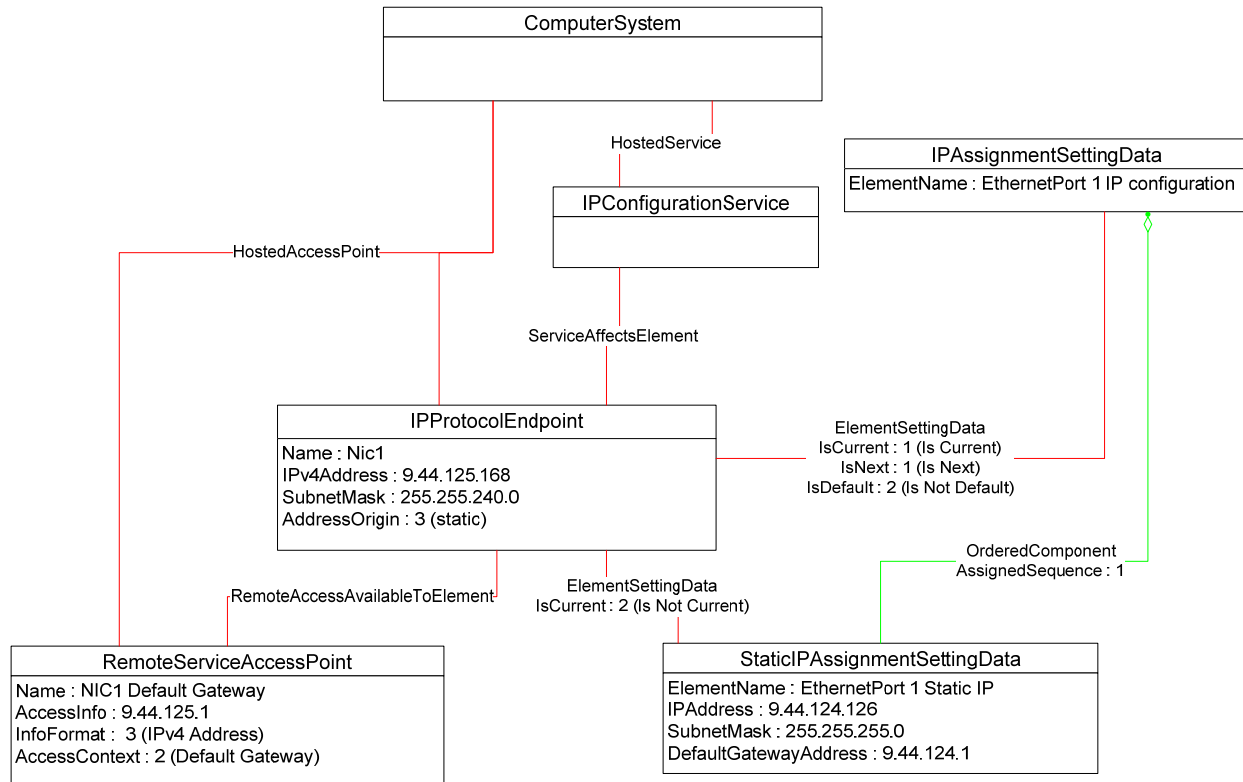
938

939

940

941

942



943

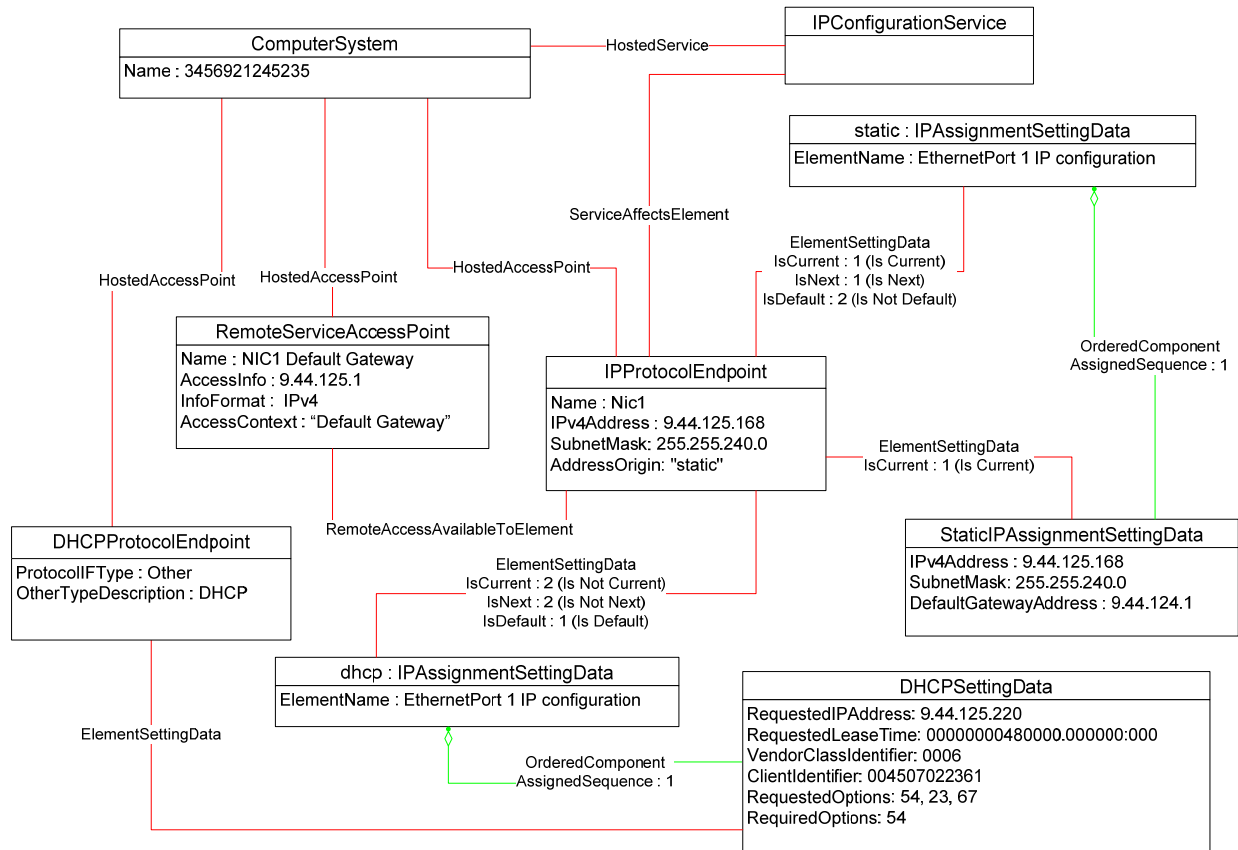
944

Figure 6 – Static Current and Pending Configuration

945 The object diagram in Figure 7 illustrates an IP interface with two supported alternate configurations. Two
 946 discrete IP configuration options are available for the IP interface. Each option is represented by an
 947 instance of CIM_IPAssignmentSettingData. One configuration option represents the ability to statically
 948 assign the IP configuration. This option is indicated by the instance of CIM_OrderedComponent that
 949 associates the CIM_IPAssignmentSettingData instance with an instance of
 950 CIM_StaticIPAssignmentSettingData. The other configuration option is to obtain the configuration through
 951 a DHCP client. This option is indicated by the instance of CIM_OrderedComponent that associates the
 952 CIM_IPAssignmentSettingData with an instance of CIM_DHCPSettingData.

953 In this example, each configuration option consists of a single instance of a subclass of
 954 CIM_IPAssignmentSettingData. Therefore, the value of the AssignedSequence property of the
 955 CIM_OrderedComponent instances is irrelevant.

956 The default configuration is to attempt to obtain a configuration through DHCP. This default is indicated
 957 by the IsDefault property having a value of 1 (Is Default) on the CIM_ElementSettingData instance that
 958 associates the CIM_IPAssignmentSettingData instance with the CIM_IPProtocolEndpoint instance.
 959 However, the current configuration of the IP interface was statically assigned using the configuration
 960 identified by the CIM_IPAssignmentSettingData instance *static*. This configuration is indicated by the
 961 value of the IsCurrent property on the instance of CIM_ElementSettingData that associates the
 962 CIM_IPAssignmentSettingData instance *static* with the CIM_IPProtocolEndpoint instance, and by the
 963 value of the AddressOrigin property on the CIM_IPProtocolEndpoint instance. When the interface is
 964 restarted, the static configuration will be used again for the IP interface. This behavior is indicated by the
 965 value of the IsNext property on the instance of CIM_ElementSettingData that associates the
 966 CIM_IPAssignmentSettingData instance *static* to the CIM_IPProtocolEndpoint instance.



967

968

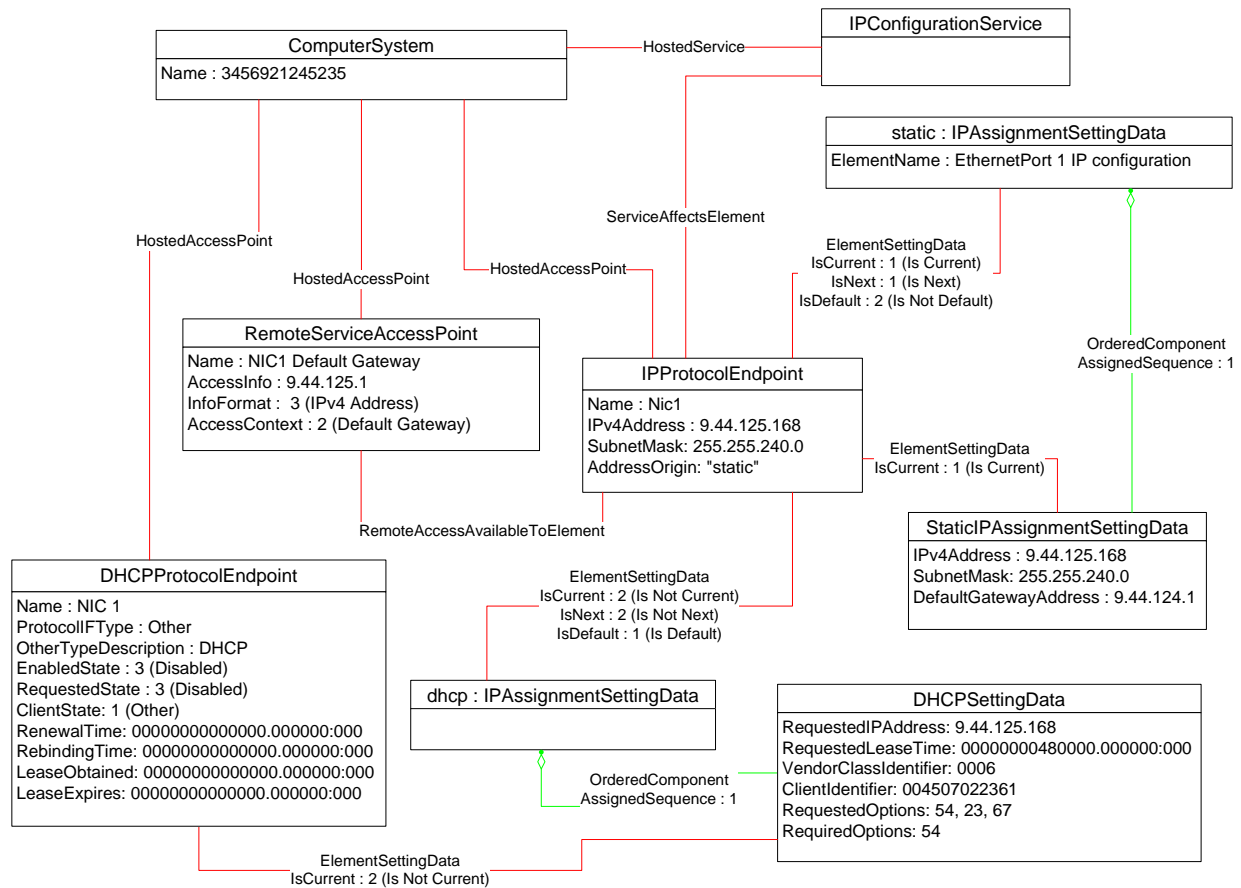
Figure 7 – Static and DHCP Pending Configurations

969 The object diagram in Figure 8 provides an example of an IP interface that was configured to default to a
 970 statically assigned IP configuration if the DHCP client failed to obtain a configuration from a DHCP server.

971 The EnabledState and ClientState properties of the CIM_DHCPProtocolEndpoint instance indicate that
 972 the DHCP client is not disabled but neither is it actively attempting to obtain a configuration any longer.
 973 No instance of CIM_RemoteServiceAccessPoint is associated with the CIM_DHCPProtocolEndpoint
 974 instance because the DHCP client failed to communicate with a DHCP server.

975 The AddressOrigin property of the CIM_IPProtocolEndpoint instance reflects that the address was
 976 assigned statically.

977

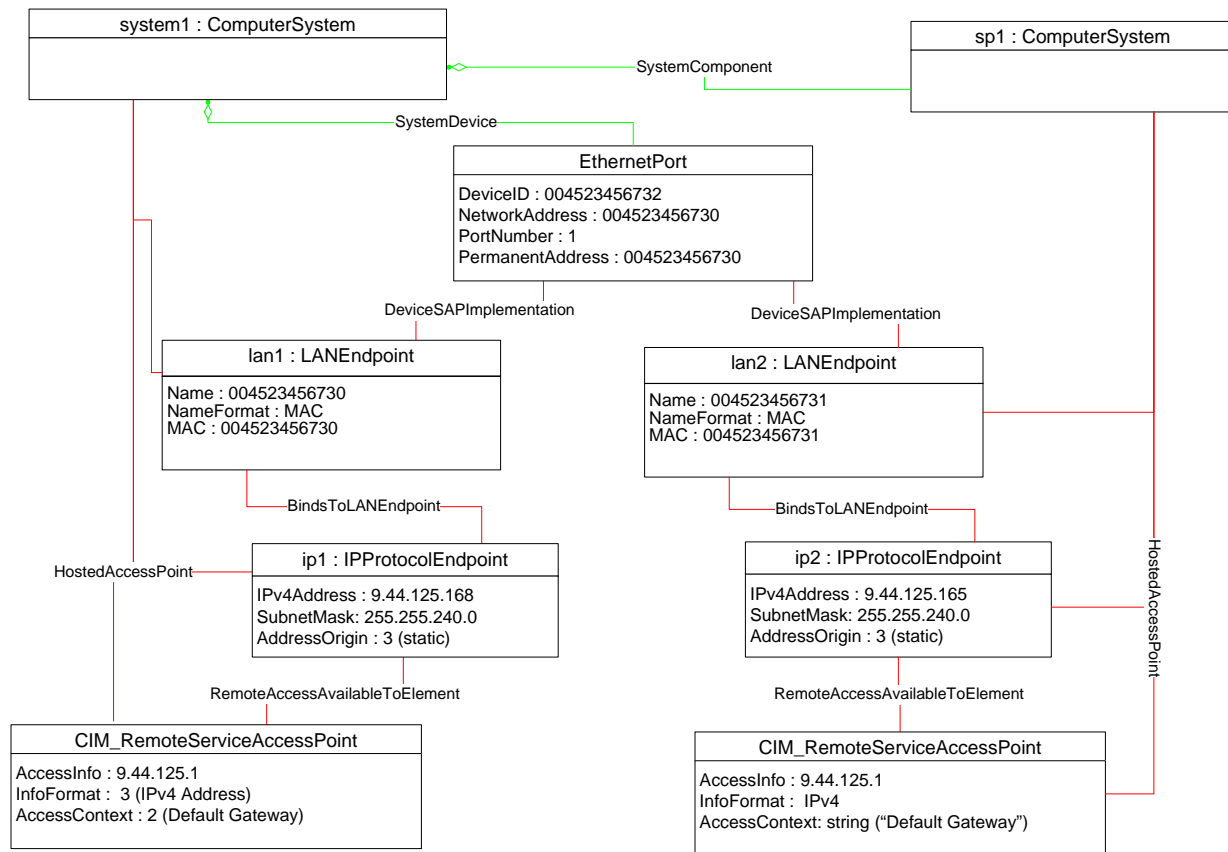


978

979

Figure 8 – DHCP Timed Out to a Static Configuration

980 The object diagram in Figure 9 illustrates a configuration in which a system contains an integrated service
 981 processor and they share the network interface of the system. The CIM_EthernetPort instance is
 982 associated with the system1 instance, which indicates that the network device is owned by the server.
 983 The MAC property of the lan1 instance matches the PermanentAddress property of the CIM_EthernetPort
 984 instance, which indicates that the server is using the hardware MAC. The MAC property of the lan2
 985 instance is different, which indicates that the service processor has been assigned a logical MAC. The
 986 system and service processor each have a unique IP interface that has been statically configured.



987

988

Figure 9 – Service Processor and Server Share an NIC

989 The object diagrams in Figure 10 through Figure 14 show different aspects of a single system. The
 990 system has support for the DNS and DHCP clients. For configurations using DHCP, the DNS
 991 configuration can be statically assigned or partially assigned through DHCP. The system itself does not
 992 support the persistence of alternate configurations. Rather the instrumentation layer presents the different
 993 configuration possibilities as distinct alternate configurations.

994 Note that in the following figures extraneous classes that are not relevant to the point being illustrated are
 995 not shown. For example, the CIM_HostedAccessPoint associations are never included.

996 The object diagram in Figure 10 outlines the alternate configurations presented by the instrumentation
 997 layer for the system. Three alternate configurations are shown: static_only, dhcp_only, and dhcp_static.

998 The system persists a single underlying static IP configuration, which is represented by static1. When the
 999 configuration selected is static only or DHCP and then static, the same client static IP configuration is
 1000 used.

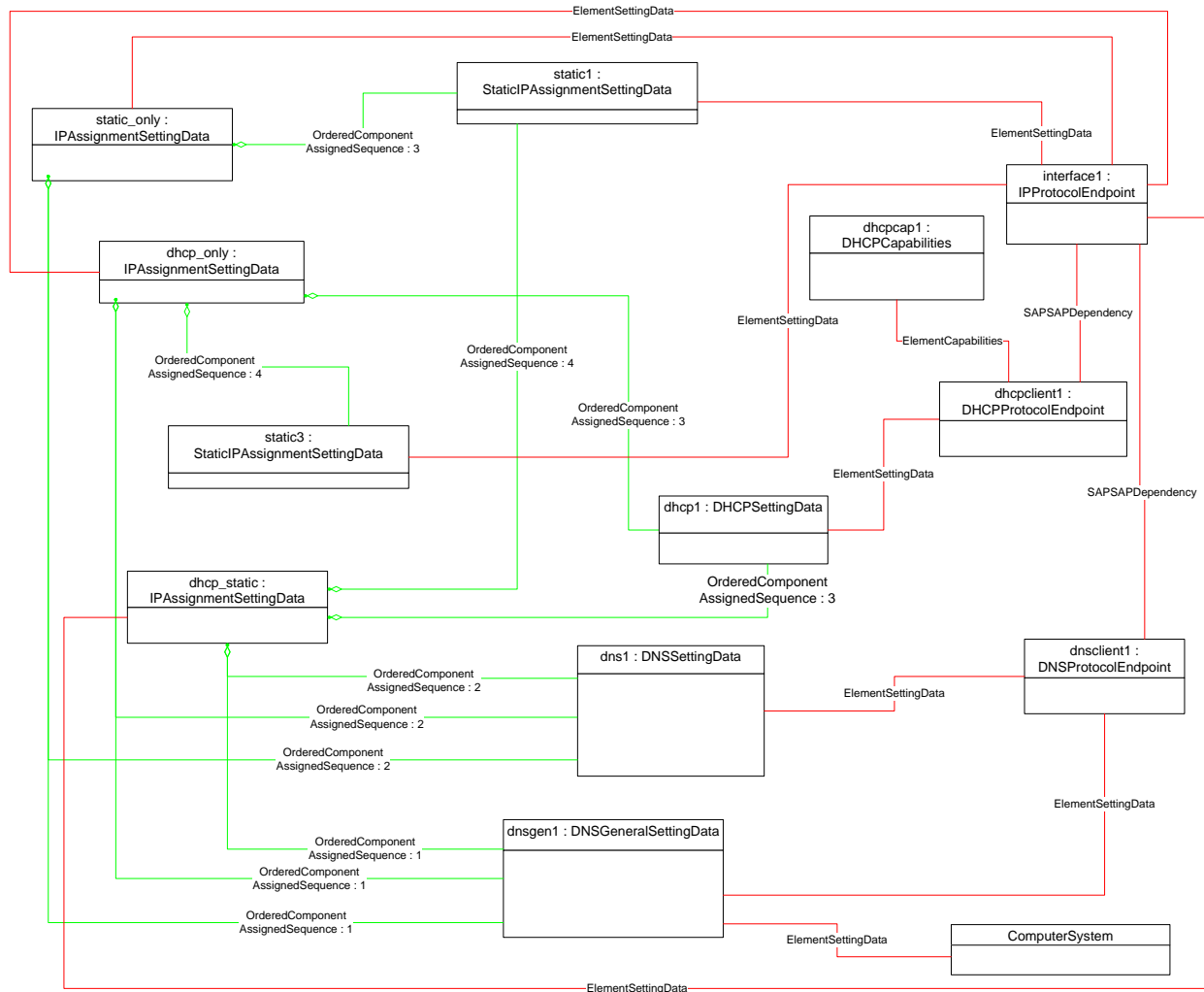
1001 The system persists a single underlying DNS configuration represented by dns1 and dnsgen1.

1002 static_only represents a configuration that uses static assignment of the IP configuration, including
 1003 support for static configuration of the DNS client. This behavior is indicated by the aggregated instances:
 1004 static1, dns1, and dnsgen1.

1005 dhcp_only represents a configuration that uses DHCP to obtain the IP configuration. This behavior is
 1006 indicated by the aggregated instance dhcp1. The DNS configuration can be assigned through DHCP or
 1007 statically assigned. This behavior is indicated by the aggregated instances dns1 and dnsgen1. In the
 1008 event the DHCP client is unable to obtain a configuration, the system is implemented to default to a hard-

1009 coded, well-known default static IP configuration. The existence of a default configuration is indicated by
 1010 the aggregated instance static3. Note that no advertisement mechanism is specified in the profile to
 1011 indicate that static3 represents hard-coded values that cannot be modified by the client. If the system
 1012 were implemented such that the DHCP client would be continually in use without a timeout to a static
 1013 configuration, the aggregated instance static3 would not exist.

1014 dhcp_static represents a configuration that attempts to use DHCP to obtain an IP configuration. In the
 1015 event the DHCP client fails to obtain a configuration, the system defaults to a client-assigned static IP
 1016 configuration. This behavior is indicated by the instances dhcp1 and static1 and the relative values of the
 1017 AssignedSequence property of the instances of CIM_OrderedComponent, which aggregate them into
 1018 dhcp_static.



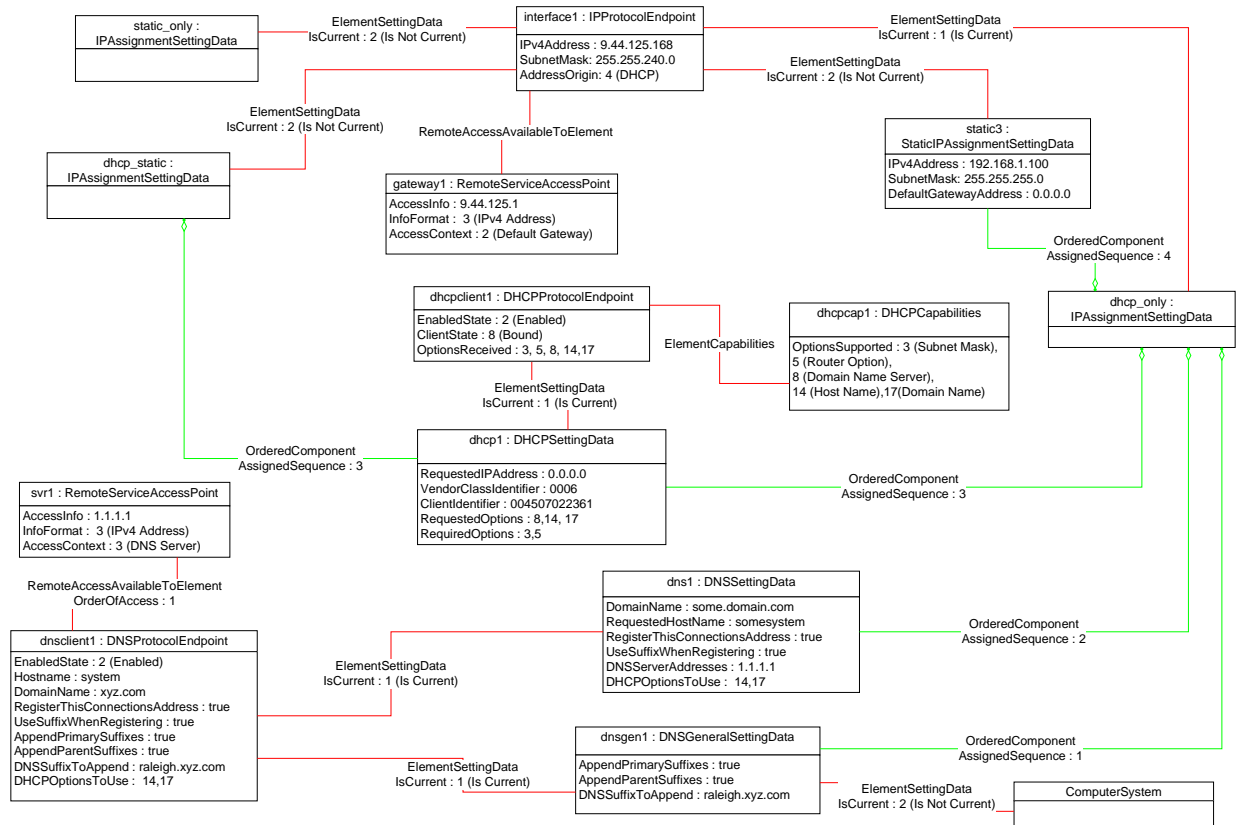
1019

1020

Figure 10 – Configuration Choices

1021 The object diagram in Figure 11 reflects the system when the DHCP configuration method has been used
 1022 and the DNS configuration has partially been assigned through DHCP and partially statically configured.

1023 The use of the DHCP-only configuration is indicated by the IsCurrent property of the instance of
 1024 CIM_ElementSettingData that associates dhcp_only to interface1 having the value 1 (Is Current). The
 1025 DHCP configuration includes DHCP options that affect the DNS configuration. The DHCP options 8, 14,
 1026 and 17 are requested as indicated by the RequestedOptions property of dhcp1. Each of these options
 1027 was in turn received by the DHCP client, which is indicated by the value of the OptionsReceived property
 1028 of dhcpclient1. The DNS client has been configured to use the values received for options 14 and 17 as
 1029 indicated by the presence of these values in the DHCPOptionsToUse property of dnsclient1. The
 1030 properties on dnsclient1 reflect the current DNS client configuration. Note that the actual current
 1031 configuration does not directly reflect the configuration indicated by dns1 and dnsngen1. The two
 1032 properties for which values were supplied by the DHCP options instead reflect the values assigned by the
 1033 DHCP server.



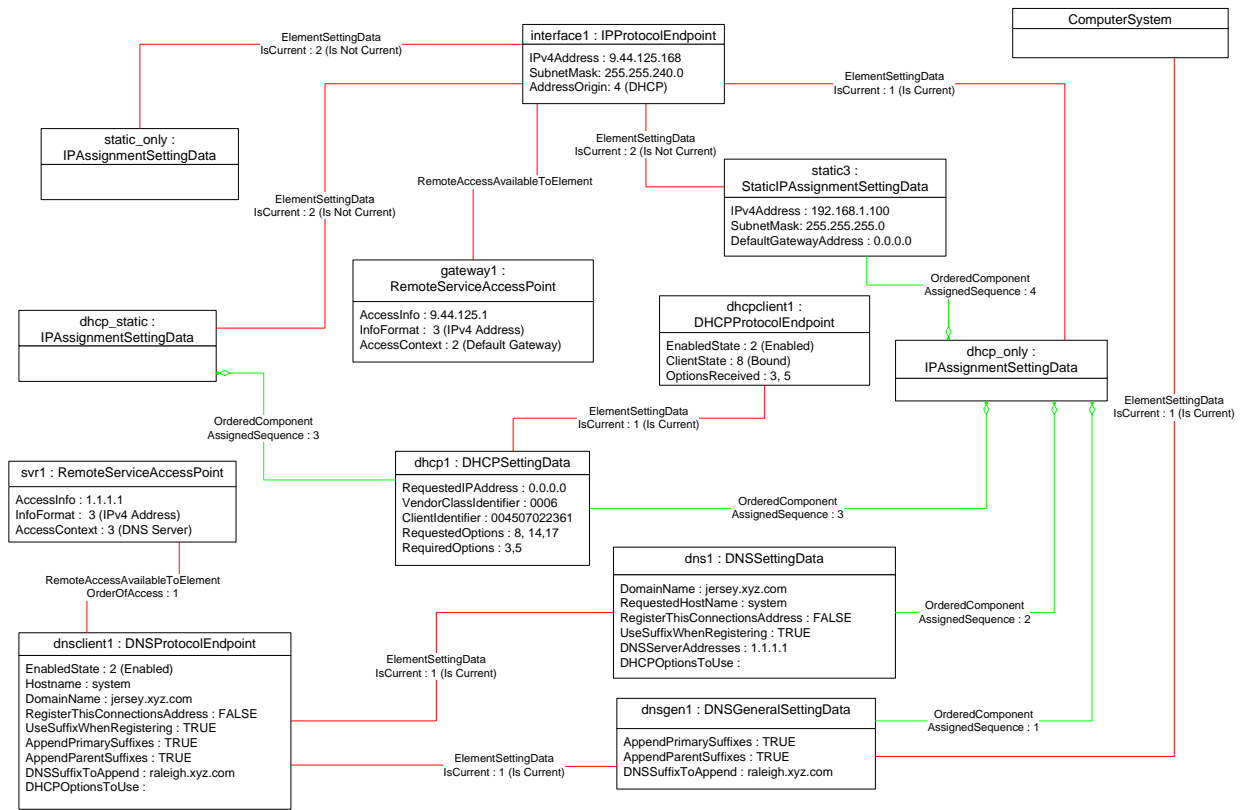
1034

1035

Figure 11 – DHCP Assigned Partial DNS

1036 The object diagram in Figure 12 reflects the system when the DHCP configuration method has been used
 1037 and the DNS configuration has been statically configured.

1038 The use of the DHCP-only configuration is indicated by the IsCurrent property of the instance of
 1039 CIM_ElementSettingData that associates dhcp_only to interface1 having the value 1 (Is Current).
 1040 Although the DHCP configuration includes DHCP options that affect the DNS configuration, the values
 1041 returned are not being used by the DNS client. This behavior is indicated by the absence of any values in
 1042 the DHCPOptionsToUse property of dnsclient1. The actual current configuration directly reflects the
 1043 configuration indicated by dns1 and dnsngen1 because no DHCP options are selected for use.



1044

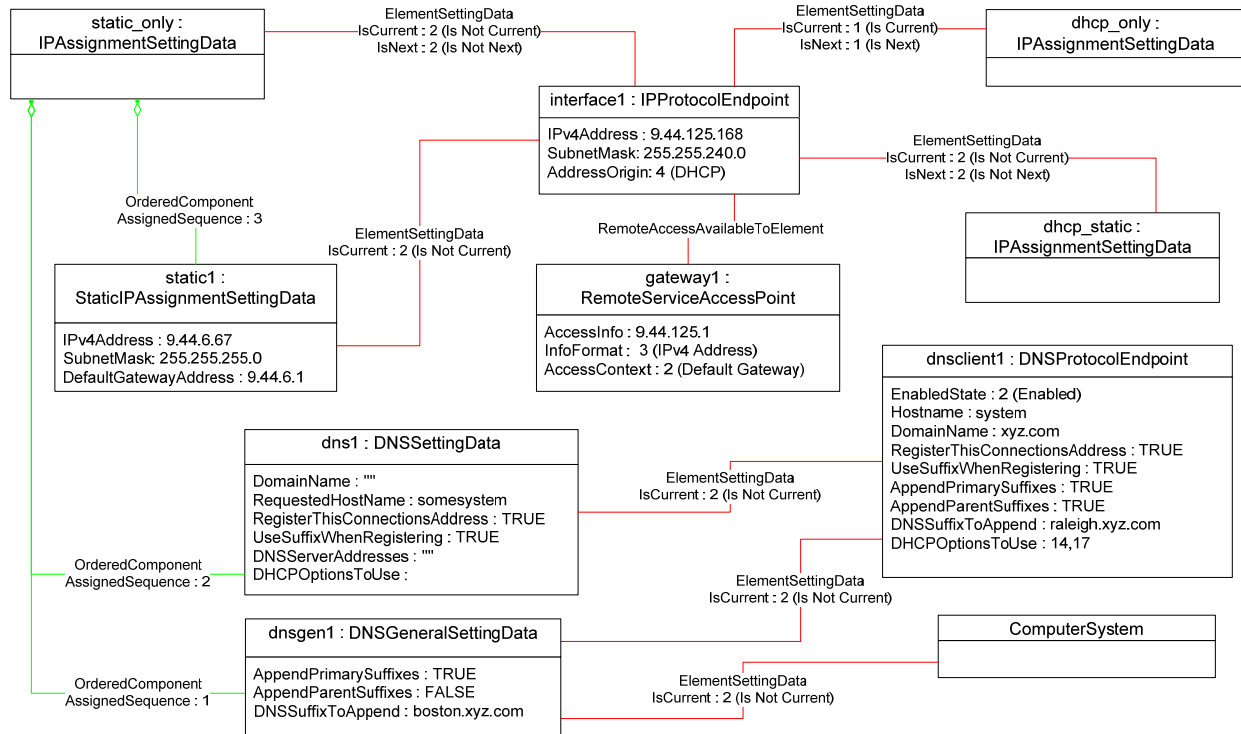
1045

Figure 12 – DHCP with DNS Statically Configured

1046 **9.1.1 Sequence for Disabled DNS Client**

1047 The following three object diagrams illustrate the system when a client is configuring it to use a static IP
 1048 configuration with the DNS client disabled. The client first modifies the pending static configuration so that
 1049 the DNS settings will not be applied. Then it disables the DNS client directly. Finally, it applies the static
 1050 configuration.

1051 The object diagram in Figure 13 illustrates the state of the system before the client begins modifying it to
 1052 use a static IP configuration with DNS disabled. The last configuration applied was the DHCP-only
 1053 configuration, which is indicated by the value of the IsCurrent property of the CIM_ElementSettingData
 1054 instance that references dhcp_only and interface1. The static_only configuration has not yet been
 1055 modified by the client. As shown, the alternate DNS configuration represented by dns1 and dnsngen1
 1056 would be applied if static_only were applied to interface1.

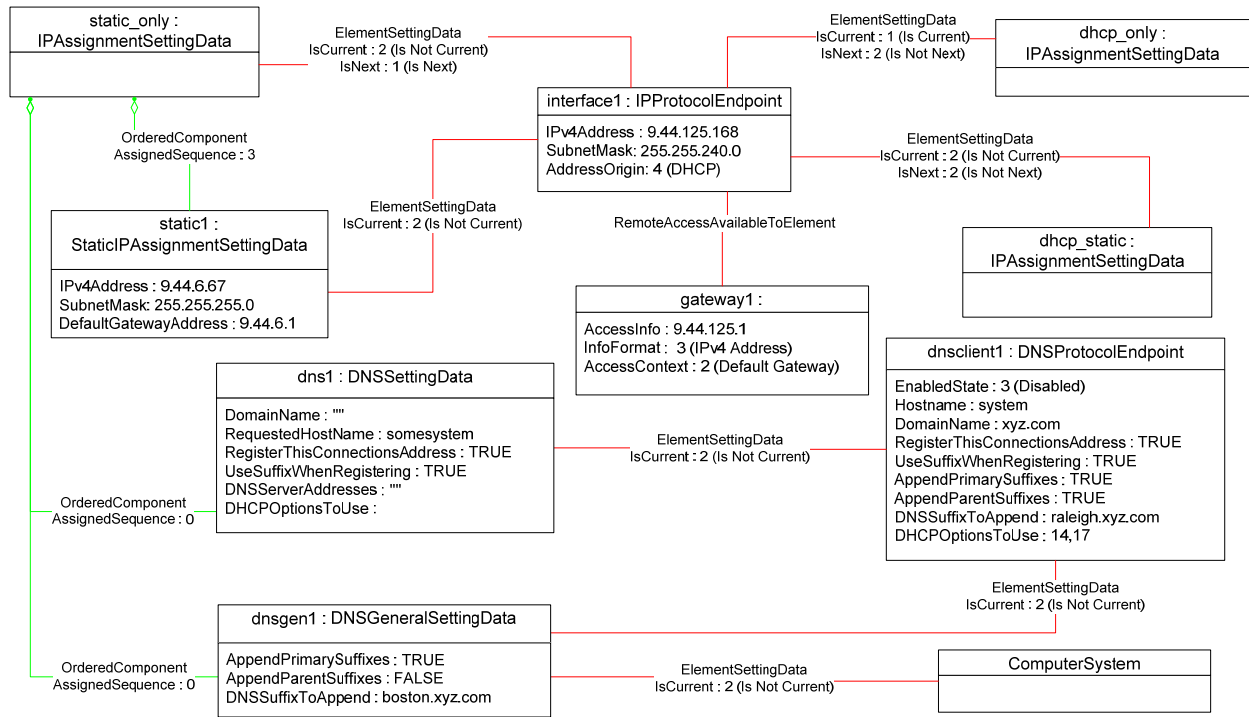


1057

1058

Figure 13 – Static without DNS Configuration — One

1059 In Figure 14, static_only has been made the pending configuration for interface1. This behavior is
 1060 indicated by the value of the IsNext property of the instance of CIM_ElementSettingData that references
 1061 static_only and interface1. static_only has been modified such that the DNS configuration will not be
 1062 applied. This behavior is indicated by the AssignedSequence property having a value of 0 (zero) for each
 1063 of the CIM_OrderedComponent instances that reference static_only and dns1 or dnsgen1. Separately,
 1064 the DNS client has been disabled, which is indicated by the value of the EnabledState property of
 1065 dnsclient1.

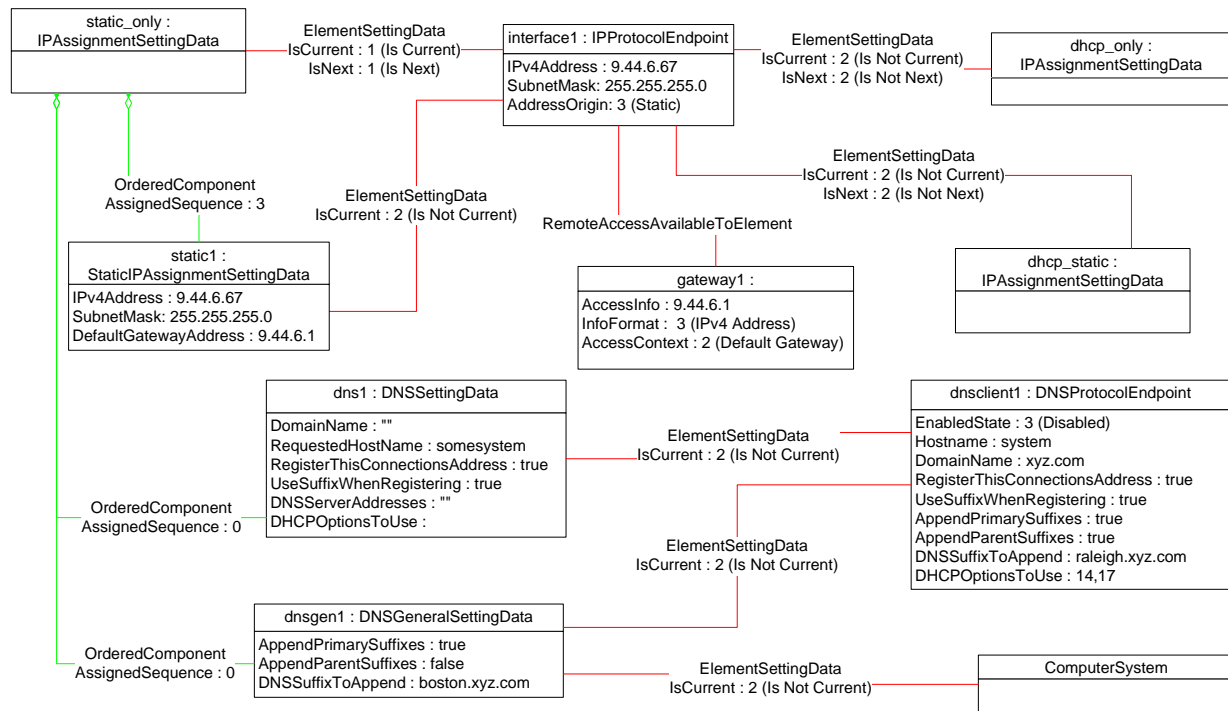


1066

1067

Figure 14 – Static without DNS Configuration — Two

1068 Figure 15 shows the system after static_only has been applied to interface1. Note that the current DNS
 1069 configuration has not changed as a result of applying static_only to interface1.



1070

1071

Figure 15 – Static without DNS Configuration — Three

1072 9.2 Determine Supported Configuration Methods

1073 A client can determine which configuration methods are supported for a given interface as follows:

- 1074 1) Find all instances of CIM_IPAssignmentSettingData that are associated with the
1075 CIM_IPProtocolEndpoint instance.
- 1076 2) For each instance of CIM_IPAssignmentSettingData:
 - 1077 • Find all instances of subclasses of CIM_IPAssignmentSettingData that are associated with the
1078 CIM_IPAssignmentSettingData instance through an instance of CIM_OrderedComponent.
 - 1079 • Query the value of the AddressOrigin property to determine the supported identified configuration
1080 method.

1081 9.3 Determine Gateway Address

1082 A client can find the default gateway in use for an IP interface as follows:

- 1083 1) Find all instances of CIM_RemoteServiceAccessPoint that are associated with the
1084 CIM_IPProtocolEndpoint instance through an instance of
1085 CIM_RemoteAccessAvailableToElement.
- 1086 2) For each instance of CIM_RemoteServiceAccessPoint, determine if the value of the
1087 AccessContext property is "Default Gateway". If so, query the value of the AccessInfo property.

1088 9.4 Determine Method Used for Current Configuration

1089 A client can determine the method by which the IP configuration was assigned by querying the
1090 AddressOrigin property of the CIM_IPProtocolEndpoint instance.

1091 9.5 Determine Whether DHCP Then Static Is Supported

1092 An implementation may support attempting to acquire its IP configuration through a DHCP client and
1093 defaulting to a static configuration if the client fails to acquire a configuration from a DHCP server. A client
1094 can determine whether this functionality is supported as follows:

- 1095 1) Find all instances of CIM_IPAssignmentSettingData (the parent class and not subclasses) that
1096 are associated with the CIM_IPProtocolEndpoint instance.
- 1097 2) For each instance of CIM_IPAssignmentSettingData:
 - 1098 a) Find all instances of CIM_DHCPSettingData that are associated through an instance of
1099 CIM_OrderedComponent.
 - 1100 b) Find all instances of CIM_StaticIPAssignmentSettingData that are associated through an
1101 instance of CIM_OrderedComponent.
- 1102 3) Determine if there is an instance of CIM_DHCPSettingData such that the value of the
1103 AssignedSequence property of the CIM_OrderedComponent that associates the instance of
1104 CIM_DHCPSettingData with the instance of CIM_IPAssignmentSettingData is less than the
1105 value of the AssignedSequence property of an instance of CIM_OrderedComponent that
1106 associates the CIM_StaticIPAssignmentSettingData with the instance of
1107 CIM_IPAssignmentSettingData. If so, DHCP then static is supported.

1108 9.6 View Default Configuration

1109 A client can view the default configuration for an IP interface as follows:

- 1110 1) Find all instances of CIM_ElementSettingData that associate an instance of
1111 CIM_IPAssignmentSettingData (the parent class and not subclasses) with the
1112 CIM_IPProtocolEndpoint instance.

- 1113 2) For each instance of CIM_ElementSettingData, see if the value of the IsDefault property is 1 (Is
1114 Default).

1115 9.7 Configure the Interface to Use DHCP

1116 An implementation may support attempting to acquire its IP configuration through a DHCP client. A client
1117 can determine whether this functionality is supported and configure the interface to use it as follows:

- 1118 1) Find all instances of CIM_IPAssignmentSettingData (the parent class and not subclasses) that
1119 are associated with the CIM_IPProtocolEndpoint instance.
- 1120 2) For each instance of CIM_IPAssignmentSettingData:
 - 1121 a) Find an instance of CIM_DHCPSettingData that is associated through an instance of
1122 CIM_OrderedComponent.
 - 1123 b) Verify that no instances of CIM_StaticIPAssignmentSettingData are associated with the
1124 instance of CIM_IPAssignmentSettingData.
- 1125 This instance of CIM_IPAssignmentSettingData represents a DHCP configuration.
- 1126 3) Find an instance of CIM_IPConfigurationService that is associated with the
1127 CIM_IPProtocolEndpoint instance through an instance of CIM_ServiceAffectsElement.
- 1128 4) Invoke the ApplySettingToIPProtocolEndpoint() method of the CIM_IPConfigurationService
1129 instance, specifying the instances of CIM_IPProtocolEndpoint and
1130 CIM_IPAssignmentSettingData.

1131 9.8 Establish a Static IP Configuration for an Interface

1132 A client can manually assign an IP configuration to an interface as follows:

- 1133 1) Find all instances of CIM_IPAssignmentSettingData (the parent class and not subclasses) that
1134 are associated with the CIM_IPProtocolEndpoint instance.
- 1135 2) For each instance of CIM_IPAssignmentSettingData:
 - 1136 a) Find an instance of CIM_StaticIPAssignmentSettingData that is associated through an
1137 instance of CIM_OrderedComponent.
 - 1138 b) Verify that no other instances of CIM_StaticIPAssignmentSettingData or instances of
1139 CIM_DHCPSettingData are associated with the instance of CIM_IPAssignmentSettingData
1140 through an instance of CIM_OrderedComponent.
 - 1141 c) For the instance of CIM_ElementSettingData that associates the
1142 CIM_IPAssignmentSettingData instance with the instance of CIM_IPProtocolEndpoint,
1143 verify that the value of the IsDefault property is 2 (Is Not Default).
- 1144 This instance of CIM_IPAssignmentSettingData represents a modifiable, static configuration for
1145 the IP interface.
- 1146 3) Modify the properties of the CIM_StaticIPAssignmentSettingData instance to contain the
1147 appropriate configuration for the IP interface.
- 1148 4) Apply the pending configuration using the steps in 9.9 or 9.10.

1149 9.9 Apply a Pending Configuration — Synchronously

1150 Some implementations may support modifying the configuration of an IP interface without requiring a
1151 restart of the underlying network interface. If this behavior is supported by the implementation, then given
1152 an instance of CIM_IPProtocolEndpoint for which the configuration should be modified and an instance of
1153 CIM_IPAssignmentSettingData that represents the new configuration, a client can:

- 1154 1) Find an instance of CIM_IPConfigurationService that is associated with the
1155 CIM_IPProtocolEndpoint instance through an instance of CIM_ServiceAffectsElement.
- 1156 2) Invoke the ApplySettingToIPProtocolEndpoint() method of the CIM_IPConfigurationService,
1157 specifying the instances of CIM_IPProtocolEndpoint and CIM_IPAssignmentSettingData.

1158 **9.10 Apply a Pending Configuration — Upon Restart**

1159 Some implementations may require that the IP interface be restarted in order for a new configuration that
1160 is bound to the interface to take effect. If an implementation requires that the IP interface be restarted,
1161 then given an instance of CIM_IPProtocolEndpoint for which the configuration should be modified and an
1162 instance of CIM_IPAssignmentSettingData that represents the new configuration, a client can:

- 1163 1) Find an instance of CIM_ElementSettingData that associates the CIM_IPAssignmentSettingData
1164 instance with the CIM_IPProtocolEndpoint instance.
- 1165 2) Set the IsNext property of the CIM_ElementSettingData instance to a value of 1 (Is Next).
- 1166 3) Invoke the RequestStateChange() method of the CIM_IPProtocolEndpoint instance, with a
1167 RequestedState of 11 (Reset).

1168 **9.11 Determine Whether DNS Configuration Was DHCP Assigned**

1169 Starting at the CIM_DNSProtocolEndpoint instance, a client can determine if any elements of the DNS
1170 configuration were assigned through DHCP as follows:

- 1171 1) Find the instance of CIM_IPProtocolEndpoint that is associated through an instance of
1172 CIM_SAPSAPDependency.
- 1173 2) Find the instance of CIM_DHCPProtocolEndpoint that is associated with the
1174 CIM_IPProtocolEndpoint instance through an instance of CIM_SAPSAPDependency.
- 1175 3) Query the EnabledState property of the CIM_DHCPProtocolEndpoint instance for the value 2
1176 (Enabled) to ensure that the DHCP client was used.
- 1177 4) Query the OptionsReceived property of the CIM_DHCPProtocolEndpoint instance to determine
1178 if one of the DNS-related options (8, 14, or 17) was received.

1179 **9.12 Determine Whether ElementName Can Be Modified**

1180 A client can determine whether it can modify the ElementName property of an instance of
1181 CIM_IPProtocolEndpoint as follows:

- 1182 1) Find the CIM_EnabledLogicalElementCapabilities instance that is associated with the
1183 CIM_IPProtocolEndpoint instance.
- 1184 2) Query the value of the ElementNameEditSupported property of the
1185 CIM_EnabledLogicalElementCapabilities instance. If the value is TRUE, the client can modify
1186 the ElementName property of the target instance.

1187 **9.13 Determine Whether State Management Is Supported**

1188 A client can determine whether state management is supported for an instance of
1189 CIM_IPProtocolEndpoint as follows:

- 1190 1) Find the CIM_EnabledLogicalElementCapabilities instance that is associated with the
1191 CIM_IPProtocolEndpoint instance.
- 1192 2) Query the value of the RequestedStatesSupported property. If at least one value is specified,
1193 state management is supported.

1194 **10 CIM Elements**

1195 Table 16 shows the instances of CIM Elements for this profile. Instances of the CIM Elements shall be
 1196 implemented as described in Table 16. Clauses 7 (“Implementation”) and 8 (“Methods”) may impose
 1197 additional requirements on these elements.

1198 **Table 16 – CIM Elements: IP Interface Profile**

Element Name	Requirement	Description
Classes		
CIM_BindsToLANEndpoint	Optional	See 7.6 and 10.1.
CIM_ElementCapabilities	Conditional	See 7.1.2 and 10.2.
CIM_EnabledLogicalElementCapabilities	Optional	See 7.1.2 and 10.5.
CIM_ElementSettingData	Conditional	See 7.4, 10.3, and 10.4.
CIM_HostedAccessPoint	Mandatory	See 10.6 and 10.7.
CIM_HostedService	Conditional	See 7.4.1 and 10.8.
CIM_IPAssignmentSettingData	Conditional	See 7.4 and 10.9.
CIM_IPConfigurationService	Optional	See 7.4 and 10.10.
CIM_IPProtocolEndpoint	Mandatory	See 10.11.
CIM_OrderedComponent	Conditional	See 10.12.
CIM_RegisteredProfile	Mandatory	See 10.13.
CIM_RemoteAccessAvailableToElement	Conditional	See 10.14.
CIM_RemoteServiceAccessPoint	Optional	See 10.15.
CIM_ServiceAffectsElement	Conditional	See 7.4 and 10.16.
CIM_StaticIPAssignmentSettingData	Conditional	See 10.17.
Indications		
None defined in this profile		

1199 **10.1 CIM_BindsToLANEndpoint**

1200 CIM_BindsToLANEndpoint relates the CIM_IPProtocolEndpoint instance with the CIM_LANEndpoint
 1201 instance on which it depends. Table 17 provides information about the properties of
 1202 CIM_BindsToLANEndpoint.

1203 **Table 17 – Class: CIM_BindsToLANEndpoint**

Elements	Requirement	Description
Antecedent	Mandatory	Key: This shall be a reference to an instance of CIM_LANEndpoint. Cardinality 0..1
Dependent	Mandatory	Key: This shall be a reference to the Central Instance. Cardinality 1

1204 **10.2 CIM_ElementCapabilities**

1205 CIM_ElementCapabilities associates an instance of CIM_EnabledLogicalElementCapabilities with the
 1206 CIM_IPProtocolEndpoint instance. Table 18 provides information about the properties of
 1207 CIM_ElementCapabilities.

1208 **Table 18 – Class: CIM_ElementCapabilities**

Elements	Requirement	Description
ManagedElement	Mandatory	Key: This shall be a reference to the Central Instance. Cardinality 1..*
Capabilities	Mandatory	Key: This shall be a reference to the instance of CIM_EnabledLogicalElementCapabilities. Cardinality 0..1

1209 **10.3 CIM_ElementSettingData — CIM_IPAssignmentSettingData Reference**

1210 CIM_ElementSettingData associates instances of CIM_IPAssignmentSettingData with the
 1211 CIM_IPProtocolEndpoint instance. Table 19 provides information about the properties of
 1212 CIM_ElementSettingData.

1213 **Table 19 – Class: CIM_ElementSettingData — CIM_IPAssignmentSettingData**

Elements	Requirement	Description
ManagedElement	Mandatory	Key: This shall be a reference to the Central Instance. Cardinality 1..*
SettingData	Mandatory	Key: This shall be a reference to an instance of CIM_IPAssignmentSettingData. Cardinality *
IsDefault	Mandatory	Matches 1 (Is Default) or 2 (Is Not Default)
IsCurrent	Mandatory	Matches 1 (Is Current) or 2 (Is Not Current)
IsNext	Mandatory	Matches 1 (Is Next), 2 (Is Not Next), or 3 (Is Next For Single Use)

1214 **10.4 CIM_ElementSettingData — CIM_StaticIPAssignmentSettingData Reference**

1215 CIM_ElementSettingData associates instances of CIM_StaticIPAssignmentSettingData with the
 1216 CIM_IPProtocolEndpoint instance. Table 20 provides information about the properties of
 1217 CIM_ElementSettingData.

1218 **Table 20 – Class: CIM_ElementSettingData — CIM_StaticIPAssignmentSettingData**

Elements	Requirement	Description
ManagedElement	Mandatory	Key: This shall be a reference to the Central Instance. Cardinality 1..*

Elements	Requirement	Description
SettingData	Mandatory	Key: This shall be a reference to an instance of CIM_StaticIPAssignmentSettingData. Cardinality *
IsCurrent	Mandatory	Matches 1 (Is Current) or 2 (Is Not Current)

1219 10.5 CIM_EnabledLogicalElementCapabilities

1220 CIM_EnabledLogicalElementCapabilities indicates support for managing the IP interface. Table 21
1221 provides information about the properties of CIM_EnabledLogicalElementCapabilities.

1222 **Table 21 – Class: CIM_EnabledLogicalElementCapabilities**

Elements	Requirement	Description
InstanceID	Mandatory	Key
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.

1223 10.6 CIM_HostedAccessPoint — CIM_RemoteServiceAccessPoint Reference

1224 An instance of CIM_HostedAccessPoint Association between an instance of CIM_ProtocolEndpoint and
1225 CIM_RemoteServiceAccessPoint shall only be instantiated if CIM_RemoteServiceAccessPoint is
1226 supported.

1227 CIM_HostedAccessPoint relates the CIM_RemoteServiceAccessPoint instance that represents the
1228 default gateway with its scoping CIM_ComputerSystem instance. Table 22 provides information about the
1229 properties of CIM_HostedAccessPoint.

1230 **Table 22 – Class: CIM_HostedAccessPoint — CIM_RemoteServiceAccessPoint**

Elements	Requirement	Description
Antecedent	Mandatory	Key: This shall be a reference to the Scoping Instance. Cardinality 1
Dependent	Mandatory	Key: This shall be a reference to an instance of CIM_RemoteServiceAccessPoint. Cardinality *

1231 **10.7 CIM_HostedAccessPoint — CIM_IPProtocolEndpoint Reference**

1232 CIM_HostedAccessPoint relates the Central Instance with its Scoping Instance. Table 23 provides
 1233 information about the properties of CIM_HostedAccessPoint.

1234 **Table 23 – Class: CIM_HostedAccessPoint — CIM_IPProtocolEndpoint**

Elements	Requirement	Description
Antecedent	Mandatory	Key: This shall be a reference to the Central Instance. Cardinality 1
Dependent	Mandatory	Key: This shall be a reference to an instance of CIM_RemoteServiceAccessPoint. Cardinality 1..*

1235 **10.8 CIM_HostedService**

1236 CIM_HostedService relates the CIM_IPConfigurationService instance to its scoping
 1237 CIM_ComputerSystem instance. Table 24 provides information about the properties of
 1238 CIM_HostedService.

1239 **Table 24 – Class: CIM_HostedService**

Elements	Requirement	Description
Antecedent	Mandatory	Key: This shall be a reference to the Central Instance. Cardinality 1
Dependent	Mandatory	Key: This shall be a reference to an instance of CIM_IPConfigurationService. Cardinality *

1240 **10.9 CIM_IPAssignmentSettingData**

1241 CIM_IPAssignmentSettingData is the aggregation point for the SettingData instances that define a
 1242 configuration that can be applied to an IP interface. Table 25 provides information about the properties of
 1243 CIM_IPAssignmentSettingData.

1244 **Table 25 – Class: CIM_IPAssignmentSettingData**

Elements	Requirement	Description
InstanceID	Mandatory	Key
AddressOrigin	Mandatory	Matches 2 (Not Applicable)
ElementName	Mandatory	Pattern ".*"

1245 **10.10 CIM_IPConfigurationService**

1246 CIM_IPConfigurationService represents the ability to configure an IP interface. Table 26 provides
 1247 information about the properties of CIM_IPConfigurationService.

1248 **Table 26 – Class: CIM_IPConfigurationService**

Elements	Requirement	Description
SystemCreationClassName	Mandatory	Key
CreationClassName	Mandatory	Key
SystemName	Mandatory	Key
Name	Mandatory	Key
ElementName	Mandatory	Pattern ".*"
ApplySettingToIPProtocolEndpoint()	Optional	See 8.1.1.1.

1249 **10.11 CIM_IPProtocolEndpoint**

1250 CIM_IPProtocolEndpoint represents an IP interface that is associated with an Ethernet interface. Table 27
 1251 provides information about the properties of CIM_IPProtocolEndpoint.

1252 **Table 27 – Class: CIM_IPProtocolEndpoint**

Elements	Requirement	Description
SystemCreationClassName	Mandatory	Key
CreationClassName	Mandatory	Key
SystemName	Mandatory	Key
Name	Mandatory	Key
NameFormat	Mandatory	Pattern ".*"
ProtocolIFTType	Mandatory	See 7.1.1.2.
RequestedState	Mandatory	See 7.1.2.2 and 7.1.3.2.
EnabledState	Mandatory	See 7.1.2.3 and 7.1.3.3.
ElementName	Mandatory	Pattern ".*"
RequestStateChange()	Conditional	See 8.1.
IPv4Address	Conditional	See 7.1.1.2.
SubnetMask	Conditional	See 7.1.1.2 and 7.1.1.4.
AddressOrigin	Mandatory	See 7.1.1.1.
IPv6Address	Conditional	See 7.1.1.2 and 7.1.1.5.
IPv6AddressType	Conditional	See 7.1.1.2.
IPv6SubnetPrefixLength	Conditional	See 7.1.1.2.

1253 **10.12 CIM_OrderedComponent**

1254 CIM_OrderedComponent associates an instance of CIM_IPAssignmentSettingData to the instances of
 1255 CIM_StaticIPAssignmentSettingData, CIM_DHCPSettingData, CIM_DNSSettingData, and

1256 CIM_DNSGeneralSettingData that compose a configuration. Table 28 provides information about the
 1257 properties of CIM_OrderedComponent.

1258 **Table 28 – Class: CIM_OrderedComponent**

Elements	Requirement	Description
GroupComponent	Mandatory	Key: See 7.4.3.1.
PartComponent	Mandatory	Key: See 7.4.3.2.
AssignedSequence	Mandatory	See 7.4.3.3.

1259 **10.13 CIM_RegisteredProfile**

1260 CIM_RegisteredProfile identifies the *IP Interface Profile* in order for a client to determine whether an
 1261 instance of CIM_IPProtocolEndpoint is conformant with this profile. The CIM_RegisteredProfile class is
 1262 defined by the [Profile Registration Profile](#). With the exception of the mandatory values specified for the
 1263 properties in Table 29, the behavior of the CIM_RegisteredProfile instance is in accordance with the
 1264 [Profile Registration Profile](#).

1265 **Table 29 – Class: CIM_RegisteredProfile**

Elements	Requirement	Description
RegisteredName	Mandatory	This property shall have a value of "IP Interface".
RegisteredVersion	Mandatory	This property shall have a value of "1.1.0".
RegisteredOrganization	Mandatory	This property shall have a value of "DMTF".

1266 NOTE: Previous versions of this document included the suffix "Profile" for the RegisteredName value. If
 1267 implementations querying for the RegisteredName value find the suffix "Profile", they should ignore the suffix, with
 1268 any surrounding white spaces, before any comparison is done with the value as specified in this document.

1269 **10.14 CIM_RemoteAccessAvailableToElement**

1270 CIM_RemoteAccessAvailableToElement associates the CIM_IPProtocolEndpoint instance with the
 1271 CIM_RemoteServiceAccessPoint instance that represents the network gateway. Table 30 provides
 1272 information about the properties of CIM_RemoteAccessAvailableToElement.

1273 **Table 30 – Class: CIM_RemoteAccessAvailableToElement**

Elements	Requirement	Description
Antecedent	Mandatory	Key: See 7.1.6.2.
Dependent	Mandatory	Key: See 7.1.6.3.
OrderOfAccess	Mandatory	See 7.1.6.4.

1274 **10.15 CIM_RemoteServiceAccessPoint**

1275 CIM_RemoteServiceAccessPoint represents the managed system's view of the default gateway. Table
 1276 31 provides information about the properties of CIM_RemoteServiceAccessPoint.

1277 **Table 31 – Class: CIM_RemoteServiceAccessPoint**

Elements	Requirement	Description
SystemCreationClassName	Mandatory	Key
CreationClassName	Mandatory	Key
SystemName	Mandatory	Key
Name	Mandatory	Key
AccessContext	Mandatory	Matches 2 (Default Gateway)
AccessInfo	Mandatory	See 7.1.6.1.
InfoFormat	Mandatory	Matches 3 (IPv4 Address)
ElementName	Mandatory	Pattern ".*"

1278 **10.16 CIM_ServiceAffectsElement**

1279 CIM_ServiceAffectsElement associates an instance of CIM_IPConfigurationService with an instance of
 1280 CIM_IPProtocolEndpoint that the service is able to configure. Table 32 provides information about the
 1281 properties of CIM_ServiceAffectsElement.

1282 **Table 32 – Class: CIM_ServiceAffectsElement**

Elements	Requirement	Description
AffectingElement	Mandatory	Key: This shall be a reference to the instance of CIM_IPConfigurationService. Cardinality *
AffectedElement	Mandatory	Key: This shall be a reference to the Central Instance. Cardinality 1..*
ElementAffects	Mandatory	Matches 5 (Manages)

1283 **10.17 CIM_StaticIPAssignmentSettingData**

1284 CIM_StaticIPAssignmentSettingData represents a static configuration that can be applied to an instance
 1285 of CIM_IPProtocolEndpoint. Table 33 provides information about the properties of
 1286 CIM_StaticIPAssignmentSettingData.

1287 **Table 33 – Class: CIM_StaticIPAssignmentSettingData**

Elements	Requirement	Description
InstanceID	Mandatory	Key
AddressOrigin	Mandatory	Matches 3 (Static)
ElementName	Mandatory	Pattern ".*"
IPv4Address	Mandatory	
SubnetMask	Mandatory	

Elements	Requirement	Description
GatewayIPv4Address	Conditional	See 7.5.3.1.
IPv6Address	Optional	
IPv6AddressType	Optional	
IPv6SubnetPrefixLength	Optional	
GatewayIPv6Address	Optional	

1288

**ANNEX A
(informative)****Change Log**

Version	Date	Description
1.0.0a	2006-07-11	Preliminary Standard
1.0.0	2008-07-27	Final Standard & addition of IPv6 support as Experimental
1.0.1	2009-09-24	Errata 1.0.1
1.1.0	2010-10-21	Experimental sections were removed, and the document was formatted for DMTF Standard release.

1289
1290
1291
12921293
1294