



1

2

3

4

Document Number: DSP1036

Date: 2012-02-23

Version: 1.1.1

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, 2012 Distributed Management Task Force, Inc. (DMTF). All rights reserved.

12 DMTF is a not-for-profit association of industry members dedicated to promoting enterprise and systems
13 management and interoperability. Members and non-members may reproduce DMTF specifications and
14 documents, provided that correct attribution is given. As DMTF specifications may be revised from time
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26 withdrawn or modified after publication, and shall be indemnified and held harmless by any party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 | 10 |
| 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 | 14 |
| 45 | 7.2 DHCP client is supported..... | 17 |
| 46 | 7.3 DNS Client is supported..... | 17 |
| 47 | 7.4 Managing alternate configurations — Optional | 18 |
| 48 | 7.5 Applying an alternate configuration | 21 |
| 49 | 7.6 Relationship with a network interface | 23 |
| 50 | 8 Methods..... | 23 |
| 51 | 8.1 CIM_IPProtocolEndpoint.RequestStateChange() | 23 |
| 52 | 8.2 CIM_IPConfigurationService.ApplySettingToIPProtocolEndpoint() | 24 |
| 53 | 8.3 Profile conventions for operations | 25 |
| 54 | 8.4 CIM_BindsToLANEndpoint..... | 25 |
| 55 | 8.5 CIM_ElementSettingData | 26 |
| 56 | 8.6 CIM_HostedAccessPoint | 27 |
| 57 | 8.7 CIM_HostedService | 27 |
| 58 | 8.8 CIM_IPAssignmentSettingData | 27 |
| 59 | 8.9 CIM_IPConfigurationService | 27 |
| 60 | 8.10 CIM_IPProtocolEndpoint | 28 |
| 61 | 8.11 CIM_OrderedComponent | 28 |
| 62 | 8.12 CIM_RemoteAccessAvailableToElement..... | 29 |
| 63 | 8.13 CIM_RemoteServiceAccessPoint..... | 29 |
| 64 | 8.14 CIM_ServiceAffectsElement..... | 29 |
| 65 | 8.15 CIM_StaticIPAssignmentSettingData | 29 |
| 66 | 9 Use cases..... | 31 |
| 67 | 9.1 Miscellaneous object diagrams..... | 31 |
| 68 | 9.2 Determine supported configuration methods..... | 44 |
| 69 | 9.3 Determine gateway address | 44 |
| 70 | 9.4 Determine method used for current configuration | 44 |
| 71 | 9.5 Determine whether DHCP then static is supported | 44 |
| 72 | 9.6 View default configuration..... | 44 |
| 73 | 9.7 Configure the interface to use DHCP | 45 |
| 74 | 9.8 Establish a static IP configuration for an interface..... | 45 |
| 75 | 9.9 Apply a pending configuration — Synchronously..... | 45 |
| 76 | 9.10 Apply a pending configuration — Upon restart..... | 46 |
| 77 | 9.11 Determine whether DNS configuration was DHCP assigned | 46 |
| 78 | 9.12 Determine whether ElementName can be modified | 46 |
| 79 | 9.13 Determine whether state management is supported | 46 |
| 80 | 10 CIM Elements..... | 47 |
| 81 | 10.1 CIM_BindsToLANEndpoint..... | 47 |
| 82 | 10.2 CIM_ElementCapabilities | 48 |
| 83 | 10.3 CIM_ElementSettingData — CIM_IPAssignmentSettingData Reference | 48 |
| 84 | 10.4 CIM_ElementSettingData — CIM_StaticIPAssignmentSettingData Reference | 48 |
| 85 | 10.5 CIM_EnabledLogicalElementCapabilities..... | 49 |

| | | |
|----|---|----|
| 86 | 10.6 CIM_HostedAccessPoint — CIM_RemoteServiceAccessPoint Reference | 49 |
| 87 | 10.7 CIM_HostedAccessPoint — CIM_IPProtocolEndpoint Reference | 50 |
| 88 | 10.8 CIM_HostedService | 50 |
| 89 | 10.9 CIM_IPAssignmentSettingData | 50 |
| 90 | 10.10 CIM_IPConfigurationService | 51 |
| 91 | 10.11 CIM_IPProtocolEndpoint | 51 |
| 92 | 10.12 CIM_OrderedComponent | 51 |
| 93 | 10.13 CIM_RegisteredProfile | 52 |
| 94 | 10.14 CIM_RemoteAccessAvailableToElement | 52 |
| 95 | 10.15 CIM_RemoteServiceAccessPoint | 53 |
| 96 | 10.16 CIM_ServiceAffectsElement | 53 |
| 97 | 10.17 CIM_StaticIPAssignmentSettingData | 53 |
| 98 | ANNEX A (informative) Change log | 55 |

99

100 Figures

| | | |
|-----|--|----|
| 101 | Figure 1 – IP Interface Profile: Class diagram | 12 |
| 102 | Figure 2 – Registered profile | 31 |
| 103 | Figure 3 – Basic configuration — IPv4 | 32 |
| 104 | Figure 4 – Basic configuration — IPv6 | 33 |
| 105 | Figure 5 – Basic configuration — IPv4 and IPv6 | 34 |
| 106 | Figure 6 – Static current and pending configuration | 35 |
| 107 | Figure 7 – Static and DHCP pending configurations | 36 |
| 108 | Figure 8 – DHCP timed out to a static configuration | 37 |
| 109 | Figure 9 – Service processor and server share an NIC | 38 |
| 110 | Figure 10 – Configuration choices | 39 |
| 111 | Figure 11 – DHCP assigned partial DNS | 40 |
| 112 | Figure 12 – DHCP with DNS statically configured | 41 |
| 113 | Figure 13 – Static without DNS configuration — One | 42 |
| 114 | Figure 14 – Static without DNS configuration — Two | 43 |
| 115 | Figure 15 – Static without DNS configuration — Three | 43 |

116

117 Tables

| | | |
|-----|---|----|
| 118 | Table 1 – Referenced profiles | 12 |
| 119 | Table 2 – CIM_IPProtocolEndpoint.RequestStateChange() Method: Return code values | 23 |
| 120 | Table 3 – CIM_IPProtocolEndpoint.RequestStateChange() Method: Parameters | 24 |
| 121 | Table 4 – CIM_IPConfigurationService.ApplySettingToIPProtocolEndpoint() Method: Return code | |
| 122 | values | 24 |
| 123 | Table 5 – CIM_IPConfigurationService.ApplySettingToIPProtocolEndpoint() Method: Parameters | 24 |
| 124 | Table 6 – Operations: CIM_BindsToLANEndpoint | 25 |
| 125 | Table 7 – Operations: CIM_ElementSettingData | 26 |
| 126 | Table 8 – Operations: CIM_HostedAccessPoint | 27 |
| 127 | Table 9 – Operations: CIM_HostedService | 27 |
| 128 | Table 10 – Operations: CIM_IPProtocolEndpoint | 28 |
| 129 | Table 11 – Operations: CIM_OrderedComponent | 28 |
| 130 | Table 12 – Operations: CIM_RemoteAccessAvailableToElement | 29 |
| 131 | Table 13 – Operations: CIM_ServiceAffectsElement | 29 |

132 Table 14 – Operations: CIM_StaticIPAssignmentSettingData..... 30

133 Table 15 – CIM Elements: IP interface profile 47

134 Table 16 – Class: CIM_BindsToLANEndpoint 47

135 Table 17 – Class: CIM_ElementCapabilities 48

136 Table 18 – Class: CIM_ElementSettingData — CIM_IPAssignmentSettingData 48

137 Table 19 – Class: CIM_ElementSettingData — CIM_StaticIPAssignmentSettingData 48

138 Table 20 – Class: CIM_EnabledLogicalElementCapabilities 49

139 Table 21 – Class: CIM_HostedAccessPoint — CIM_RemoteServiceAccessPoint 49

140 Table 22 – Class: CIM_HostedAccessPoint — CIM_IPProtocolEndpoint 50

141 Table 23 – Class: CIM_HostedService 50

142 Table 24 – Class: CIM_IPAssignmentSettingData 50

143 Table 25 – Class: CIM_IPConfigurationService..... 51

144 Table 26 – Class: CIM_IPProtocolEndpoint..... 51

145 Table 27 – Class: CIM_OrderedComponent..... 52

146 Table 28 – Class: CIM_RegisteredProfile 52

147 Table 29 – Class: CIM_RemoteAccessAvailableToElement 52

148 Table 30 – Class: CIM_RemoteServiceAccessPoint 53

149 Table 31 – Class: CIM_ServiceAffectsElement 53

150 Table 32 – Class: CIM_StaticIPAssignmentSettingData 53

151

152

Foreword

153 The *IP Interface Profile* (DSP1036) was prepared by the Server Management Working Group, the
154 Physical Platform Profiles Working Group and the Server Desktop Mobile Platforms Working Group of the
155 DMTF.

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

158 Acknowledgments

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

160 Editors:

- 161 • Hemal Shah – Broadcom
- 162 • Jeff Hilland – Hewlett-Packard Company
- 163 • Aaron Merkin – IBM
- 164 • Satheesh Thomas – AMI

165 Contributors:

- 166 • RadhaKrishna Dasari – Dell
- 167 • Jon Hass – Dell
- 168 • John Leung – Intel
- 169 • Khachatur Papanyan – Dell
- 170 • Christina Shaw – Hewlett-Packard Company
- 171 • Enoch Suen – Dell
- 172 • Perry Vincent – Intel

173

174

Introduction

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

180 **Document conventions**

181 **Typographical conventions**

182 The following typographical conventions are used in this document:

- 183 • Document titles are marked in *italics*.
- 184

185

186

187

IP Interface Profile

188 1 Scope

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

193 2 Normative references

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

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

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

202 DMTF DSP0223, *Generic Operations 1.0*,
203 http://www.dmtf.org/standards/published_documents/DSP0223_1.0.pdf

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

206 DMTF DSP1004, *Base Server Profile 1.0*,
207 http://www.dmtf.org/standards/published_documents/DSP1004_1.0.pdf

208 DMTF DSP1014, *Ethernet Port Profile 1.0*,
209 http://www.dmtf.org/standards/published_documents/DSP1014_1.0.pdf

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

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

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

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

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

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

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

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

223 3 Terms and definitions

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

226 The terms "shall" ("required"), "shall not," "should" ("recommended"), "should not" ("not recommended"),
227 "may," "need not" ("not required"), "can" and "cannot" in this document are to be interpreted as described
228 in [ISO/IEC Directives, Part 2](#), Annex H. The terms in parenthesis are alternatives for the preceding term,
229 for use in exceptional cases when the preceding term cannot be used for linguistic reasons. Note that
230 [ISO/IEC Directives, Part 2](#), Annex H specifies additional alternatives. Occurrences of such additional
231 alternatives shall be interpreted in their normal English meaning.

232 The terms "clause," "subclause," "paragraph," and "annex" in this document are to be interpreted as
233 described in [ISO/IEC Directives, Part 2](#), Clause 5.

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

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

239 3.1

240 **can**

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

242 3.2

243 **cannot**

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

245 3.3

246 **conditional**

247 indicates requirements to be followed strictly to conform to the document when the specified conditions
248 are met

249 3.4

250 **mandatory**

251 indicates requirements to be followed strictly to conform to the document and from which no deviation is
252 permitted

253 3.5

254 **may**

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

256 3.6

257 **need not**

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

259 3.7

260 **optional**

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

262 3.8

263 **pending configuration**

264 the configuration that will be applied to an IP interface the next time the interface accepts a configuration

- 265 **3.9**
266 **referencing profile**
267 indicates a profile that owns the definition of this class and can include a reference to this profile in its
268 “Referenced Profiles” table
- 269 **3.10**
270 **shall**
271 indicates requirements to be followed strictly to conform to the document and from which no deviation is
272 permitted
- 273 **3.11**
274 **shall not**
275 indicates requirements to be followed strictly to conform to the document and from which no deviation is
276 permitted
- 277 **3.12**
278 **should**
279 indicates that among several possibilities, one is recommended as particularly suitable, without
280 mentioning or excluding others, or that a certain course of action is preferred but not necessarily required
- 281 **3.13**
282 **should not**
283 indicates that a certain possibility or course of action is deprecated but not prohibited
- 284 **3.14**
285 **unspecified**
286 indicates that this profile does not define any constraints for the referenced CIM element or operation

287 **4 Symbols and abbreviated terms**

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

- 290 **4.1**
291 **DHCP**
292 Dynamic Host Configuration Protocol
- 293 **4.2**
294 **DNS**
295 Domain Name System
- 296 **4.3**
297 **IP**
298 Internet Protocol

299 **5 Synopsis**

- 300 **Profile name:** IP Interface
301 **Version:** 1.1.1
302 **Organization:** DMTF
303 **CIM Schema version:** 2.26
304 **Central class:** CIM_IPProtocolEndpoint

305 **Scoping class:** CIM_ComputerSystem

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

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

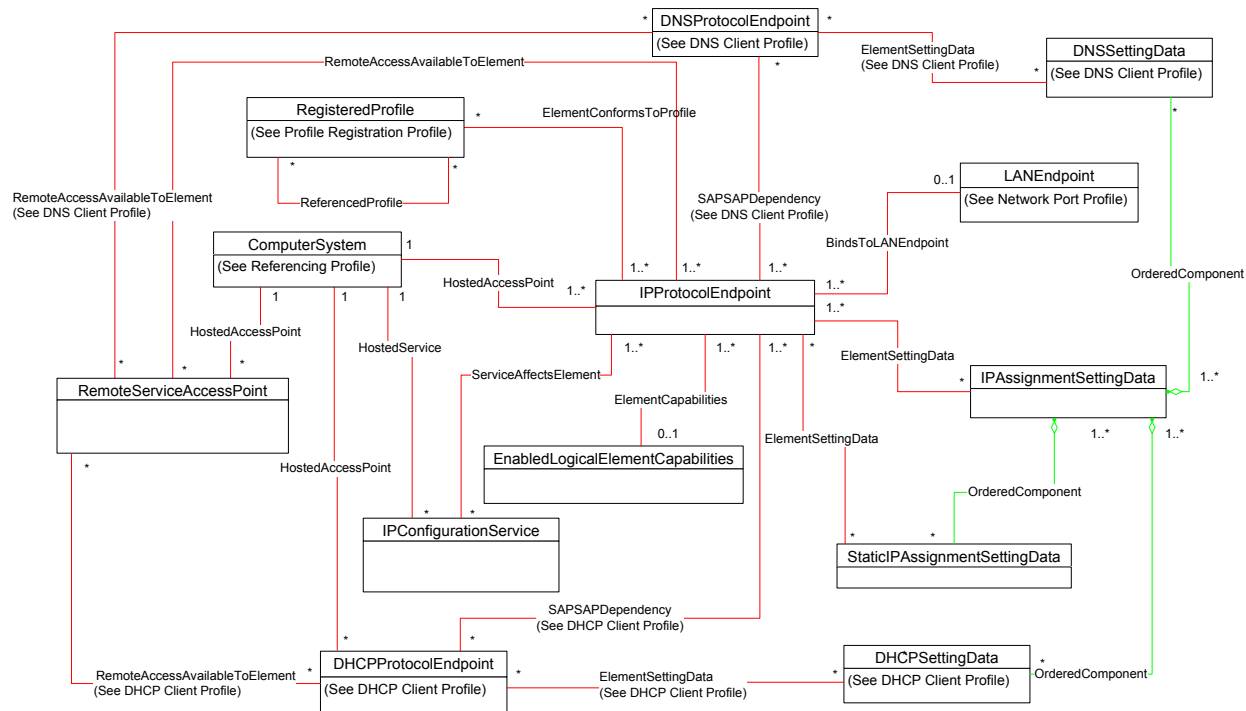
311 **Table 1 – Referenced profiles**

| 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. |

312 6 Description

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

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



318

319

Figure 1 – IP Interface Profile: Class diagram

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

- 323 • IPv4 interface (optionally associated with a network interface)
- 324 • optional relationship with a DNS client
- 325 • optional relationship with a DHCP client
- 326 • current and pending configurations

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

- 328 • modeling of the network gateway
- 329 • modeling of TCP/UDP ports

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

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

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

339 6.1 Pending and alternate configuration management

340 Pending configurations, which are associated with the IP interface and could be applied in the future, are
341 represented by instances of `CIM_IPAssignmentSettingData` and its subclasses. Each pending
342 configuration can include multiple settings that will be applied to the different elements of the endpoint
343 configuration. Settings for a particular element of the configuration are represented with the appropriate
344 subclass of `CIM_IPAssignmentSettingData` and aggregated into one or more instances of
345 `CIM_IPAssignmentSettingData` that represent the configuration.

346 The management of DNS and DHCP clients as part of an alternate configuration is handled differently for
347 the two clients. DHCP and static IP configuration management are generally treated as alternatives to
348 each other. For the basic configuration of an IP interface, the information is assigned either statically or
349 through DHCP. DNS configuration occurs differently. When DNS and static configuration occur together,
350 there is no overlap. Thus the DNS settings that are part of the configuration are applied to the DNS client.
351 When DHCP and DNS settings are used together, portions of the DNS configuration can potentially be
352 assigned through DHCP.

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

359 7 Implementation

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

362 7.1 Basic IP configuration

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

364 7.1.1 CIM_IPProtocolEndpoint

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

367 7.1.1.1 CIM_IPProtocolEndpoint.AddressOrigin

368 The AddressOrigin property indicates the configuration method that resulted in the configuration being
369 assigned to the CIM_IPProtocolEndpoint.

370 7.1.1.1.1 AddressOrigin — Static

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

375 7.1.1.1.2 AddressOrigin — DHCP

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

379 7.1.1.2 CIM_IPProtocolEndpoint.ProtocolIFType

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

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

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

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

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

388 7.1.1.3 CIM_IPProtocolEndpoint.IPv4Address

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

393 7.1.1.4 CIM_IPProtocolEndpoint.SubnetMask

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

397 7.1.1.5 CIM_IPProtocolEndpoint.IPv6Address

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

401 7.1.2 IP interface state management is supported — Conditional

402 When management of the state of an IP interface is supported, exactly one instance of
403 CIM_EnabledLogicalElementCapabilities shall be associated with the CIM_IPProtocolEndpoint instance
404 through an instance of CIM_ElementCapabilities. The existence of the CIM_ElementCapabilities instance
405 is conditional on the existence of the CIM_EnabledLogicalElementCapabilities instance.

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

408 7.1.2.1 CIM_EnabledLogicalElementCapabilities

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

411 7.1.2.1.1 CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported

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

414 7.1.2.2 CIM_IPProtocolEndpoint.RequestedState

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

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

421 7.1.2.3 CIM_IPProtocolEndpoint.EnabledState

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

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

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

428 7.1.3 IP interface state management is not supported

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

431 7.1.3.1 CIM_EnabledLogicalElementCapabilities

432 When state management is not supported, exactly one instance of
433 CIM_EnabledLogicalElementCapabilities may be associated with the CIM_IPProtocolEndpoint instance
434 through an instance of CIM_ElementCapabilities.

435 7.1.3.1.1 CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported

436 The CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported property shall not contain any
437 values.

438 7.1.3.2 CIM_IPProtocolEndpoint.RequestedState

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

440 7.1.3.3 CIM_IPProtocolEndpoint.EnabledState

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

443 7.1.4 Modifying ElementName is supported — Conditional

444 The CIM_IPProtocolEndpoint.ElementName property may support being modified by the ModifyInstance
445 operation. See 8.10.1.1.

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

448 7.1.4.1 CIM_EnabledLogicalElementCapabilities

449 An instance of CIM_EnabledLogicalElementCapabilities shall be associated with the
450 CIM_IPProtocolEndpoint instance through an instance of CIM_ElementCapabilities.

451 7.1.4.1.1 CIM_EnabledLogicalElementCapabilities.ElementNameEditSupported

452 The ElementNameEditSupported property shall have a value of TRUE.

453 7.1.4.1.2 CIM_EnabledLogicalElementCapabilities.MaxElementNameLen

454 The MaxElementNameLen property shall be implemented.

455 7.1.5 Modifying ElementName is not supported

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

459 7.1.5.1 CIM_EnabledLogicalElementCapabilities

460 An instance of CIM_EnabledLogicalElementCapabilities may be associated with the
461 CIM_IPProtocolEndpoint instance through an instance of CIM_ElementCapabilities.

462 7.1.5.1.1 CIM_EnabledLogicalElementCapabilities.ElementNameEditSupported

463 The ElementNameEditSupported property shall have a value of FALSE.

464 7.1.5.1.2 CIM_EnabledLogicalElementCapabilities.MaxElementNameLen

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

467 7.1.6 Default gateway

468 An IP interface can be configured with the address of a network gateway. Modeling of the default gateway
469 is optional. When the IP interface is configured with the address of a default gateway, an instance of
470 CIM_RemoteServiceAccessPoint shall represent the default gateway. The instance of

471 CIM_RemoteServiceAccessPoint shall be associated with the instance of CIM_IPProtocolEndpoint
472 through an instance of CIM_RemoteAccessAvailableToElement. An instance of
473 CIM_RemoteServiceAccessPoint may represent the default gateway even when a valid default gateway
474 has not been configured for the IP interface. It can be more convenient for an implementation to always
475 instantiate the instance of CIM_RemoteServiceAccessPoint even if a default gateway has not been
476 assigned to the IP interface rather than conditionally provide the relevant instances. For IPv4, this will
477 result in a single instance of CIM_RemoteServiceAccessPoint associated with the instance of
478 CIM_IPProtocolEndpoint.

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

483 7.1.6.1 CIM_RemoteServiceAccessPoint.AccessInfo

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

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

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

496 7.1.6.2 CIM_RemoteAccessAvailableToElement.Antecedent

497 The value of the Antecedent reference shall be the instance of CIM_RemoteServiceAccessPoint.
498 Cardinality *.

499 7.1.6.3 CIM_RemoteAccessAvailableToElement.Dependent

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

501 7.1.6.4 CIM_RemoteAccessAvailableToElement.OrderOfAccess

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

504 7.2 DHCP client is supported

505 When a DHCP client is supported for the IP interface, the *DHCP Client Profile* ([DSP1037](#)) shall be
506 supported. This behavior is optional.

507 7.3 DNS Client is supported

508 When a DNS client is supported for the IP interface, the *DNS Client Profile* ([DSP1038](#)) shall be
509 supported. This behavior is optional.

510 **7.4 Managing alternate configurations — Optional**

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

514 **7.4.1 Configuration management is supported**

515 The CIM_IPConfigurationService class provides management of alternate configurations and support for
516 configuring additional interfaces. When an implementation supports management of alternate
517 configurations, exactly one instance of CIM_IPConfigurationService shall be associated with the Central
518 Instance of the profile through an instance of CIM_ServiceAffectsElement. The existence of the
519 CIM_ServiceAffectsElement association is conditional on the existence of the
520 CIM_IPConfigurationService instance.

521 The CIM_IPConfigurationService instance shall be associated with a CIM_ComputerSystem instance
522 through an instance of CIM_HostedService. The existence of the CIM_HostedService association is
523 conditional on the existence of the CIM_IPConfigurationService instance.

524 **7.4.2 Representing an alternate configuration using CIM_IPAssignmentSettingData**

525 Each instance of CIM_IPAssignmentSettingData shall represent a possible configuration for an IP
526 interface. The detailed settings for the IP interface shall be contained in the instances of subclasses of
527 CIM_IPAssignmentSettingData, which are associated with the instance of CIM_IPAssignmentSettingData
528 through instances of CIM_OrderedComponent.

529 The existence of one or more instances of CIM_IPAssignmentSettingData is conditional on the existence
530 of the CIM_IPConfigurationService instance. The existence of one or more instances of
531 CIM_ElementSettingData is conditional on the existence of one or more instances of
532 CIM_IPAssignmentSettingData.

533 **7.4.2.1 Associating an alternate configuration with an IP interface**

534 The instance of CIM_IPAssignmentSettingData shall be associated with the instance of
535 CIM_IPProtocolEndpoint through an instance of CIM_ElementSettingData.

536 **7.4.2.1.1 CIM_ElementSettingData.IsCurrent**

537 When an instance of CIM_ElementSettingData associates an instance of CIM_IPAssignmentSettingData
538 with an instance of CIM_IPProtocolEndpoint, the CIM_ElementSettingData.IsCurrent property shall have
539 a value of 1 (Is Current) when the configuration represented by the referenced instance of
540 CIM_IPAssignmentSettingData is the last configuration applied to the IP interface represented by the
541 referenced instance of CIM_IPProtocolEndpoint.

542 When an instance of CIM_ElementSettingData associates an instance of CIM_IPAssignmentSettingData
543 with an instance of CIM_IPProtocolEndpoint, the CIM_ElementSettingData.IsCurrent property shall have
544 a value of 2 (Is Not Current) when the configuration represented by the referenced instance of
545 CIM_IPAssignmentSettingData is not the last configuration applied to the IP interface represented by the
546 referenced instance of CIM_IPProtocolEndpoint.

547 **7.4.3 Associating settings using CIM_OrderedComponent**

548 Instances of the subclasses of CIM_IPAssignmentSettingData contain the details of the IP configuration.
549 The CIM_OrderedComponent association aggregates these instances into instances of
550 CIM_IPAssignmentSettingData. An instance of CIM_IPAssignmentSettingData will have one or more
551 instances of its subclasses associated with it through an instance of CIM_OrderedComponent. An
552 instance of a subclass of CIM_IPAssignmentSettingData will be associated with one or more instances of
553 CIM_IPAssignmentSettingData.

554 7.4.3.1 CIM_OrderedComponent.GroupComponent

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

557 7.4.3.2 CIM_OrderedComponent.PartComponent

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

560 7.4.3.3 Interpretation of CIM_OrderedComponent.AssignedSequence

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

564 7.4.3.3.1 Use of 0 (zero)

565 When the CIM_OrderedComponent.AssignedSequence property has a value of 0 (zero), the instance of
566 CIM_SettingData referenced by the CIM_OrderedComponent.PartComponent property shall not be
567 applied when the configuration represented by the CIM_IPAssignmentSettingData instance that is the
568 value of the CIM_OrderedComponent.GroupComponent property is applied. The
569 CIM_OrderedComponent.AssignedSequence property may have the value 0 (zero) when the instance of
570 CIM_OrderedComponent references an instance of CIM_DNSSettingData or
571 CIM_DNSGeneralSettingData. The CIM_OrderedComponent.AssignedSequence property shall not have
572 the value 0 (zero) when the instance of CIM_OrderedComponent does not reference an instance of
573 CIM_DNSSettingData or CIM_DNSGeneralSettingData.

574 7.4.3.3.2 Discreteness

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

578 7.4.4 Alternate static configuration

579 When an implementation supports the manual assignment of an IP configuration to the IP endpoint, an
580 instance of CIM_StaticIPAssignmentSettingData shall be associated with the CIM_IPProtocolEndpoint
581 through an instance of CIM_ElementSettingData. This instance of CIM_StaticIPAssignmentSettingData
582 shall be associated with at least one instance of CIM_IPAssignmentSettingData through an instance of
583 CIM_OrderedComponent. When the aggregating IP configuration has been applied to the IP interface
584 and the IP interface is using the settings contained in the instance of
585 CIM_StaticIPAssignmentSettingData, the IsCurrent property of the CIM_ElementSettingData instance
586 has the value 1 (Is Current). Otherwise, the CIM_ElementSettingData.IsCurrent property shall have the
587 value 2 (Is Not Current).

588 7.4.5 Alternate DHCP configuration

589 When an alternate configuration includes the configuration of the DHCP client, the implementation will
590 follow the rules for representing a pending DHCP configuration defined in the *DHCP Client Profile*
591 ([DSP1037](#)).

592 7.4.6 DNS client alternate configuration

593 When an alternate configuration includes the configuration of the DNS client, the implementation will
594 follow the rules for representing a pending DNS configuration defined in the *DNS Client Profile*
595 ([DSP1038](#)).

596 7.4.7 Relationship between DHCP and DNS configuration

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

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

604 7.4.7.1 Relationship between DHCP options and the DNS configuration

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

608 DHCPPE – the instance of CIM_DHCPProtocolEndpoint that represents the DHCP client for an IP
609 interface

610 DNSPE – the instance of CIM_DNSProtocolEndpoint that represents the DNS client that is associated
611 through an instance of CIM_SAPSAPDependency with the same instance of CIM_IPProtocolEndpoint
612 with which the DHCPPE is associated through an instance of CIM_SAPSAPDependency

613 DNS Pending – the instance of CIM_DNSSettingData that is associated through an instance of
614 CIM_OrderedComponent with the instance of CIM_IPAssignmentSettingData that is being applied to the
615 CIM_IPProtocolEndpoint instance

616 DHCP Pending – the instance of CIM_DHCPSettingData that is associated through an instance of
617 CIM_OrderedComponent with the instance of CIM_IPAssignmentSettingData that is being applied to the
618 CIM_IPProtocolEndpoint instance

619 The following requirements shall be met when the *DHCP Client Profile* ([DSP1037](#)) and the *DNS Client*
620 *Profile* ([DSP1038](#)) are implemented:

- 621 • When the OptionsReceived property of the DHCPPE instance and the DHCPOptionsToUse
622 property of the DNSPE instance both contain the value 8 (Domain Name Server), the DNS
623 Servers instrumented in accordance with the "DNS Server Representation" section of the *DNS*
624 *Client Profile* ([DSP1038](#)) shall identify the DNS server addresses specified by the DHCP server
625 as the data for the Domain Name Server DHCP option.
- 626 • When the OptionsReceived property of the DHCPPE instance and the DHCPOptionsToUse
627 property of the DNSPE instance both contain the value 14 (Host Name), the value of the
628 Hostname property of the DNSPE instance shall be the hostname specified by the DHCP server
629 as the data for the Host Name DHCP option.
- 630 • When the OptionsReceived property of the DHCPPE instance and the DHCPOptionsToUse
631 property of the DNSPE instance both contain the value 17 (Domain Name), the value of the
632 DomainName property of the DNSPE instance shall be the domain name specified by the DHCP
633 server as the data for the Domain Name DHCP option.

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

638 7.4.8 Representing a pending configuration

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

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

645 If an instance of CIM_IPAssignmentSettingData is associated with the Central Instance through an
646 instance of CIM_ElementSettingData whose IsNext property has the value 3 (Is Next For Single Use), this
647 instance of CIM_IPAssignmentSettingData shall represent the pending configuration. If no instance of
648 CIM_IPAssignmentSettingData is associated with the Central Instance through an instance of
649 CIM_ElementSettingData whose IsNext property has the value 3 (Is Next For Single Use), the instance of
650 CIM_IPAssignmentSettingData that is associated with the Central Instance through an instance of
651 CIM_ElementSettingData whose IsNext property has the value 1 (Is Next) shall represent the pending
652 configuration.

653 7.5 Applying an alternate configuration

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

660 7.5.1 Applying the pending configuration upon transition to enabled

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

664 7.5.2 Determining the target CIM_ProtocolEndpoint instance

665 An instance of CIM_IPAssignmentSettingData or its subclasses may be associated with more than one
666 instance of a subclass of CIM_ProtocolEndpoint through instances of CIM_ElementSettingData.
667 Instances of subclasses of CIM_IPAssignmentSettingData may be aggregated into one or more instances
668 of CIM_IPAssignmentSettingData where the aggregating CIM_IPAssignmentSettingData instances are
669 associated with different instances of CIM_IPProtocolEndpoint. This is allowed as a convenience for
670 instrumentation to reduce the number of instances required when multiple IP interfaces share a common
671 configuration.

672 The following rules unambiguously identify the instance of a subclass of CIM_ProtocolEndpoint that will
673 have an instance of a subclass of CIM_SettingData applied to it when a pending configuration is applied
674 to an instance of CIM_IPProtocolEndpoint. Note that the DNS and DHCP related classes are owned by
675 the *DNS Client Profile* ([DSP1038](#)) and *DHCP Client Profile* ([DSP1037](#)) , respectively. The algorithm for
676 determining their use is provided here because it is part of the behavior of applying a configuration.

677 When a pending IP configuration is applied, each instance of CIM_StaticIPAssignmentSettingData that is
678 associated with the CIM_IPAssignmentSettingData instance through an instance of
679 CIM_OrderedComponent shall be applied to the CIM_IPProtocolEndpoint instance that is identified as
680 follows:

- 681 1) The CIM_IPProtocolEndpoint instance shall be associated with the
682 CIM_StaticIPAssignmentSettingData instance through an instance of CIM_ElementSettingData.

683 2) The CIM_IPProtocolEndpoint instance shall be the CIM_IPProtocolEndpoint instance to which
684 the aggregating CIM_IPAssignmentSettingData is being applied.

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

688 1) The CIM_DHCPProtocolEndpoint instance shall be associated with the CIM_DHCPSettingData
689 instance through an instance of CIM_ElementSettingData.

690 2) The CIM_DHCPProtocolEndpoint instance shall be associated through an instance of
691 CIM_SAPSAPDependency with the CIM_IPProtocolEndpoint instance to which the aggregating
692 CIM_IPAssignmentSettingData is being applied.

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

696 1) The CIM_DNSProtocolEndpoint instance shall be associated with the CIM_DNSSettingData
697 instance through an instance of CIM_ElementSettingData.

698 2) The CIM_DNSProtocolEndpoint instance shall be associated through an instance of
699 CIM_SAPSAPDependency with the CIM_IPProtocolEndpoint instance to which the aggregating
700 CIM_IPAssignmentSettingData is being applied.

701 7.5.3 Applying static IP settings

702 When an instance of CIM_StaticIPAssignmentSettingData is applied to the CIM_IPProtocolEndpoint
703 instance, the values of the properties of the CIM_IPProtocolEndpoint instance shall be the values of the
704 properties of the CIM_StaticIPAssignmentSettingData instance.

705 7.5.3.1 CIM_StaticIPAssignmentSettingData.GatewayIPv4Address

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

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

712 7.5.3.2 CIM_StaticIPAssignmentSettingData.GatewayIPv6Address

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

716 7.5.3.3 Successful application of settings

717 An instance of CIM_StaticIPAssignmentSettingData shall be considered successfully applied when the
718 properties of the associated instance of CIM_IPProtocolEndpoint to which the instance of
719 CIM_StaticIPAssignmentSettingData has been applied have the values of the relevant properties of the
720 CIM_StaticIPAssignmentSettingData instance.

721 7.5.4 Applying DHCP settings

722 When a pending configuration includes the configuration of the DHCP client, the DHCP configuration is
723 applied as defined in the *DHCP Client Profile* ([DSP1037](#)).

724 **7.5.5 Applying DNS settings**

725 When a pending configuration includes DNS client configuration, the DNS configuration is applied as
 726 defined in the *DNS Client Profile* ([DSP1038](#)). When the AssignedSequence property of the
 727 CIM_OrderedComponent association that references an instance of CIM_DNSSettingData or
 728 CIM_DNSGeneralSettingData has a non-zero value, the referenced instance of CIM_DNSSettingData or
 729 CIM_DNSGeneralSettingData shall be applied, regardless of whether the application of a preceding
 730 CIM_SettingData instance was successful.

731 **7.5.6 Resolving overlapped settings**

732 When more than one instance of CIM_StaticIPAssignmentSettingData or CIM_DHCPSettingData is
 733 associated with the same instance of CIM_IPAssignmentSettingData, each CIM_SettingData instance
 734 shall be applied in order (as described in 7.4.3.3) until the implementation determines that the resultant
 735 configuration is valid. The amount of time an implementation waits after applying an instance of
 736 CIM_SettingData before deciding whether the resultant configuration is valid is implementation specific
 737 and outside the scope of this specification. The criterion for determining whether a configuration that is
 738 represented by a specific CIM_SettingData instance is valid is implementation specific and outside the
 739 scope of this specification.

740 **7.6 Relationship with a network interface**

741 An IP interface is generally bound to an underlying network interface. The underlying network interface
 742 might participate in a LAN and be modeled using the *Host LAN Network Port Profile* ([DSP1035](#)) or a
 743 specialization thereof. When the underlying network interface is modeled with instrumentation compliant
 744 with the *Host LAN Network Port Profile* ([DSP1035](#)), an instance of CIM_BindsToLANEndpoint shall
 745 associate the Central Instance of this profile with an instance of CIM_LANEndpoint that is compliant with
 746 the *Host LAN Network Port Profile* ([DSP1035](#)).

747 **8 Methods**

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

750 **8.1 CIM_IPProtocolEndpoint.RequestStateChange()**

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

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

756 No standard messages are defined.

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

759 **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 |

760

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 |

761 8.1.1.1 CIM_IPProtocolEndpoint.RequestStateChange() — Conditional support

762 When an instance of CIM_EnabledLogicalElementCapabilities is associated with the
763 CIM_IPProtocolEndpoint instance and the
764 CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported property contains at least one
765 value, the CIM_IPProtocolEndpoint.RequestStateChange() method shall be implemented and supported.
766 The CIM_IPProtocolEndpoint.RequestStateChange() method shall not return a value of 1 (Not
767 Supported).

768 8.2 CIM_IPConfigurationService.ApplySettingToIPProtocolEndpoint()

769 The CIM_IPConfigurationService.ApplySettingToIPProtocolEndpoint() method is used to apply a
770 configuration, as represented by an aggregating instance of CIM_IPAssignmentSettingData, to an IP
771 interface, as represented by an instance of CIM_IPProtocolEndpoint. Implementation of this method is
772 optional.

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

775 No standard messages are defined.

**776 Table 4 – CIM_IPConfigurationService.ApplySettingToIPProtocolEndpoint() Method: Return code
777 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. |

778 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 |

| Qualifiers | Name | Type | Description/Values |
|------------|------|---------------------|-------------------------|
| OUT | Job | CIM_ConcreteJob REF | Returned if job started |

779 The CIM_IPConfigurationService.ApplySettingToIPProtocolEndpoint() method shall be implemented as
 780 follows:

- 781 • The implementation shall validate that an instance of CIM_ServiceAffectsElement references
 782 the CIM_IPConfigurationService instance and the CIM_IPProtocolEndpoint instance that is
 783 identified by the Endpoint parameter to the method. If the association does not exist, the return
 784 code of the method shall be 4 (Failed).
- 785 • The implementation shall validate that an instance of CIM_ElementSettingData associates the
 786 instance of CIM_IPProtocolEndpoint that is identified by the Endpoint parameter with the
 787 instance of CIM_IPAssignmentSettingData that is identified by the Configuration parameter. If
 788 the association does not exist, the return code of the method shall be 4 (Failed).

789 When the parameters have been validated and the method is applying the settings, the method shall
 790 apply the settings as documented in 7.5 and its subclauses.

791 **8.3 Profile conventions for operations**

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

794 The default list of operations is as follows:

- 795 • GetInstance
- 796 • EnumerateInstances
- 797 • EnumerateInstanceNames
- 798 • Associators
- 799 • AssociatorNames
- 800 • References
- 801 • ReferenceNames

802 **8.4 CIM_BindsToLANEndpoint**

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

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

807 **Table 6 – Operations: CIM_BindsToLANEndpoint**

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

8.5 CIM_ElementSettingData

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

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

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 |

8.5.1 CIM_ElementSettingData — ModifyInstance

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

8.5.1.1 CIM_ElementSettingData Referencing CIM_IPAssignmentSettingData

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

- The ModifyInstance operation shall not allow the IsDefault property to be modified.
- The ModifyInstance operation shall not allow the IsCurrent property to be modified.
- When the ModifyInstance operation is used to set the IsNext property to a value of 1 (Is Next), the ModifyInstance operation shall implement the following behavior:
 - 1) The ModifyInstance operation shall find all other instances of CIM_ElementSettingData that associate an instance of CIM_IPAssignmentSettingData with the instance of CIM_IPProtocolEndpoint referenced by the target instance of CIM_ElementSettingData where the IsNext property has a value of 1 (Is Next).
 - 2) For each instance of CIM_ElementSettingData found, the ModifyInstance operation shall modify the value of its IsNext property to have a value of 2 (Is Not Next).
- When the IsNext property has a value of 1 (Is Next), the ModifyInstance operation shall not be supported.
- When the ModifyInstance operation is used to set the IsNext property to a value of 3 (Is Next for Single Use), the ModifyInstance operation shall implement the following behavior:
 - 1) The ModifyInstance operation shall find all other instances of CIM_ElementSettingData that associate an instance of CIM_IPAssignmentSettingData with the instance of CIM_IPProtocolEndpoint referenced by the target instance of CIM_ElementSettingData where the IsNext property has a value of 3 (Is Next For Single Use).
 - 2) For each instance of CIM_ElementSettingData found, the ModifyInstance operation shall modify the value of its IsNext property to have a value of 2 (Is Not Next).

841 **8.5.1.2 CIM_ElementSettingData Referencing CIM_StaticIPAssignmentSettingData**

842 When an instance of CIM_ElementSettingData associates an instance of
 843 CIM_StaticIPAssignmentSettingData with an instance of CIM_IPProtocolEndpoint, the ModifyInstance
 844 operation shall not be supported.

845 **8.6 CIM_HostedAccessPoint**

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

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

850 **Table 8 – Operations: CIM_HostedAccessPoint**

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

851 **8.7 CIM_HostedService**

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

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

856

857 **Table 9 – Operations: CIM_HostedService**

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

858 **8.8 CIM_IPAssignmentSettingData**

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

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

861 **8.9 CIM_IPConfigurationService**

862 All operations 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 **8.10 CIM_IPProtocolEndpoint**

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

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

869 **Table 10 – Operations: CIM_IPProtocolEndpoint**

| Operation | Requirement | Messages |
|----------------|--------------------------|----------|
| ModifyInstance | Conditional. See 8.10.1. | None |

870 **8.10.1 CIM_IPProtocolEndpoint — ModifyInstance operation**

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

873 **8.10.1.1 CIM_IPProtocolEndpoint.ElementName Property**

874 When an instance of CIM_EnabledLogicalElementCapabilities is associated with the
 875 CIM_IPProtocolEndpoint instance and the
 876 CIM_EnabledLogicalElementCapabilities.ElementNameEditSupported property has a value of TRUE, the
 877 implementation shall allow the ModifyInstance operation to change the value of the ElementName
 878 property of the CIM_IPProtocolEndpoint instance. The ModifyInstance operation shall enforce the length
 879 restriction specified in the MaxElementNameLen property of the instance of
 880 CIM_EnabledLogicalElementCapabilities.

881 When no instance of CIM_EnabledLogicalElementCapabilities is associated with the
 882 CIM_IPProtocolEndpoint instance, or the ElementNameEditSupported property of the
 883 CIM_EnabledLogicalElementCapabilities instance has a value of FALSE, the implementation shall not
 884 allow the ModifyInstance operation to change the value of the ElementName property of the
 885 CIM_IPProtocolEndpoint instance.

886 **8.11 CIM_OrderedComponent**

887 Table 11 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 11, 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 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 |

892 **8.11.1 CIM_OrderedComponent — ModifyInstance**

893 The ModifyInstance operation may be supported for CIM_OrderedComponent. When an instance of
 894 CIM_OrderedComponent references an instance of CIM_DNSSettingData or an instance of
 895 CIM_DNSGeneralSettingData, the AssignedSequence property may be modified. When an instance of
 896 CIM_OrderedComponent references an instance of CIM_StaticIPAssignmentSettingData or an instance
 897 of CIM_DHCPSettingData, the AssignedSequence property shall not be modified.

898 **8.12 CIM_RemoteAccessAvailableToElement**

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

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

903 **Table 12 – Operations: CIM_RemoteAccessAvailableToElement**

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

904 **8.13 CIM_RemoteServiceAccessPoint**

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

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

907 **8.14 CIM_ServiceAffectsElement**

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

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

912 **Table 13 – Operations: CIM_ServiceAffectsElement**

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

913 **8.15 CIM_StaticIPAssignmentSettingData**

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

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

918

Table 14 – Operations: CIM_StaticIPAssignmentSettingData

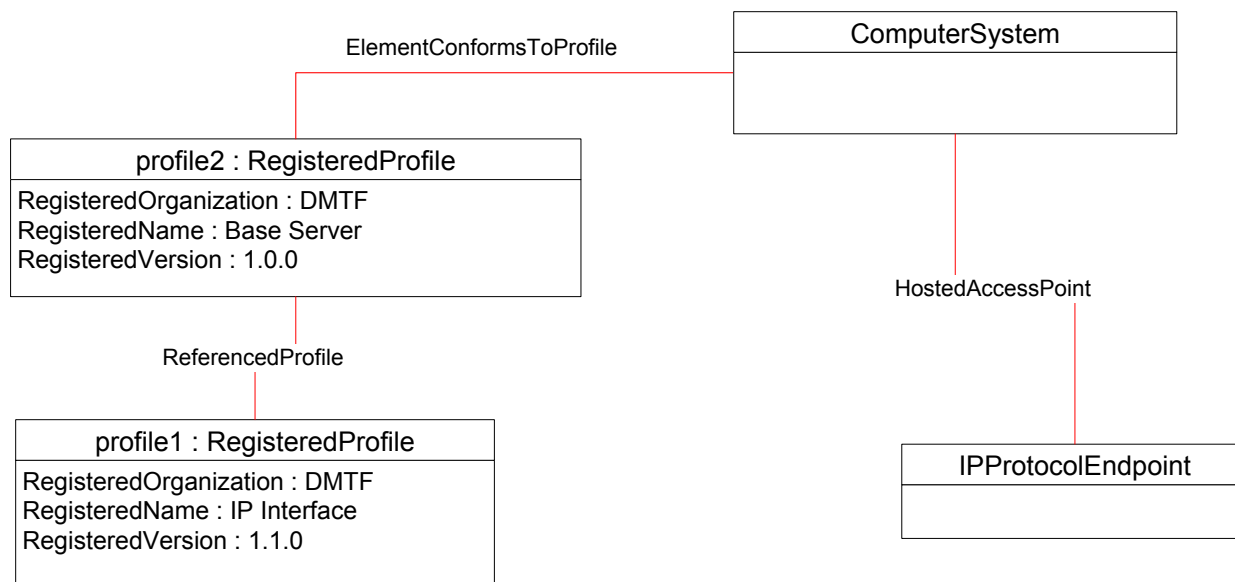
| Operation | Requirement | Messages |
|----------------|-------------|----------|
| ModifyInstance | Optional | None |

920 **9 Use cases**

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

922 **9.1 Miscellaneous object diagrams**

923 The object diagram in Figure 2 shows one possible method for advertising profile conformance. The
 924 instances of CIM_RegisteredProfile are used to identify the version of the *IP Interface Profile* with which
 925 an instance of CIM_IPProtocolEndpoint and its associated instances are conformant. An instance of
 926 CIM_RegisteredProfile exists for each profile that is instrumented in the system. One instance of
 927 CIM_RegisteredProfile identifies the “DMTF Base Server Profile version 1.0.0”. The other instance
 928 identifies the “DMTF IP Interface Profile version 1.1.0”. The CIM_IPProtocolEndpoint instance is scoped
 929 to an instance of CIM_ComputerSystem. This instance of CIM_ComputerSystem is conformant with the
 930 DMTF [Base Server Profile](#) version 1.0.0 as indicated by the CIM_ElementConformsToProfile association
 931 to the CIM_RegisteredProfile instance.



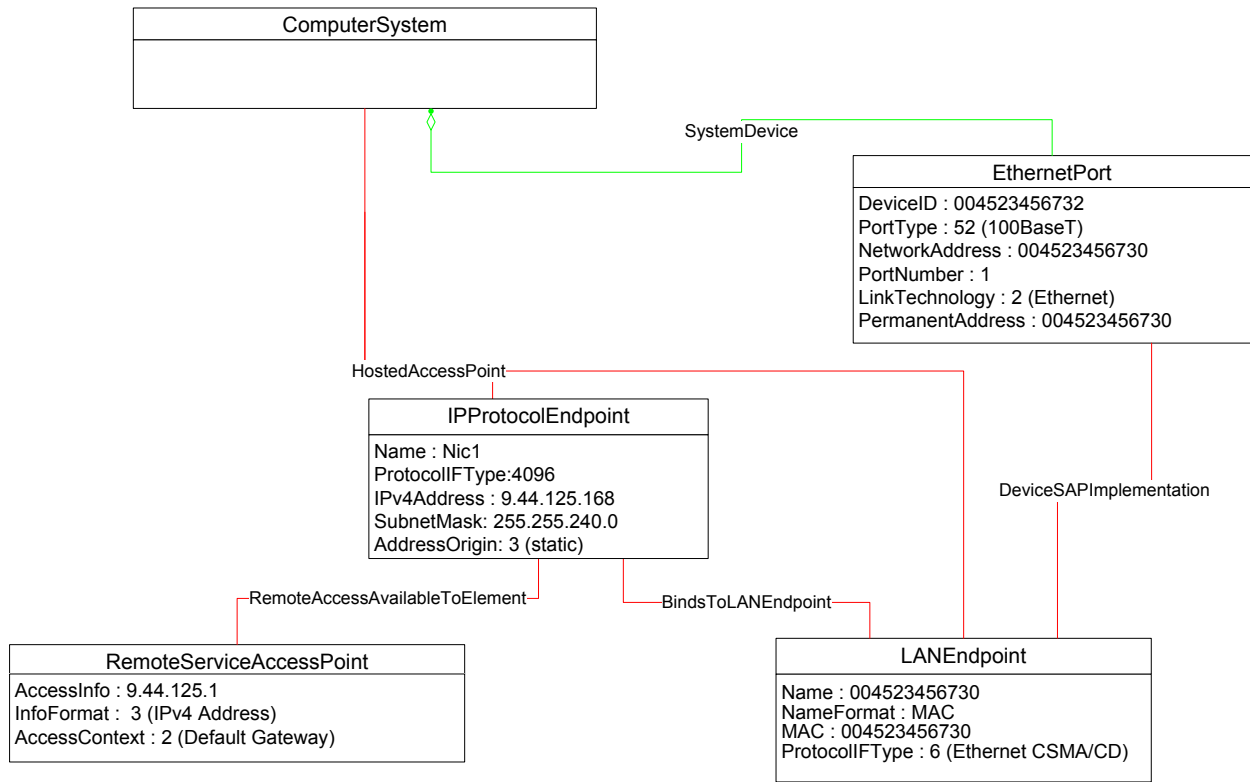
932

933 **Figure 2 – Registered profile**

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

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

944



945

946

Figure 3 – Basic configuration — IPv4

947

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.

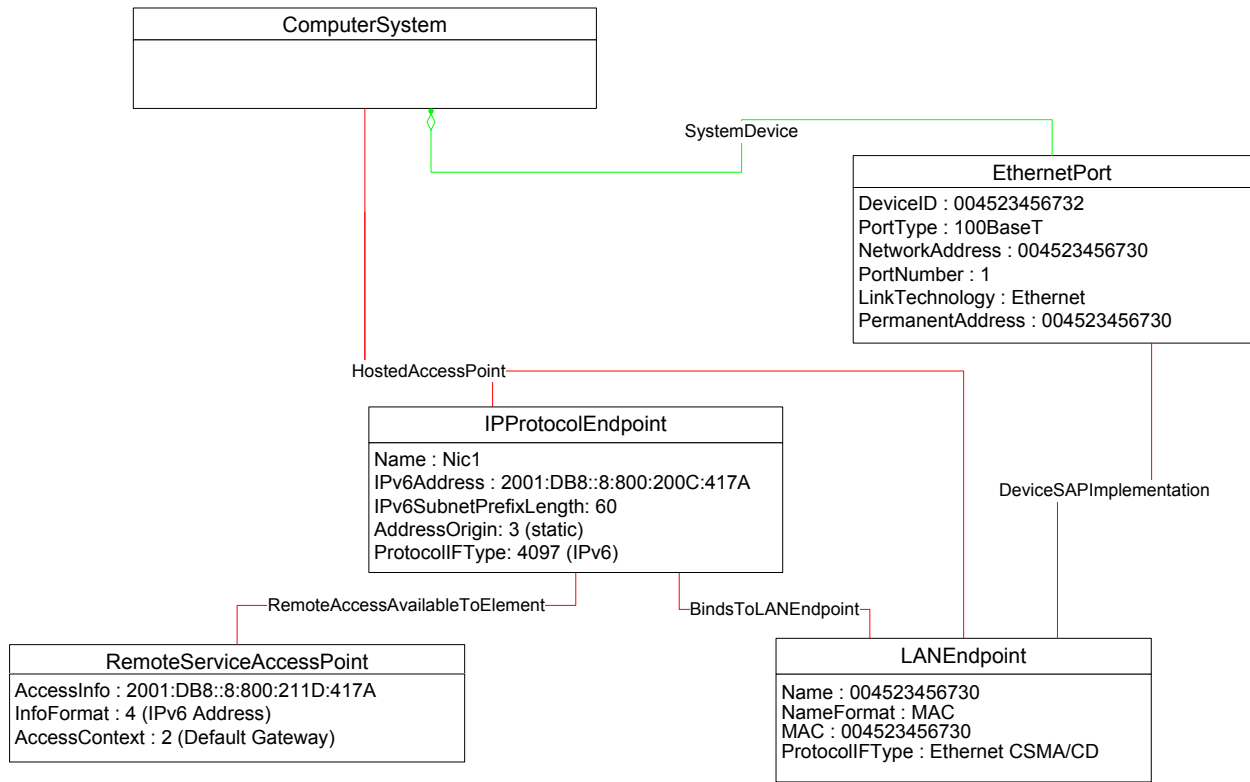
948

949

950

951

952



953

954

Figure 4 – Basic configuration — IPv6

955

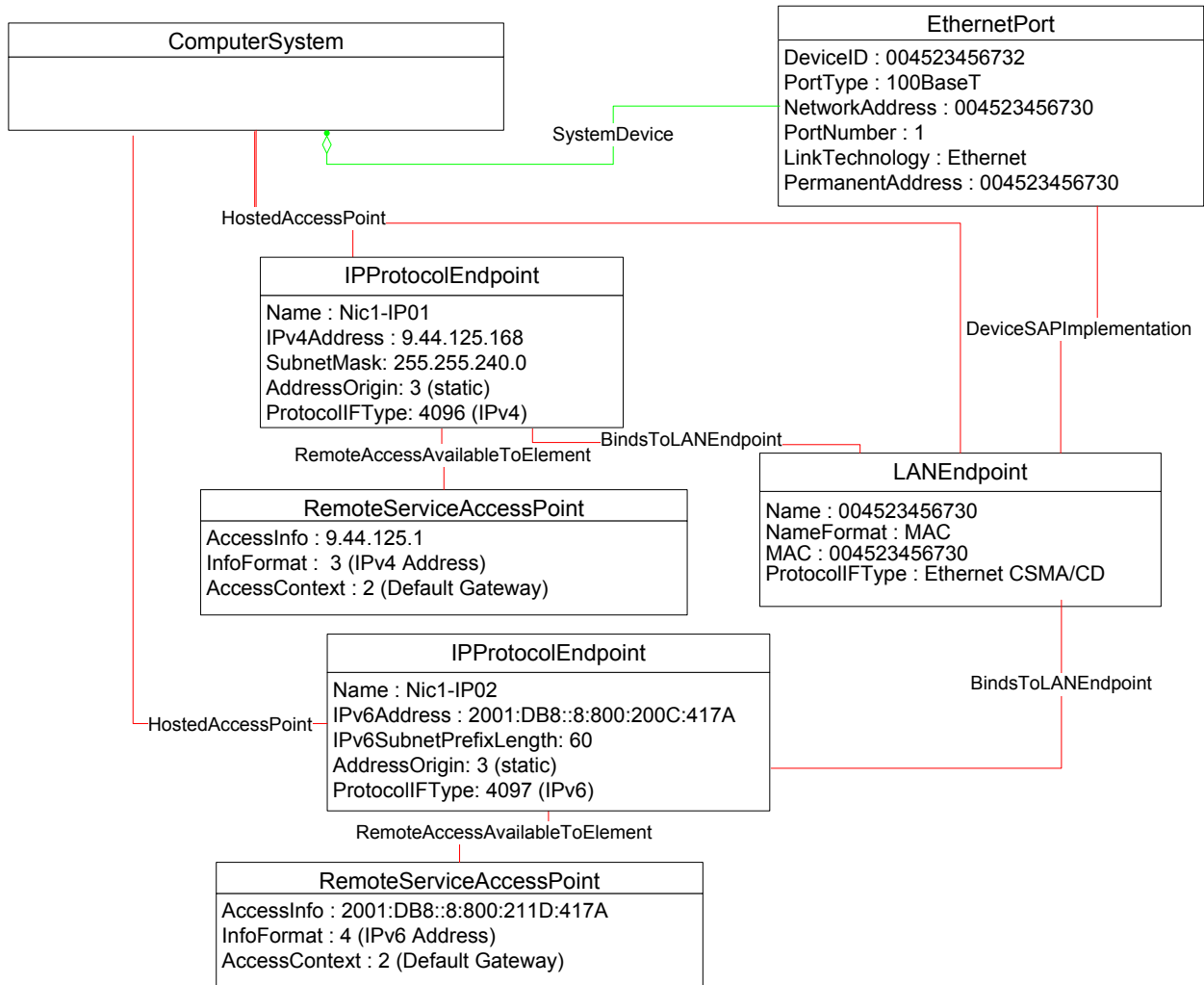
The object diagram shown in Figure 5 contains the basic elements used to model the current configuration of two IP interfaces on a single EthernetPort — one that has an IPv4 address and one that has an IPv6 address. In this diagram, the [Ethernet Port Profile](#) and *IP Interface Profile* have been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

956

957

958

959

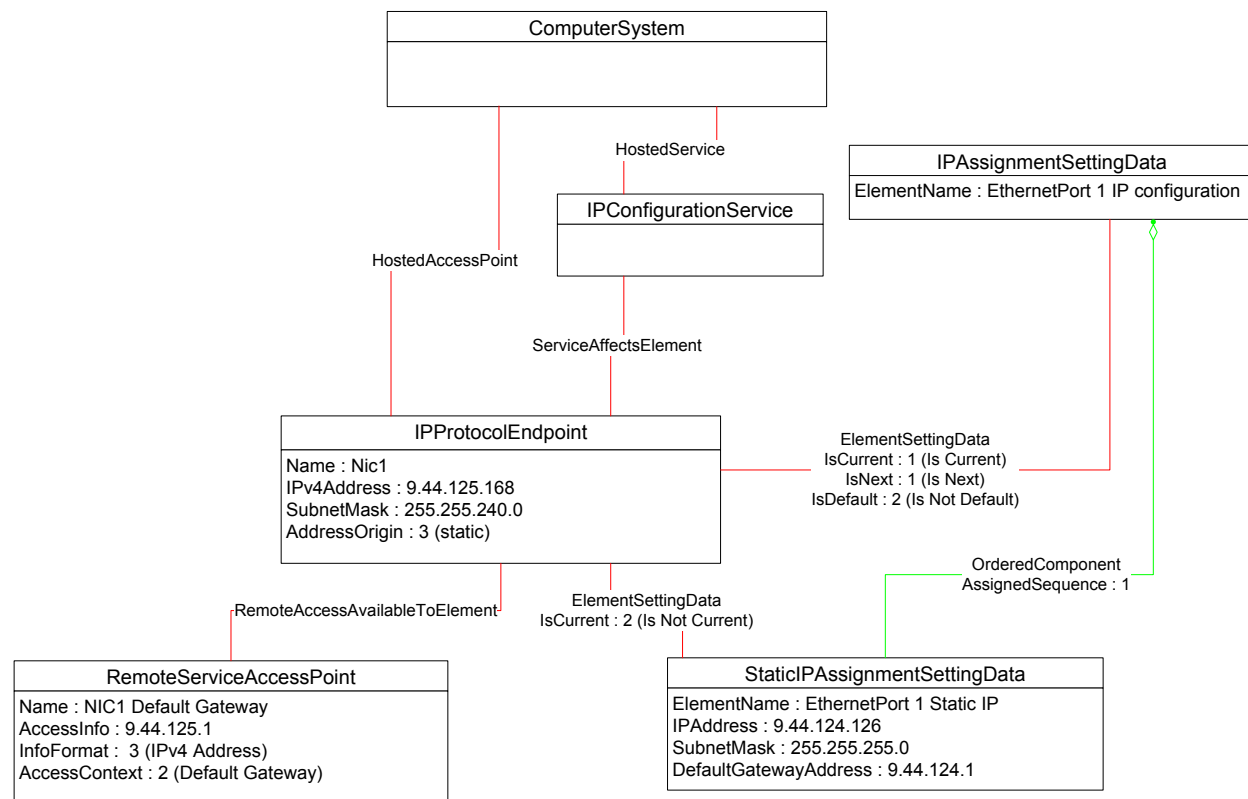


960
961

962

Figure 5 – Basic configuration — IPv4 and IPv6

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



969

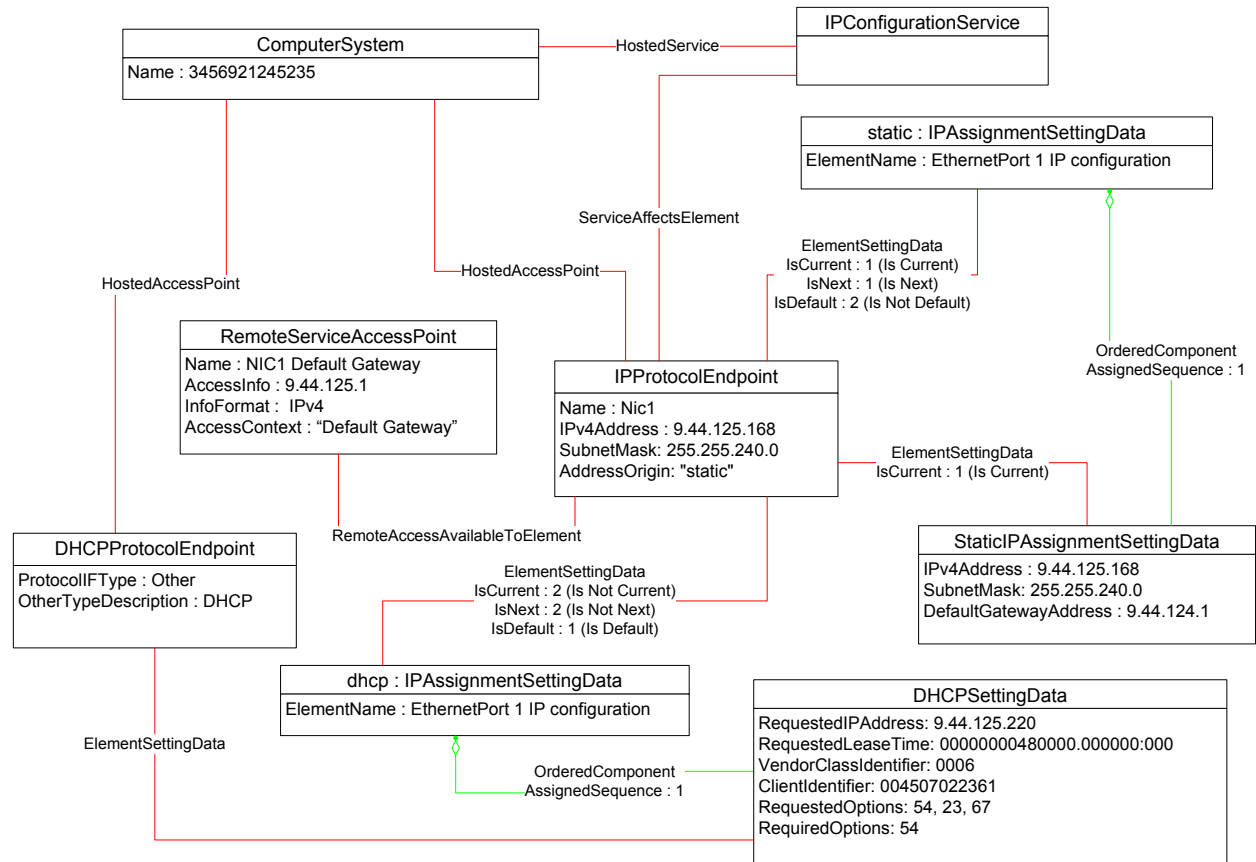
970

Figure 6 – Static current and pending configuration

971 The object diagram in Figure 7 illustrates an IP interface with two supported alternate configurations. Two
 972 discrete IP configuration options are available for the IP interface. Each option is represented by an
 973 instance of CIM_IPAssignmentSettingData. One configuration option represents the ability to statically
 974 assign the IP configuration. This option is indicated by the instance of CIM_OrderedComponent that
 975 associates the CIM_IPAssignmentSettingData instance with an instance of
 976 CIM_StaticIPAssignmentSettingData. The other configuration option is to obtain the configuration through
 977 a DHCP client. This option is indicated by the instance of CIM_OrderedComponent that associates the
 978 CIM_IPAssignmentSettingData with an instance of CIM_DHCPSettingData.

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

982 The default configuration is to attempt to obtain a configuration through DHCP. This default is indicated
 983 by the IsDefault property having a value of 1 (Is Default) on the CIM_ElementSettingData instance that
 984 associates the CIM_IPAssignmentSettingData instance with the CIM_IPProtocolEndpoint instance.
 985 However, the current configuration of the IP interface was statically assigned using the configuration
 986 identified by the CIM_IPAssignmentSettingData instance *static*. This configuration is indicated by the
 987 value of the IsCurrent property on the instance of CIM_ElementSettingData that associates the
 988 CIM_IPAssignmentSettingData instance *static* with the CIM_IPProtocolEndpoint instance, and by the
 989 value of the AddressOrigin property on the CIM_IPProtocolEndpoint instance. When the interface is
 990 restarted, the static configuration will be used again for the IP interface. This behavior is indicated by the
 991 value of the IsNext property on the instance of CIM_ElementSettingData that associates the
 992 CIM_IPAssignmentSettingData instance *static* to the CIM_IPProtocolEndpoint instance.



993

994

Figure 7 – Static and DHCP pending configurations

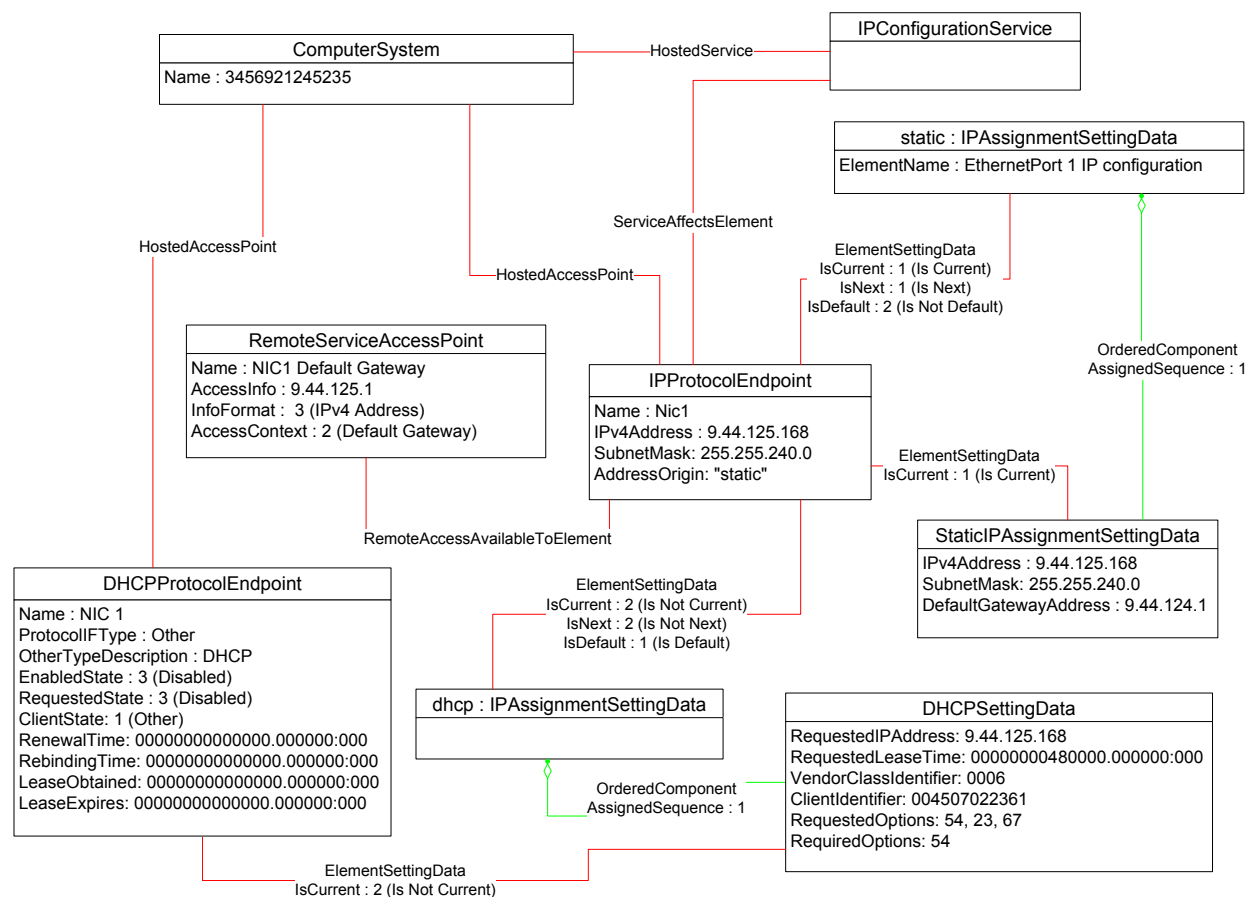
995 The object diagram in Figure 8 provides an example of an IP interface that was configured to default to a
 996 statically assigned IP configuration if the DHCP client failed to obtain a configuration from a DHCP server.

997 The EnabledState and ClientState properties of the CIM_DHCPProtocolEndpoint instance indicate that
 998 the DHCP client is not disabled but neither is it actively attempting to obtain a configuration any longer.

999 No instance of CIM_RemoteServiceAccessPoint is associated with the CIM_DHCPProtocolEndpoint
 1000 instance because the DHCP client failed to communicate with a DHCP server.

1001 The AddressOrigin property of the CIM_IPProtocolEndpoint instance reflects that the address was
 1002 assigned statically.

1003

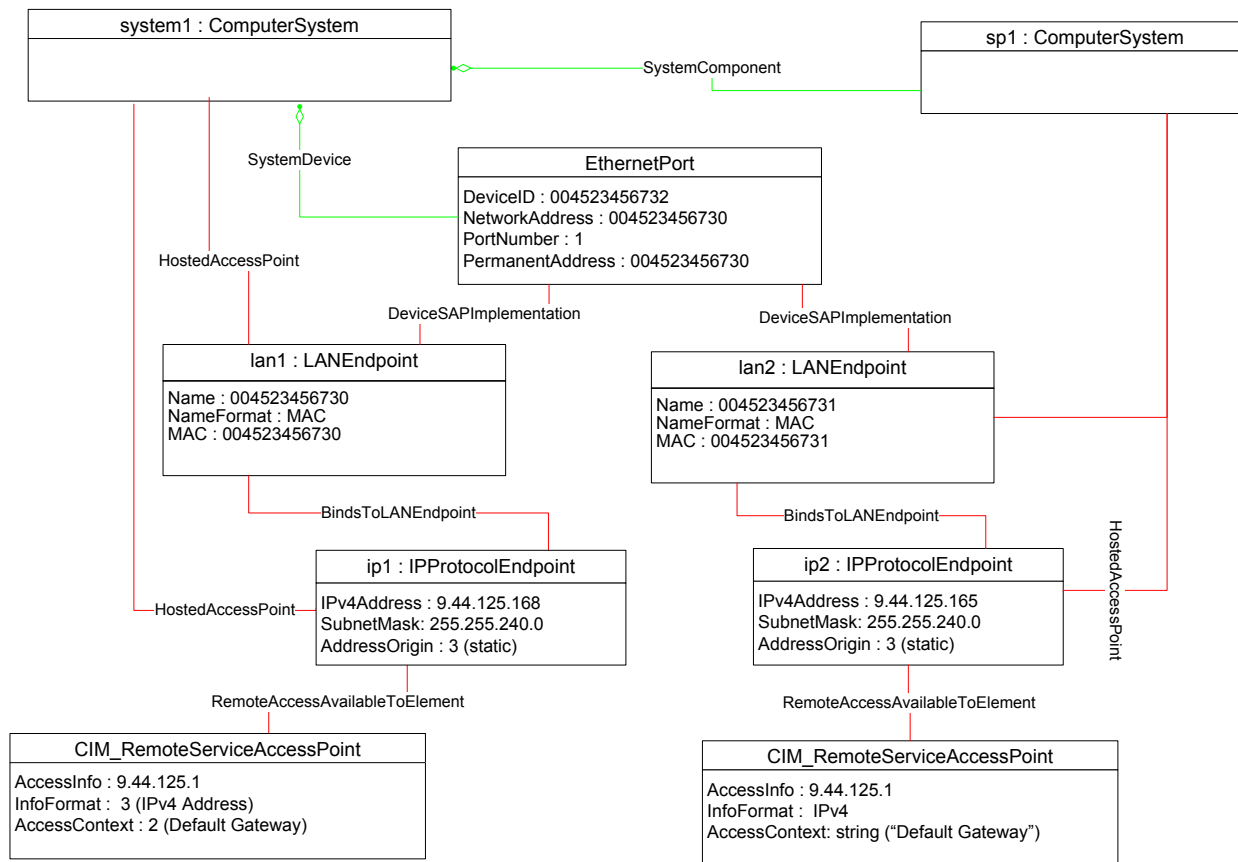


1004

1005

Figure 8 – DHCP timed out to a static configuration

1006 The object diagram in Figure 9 illustrates a configuration in which a system contains an integrated service
 1007 processor and they share the network interface of the system. The CIM_EthernetPort instance is
 1008 associated with the system1 instance, which indicates that the network device is owned by the server.
 1009 The MAC property of the lan1 instance matches the PermanentAddress property of the CIM_EthernetPort
 1010 instance, which indicates that the server is using the hardware MAC. The MAC property of the lan2
 1011 instance is different, which indicates that the service processor has been assigned a logical MAC. The
 1012 system and service processor each have a unique IP interface that has been statically configured.



1013

1014

Figure 9 – Service processor and server share an NIC

1015 The object diagrams in Figure 10 through Figure 14 show different aspects of a single system. The
 1016 system has support for the DNS and DHCP clients. For configurations using DHCP, the DNS
 1017 configuration can be statically assigned or partially assigned through DHCP. The system itself does not
 1018 support the persistence of alternate configurations. Rather the instrumentation layer presents the different
 1019 configuration possibilities as distinct alternate configurations.

1020 Note that in the following figures extraneous classes that are not relevant to the point being illustrated are
 1021 not shown. For example, the CIM_HostedAccessPoint associations are never included.

1022 The object diagram in Figure 10 outlines the alternate configurations presented by the instrumentation
 1023 layer for the system. Three alternate configurations are shown: static_only, dhcp_only, and dhcp_static.

1024 The system persists a single underlying static IP configuration, which is represented by static1. When the
 1025 configuration selected is static only or DHCP and then static, the same client static IP configuration is
 1026 used.

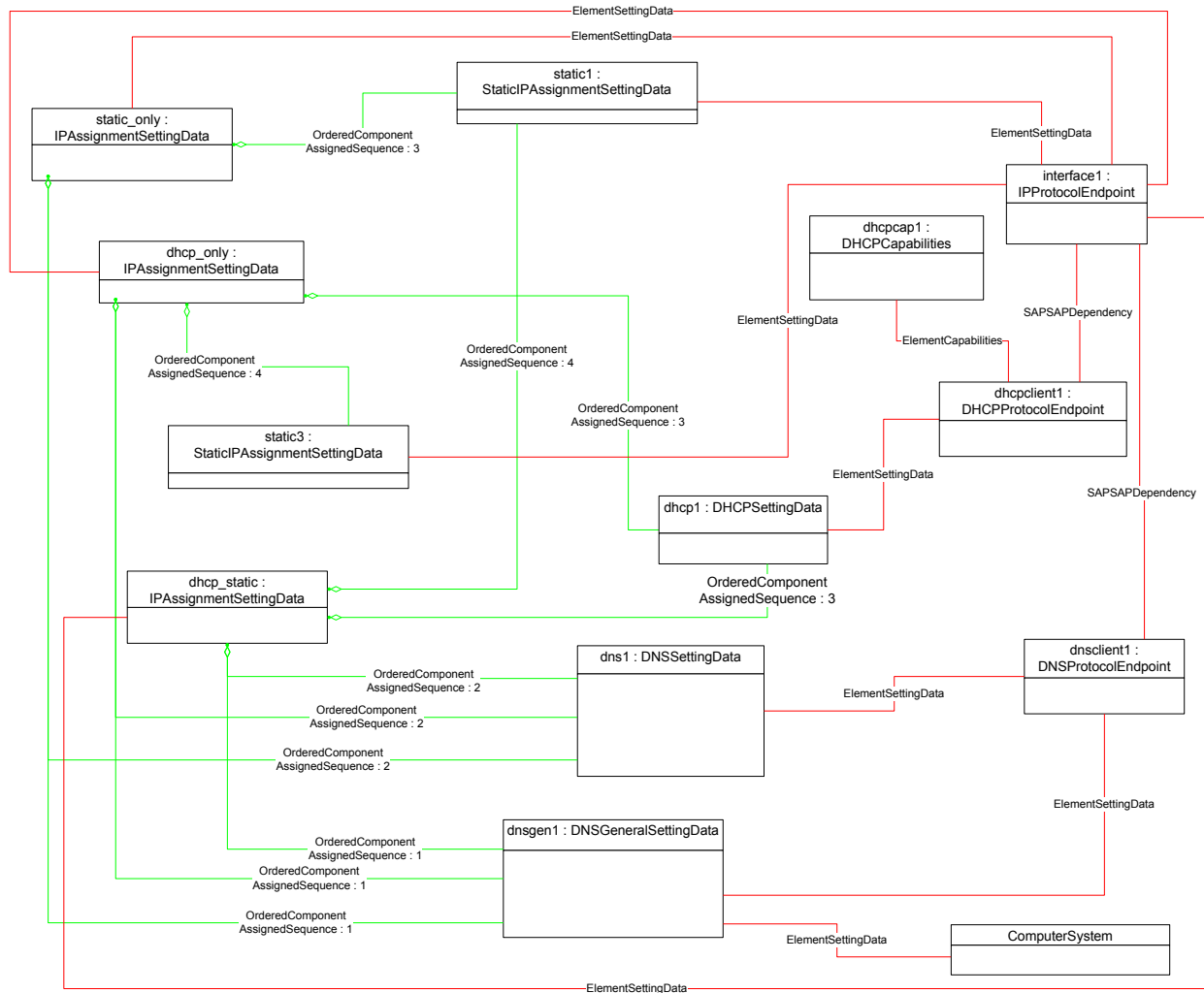
1027 The system persists a single underlying DNS configuration represented by dns1 and dnsgen1.

1028 static_only represents a configuration that uses static assignment of the IP configuration, including
 1029 support for static configuration of the DNS client. This behavior is indicated by the aggregated instances:
 1030 static1, dns1, and dnsgen1.

1031 dhcp_only represents a configuration that uses DHCP to obtain the IP configuration. This behavior is
 1032 indicated by the aggregated instance dhcp1. The DNS configuration can be assigned through DHCP or
 1033 statically assigned. This behavior is indicated by the aggregated instances dns1 and dnsgen1. In the
 1034 event the DHCP client is unable to obtain a configuration, the system is implemented to default to a hard-

1035 coded, well-known default static IP configuration. The existence of a default configuration is indicated by
 1036 the aggregated instance static3. Note that no advertisement mechanism is specified in the profile to
 1037 indicate that static3 represents hard-coded values that cannot be modified by the client. If the system
 1038 were implemented such that the DHCP client would be continually in use without a timeout to a static
 1039 configuration, the aggregated instance static3 would not exist.

1040 dhcp_static represents a configuration that attempts to use DHCP to obtain an IP configuration. In the
 1041 event the DHCP client fails to obtain a configuration, the system defaults to a client-assigned static IP
 1042 configuration. This behavior is indicated by the instances dhcp1 and static1 and the relative values of the
 1043 AssignedSequence property of the instances of CIM_OrderedComponent, which aggregate them into
 1044 dhcp_static.



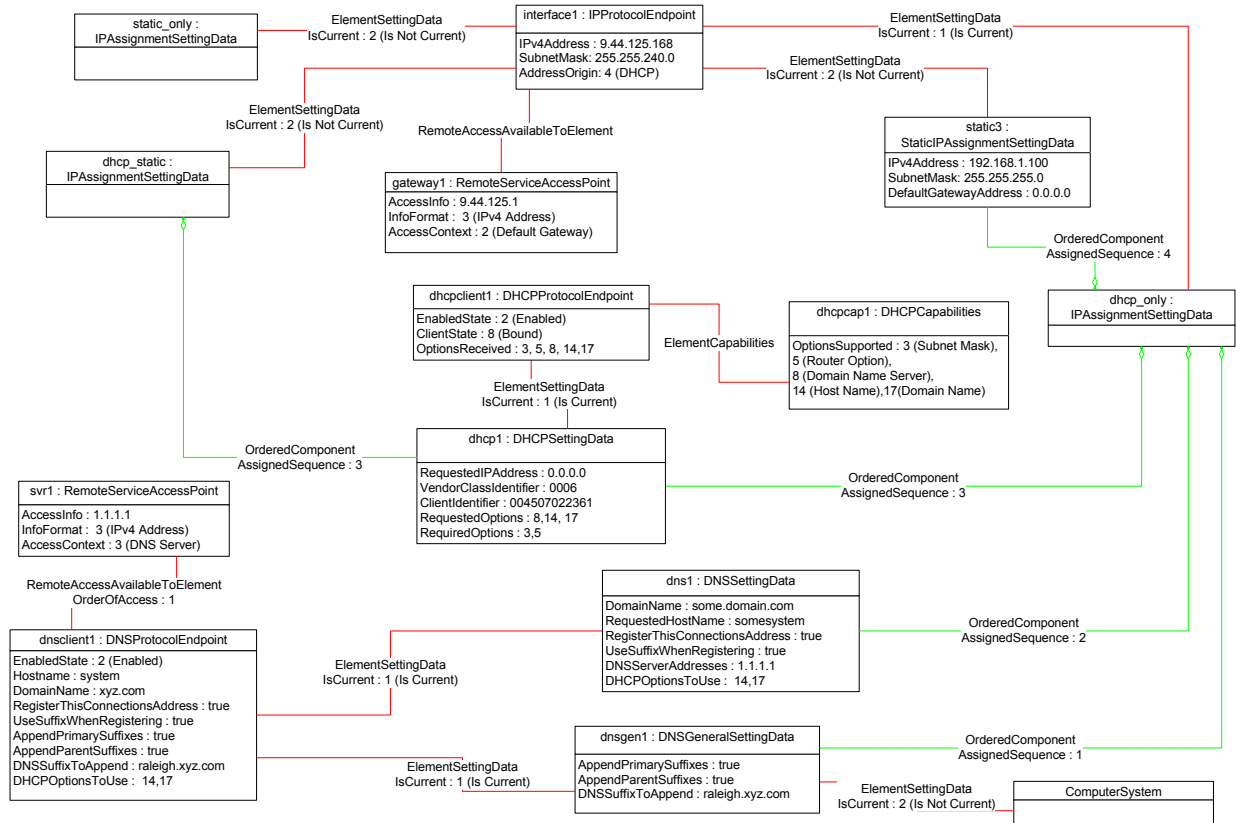
1045

1046

Figure 10 – Configuration choices

1047 The object diagram in Figure 11 reflects the system when the DHCP configuration method has been used
 1048 and the DNS configuration has partially been assigned through DHCP and partially statically configured.

1049 The use of the DHCP-only configuration is indicated by the IsCurrent property of the instance of
 1050 CIM_ElementSettingData that associates dhcp_only to interface1 having the value 1 (Is Current). The
 1051 DHCP configuration includes DHCP options that affect the DNS configuration. The DHCP options 8, 14,
 1052 and 17 are requested as indicated by the RequestedOptions property of dhcp1. Each of these options
 1053 was in turn received by the DHCP client, which is indicated by the value of the OptionsReceived property
 1054 of dhcpclient1. The DNS client has been configured to use the values received for options 14 and 17 as
 1055 indicated by the presence of these values in the DHCPOptionsToUse property of dnsclient1. The
 1056 properties on dnsclient1 reflect the current DNS client configuration. Note that the actual current
 1057 configuration does not directly reflect the configuration indicated by dns1 and dnsngen1. The two
 1058 properties for which values were supplied by the DHCP options instead reflect the values assigned by the
 1059 DHCP server.



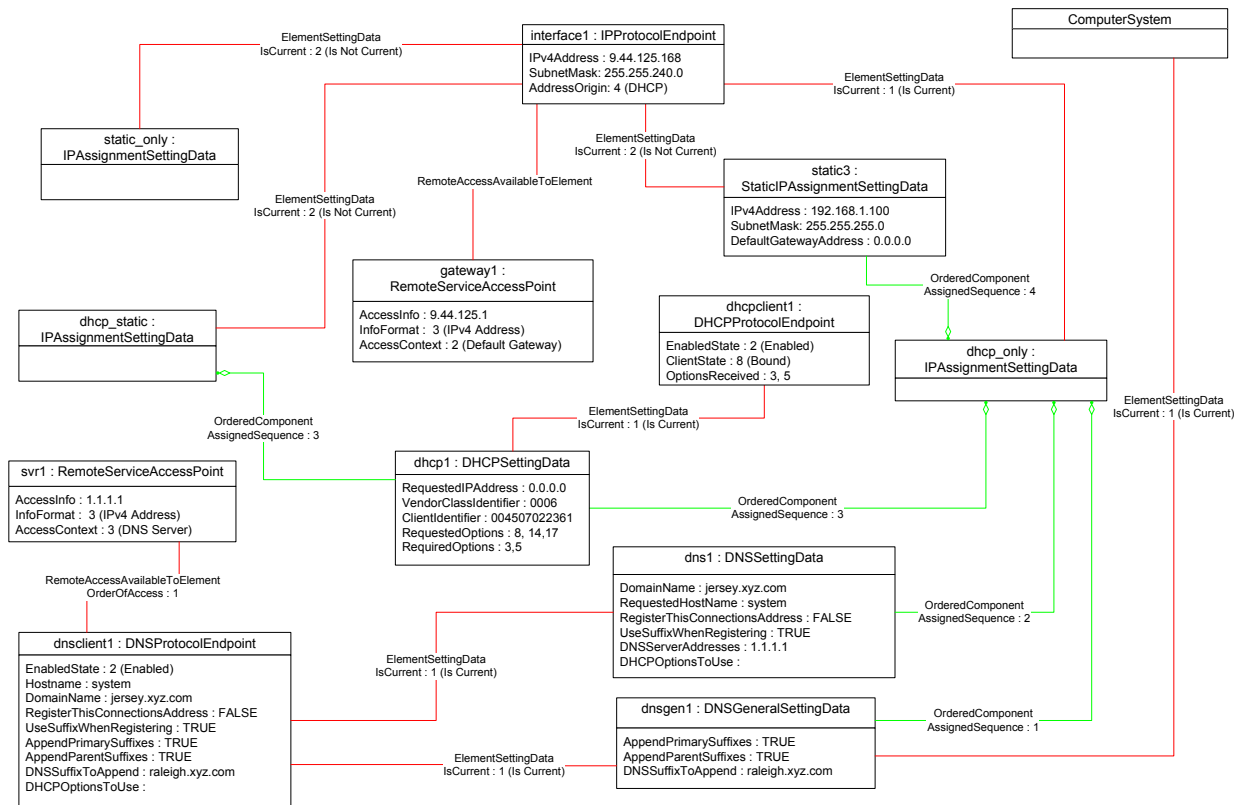
1060

1061

Figure 11 – DHCP assigned partial DNS

1062 The object diagram in Figure 12 reflects the system when the DHCP configuration method has been used
 1063 and the DNS configuration has been statically configured.

1064 The use of the DHCP-only configuration is indicated by the IsCurrent property of the instance of
 1065 CIM_ElementSettingData that associates dhcp_only to interface1 having the value 1 (Is Current).
 1066 Although the DHCP configuration includes DHCP options that affect the DNS configuration, the values
 1067 returned are not being used by the DNS client. This behavior is indicated by the absence of any values in
 1068 the DHCPOptionsToUse property of dnsclient1. The actual current configuration directly reflects the
 1069 configuration indicated by dns1 and dnsngen1 because no DHCP options are selected for use.



1070

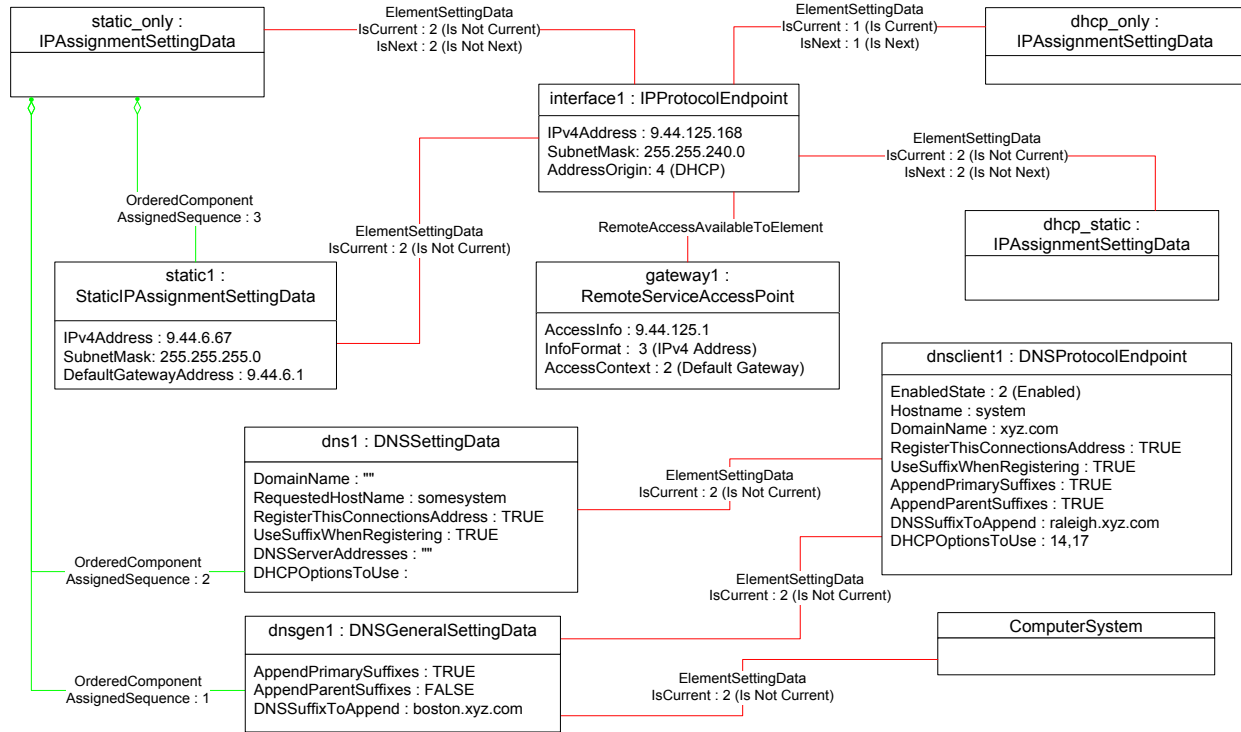
1071

Figure 12 – DHCP with DNS statically configured

1072 **9.1.1 Sequence for disabled DNS client**

1073 The following three object diagrams illustrate the system when a client is configuring it to use a static IP
 1074 configuration with the DNS client disabled. The client first modifies the pending static configuration so that
 1075 the DNS settings will not be applied. Then it disables the DNS client directly. Finally, it applies the static
 1076 configuration.

1077 The object diagram in Figure 13 illustrates the state of the system before the client begins modifying it to
 1078 use a static IP configuration with DNS disabled. The last configuration applied was the DHCP-only
 1079 configuration, which is indicated by the value of the IsCurrent property of the CIM_ElementSettingData
 1080 instance that references dhcp_only and interface1. The static_only configuration has not yet been
 1081 modified by the client. As shown, the alternate DNS configuration represented by dns1 and dnsgen1
 1082 would be applied if static_only were applied to interface1.



1083

1084

Figure 13 – Static without DNS configuration — One

1085

1086

1087

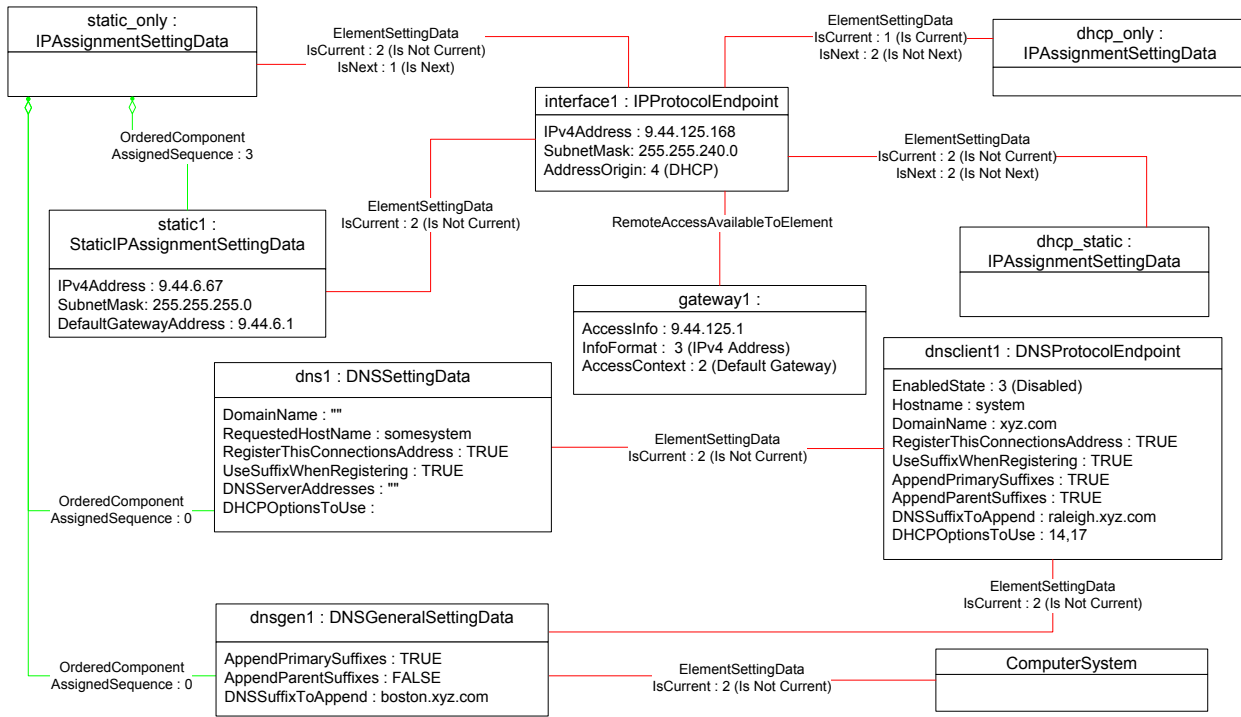
1088

1089

1090

1091

In Figure 14, static_only has been made the pending configuration for interface1. This behavior is indicated by the value of the IsNext property of the instance of CIM_ElementSettingData that references static_only and interface1. static_only has been modified such that the DNS configuration will not be applied. This behavior is indicated by the AssignedSequence property having a value of 0 (zero) for each of the CIM_OrderedComponent instances that reference static_only and dns1 or dnsgen1. Separately, the DNS client has been disabled, which is indicated by the value of the EnabledState property of dnscient1.



1092

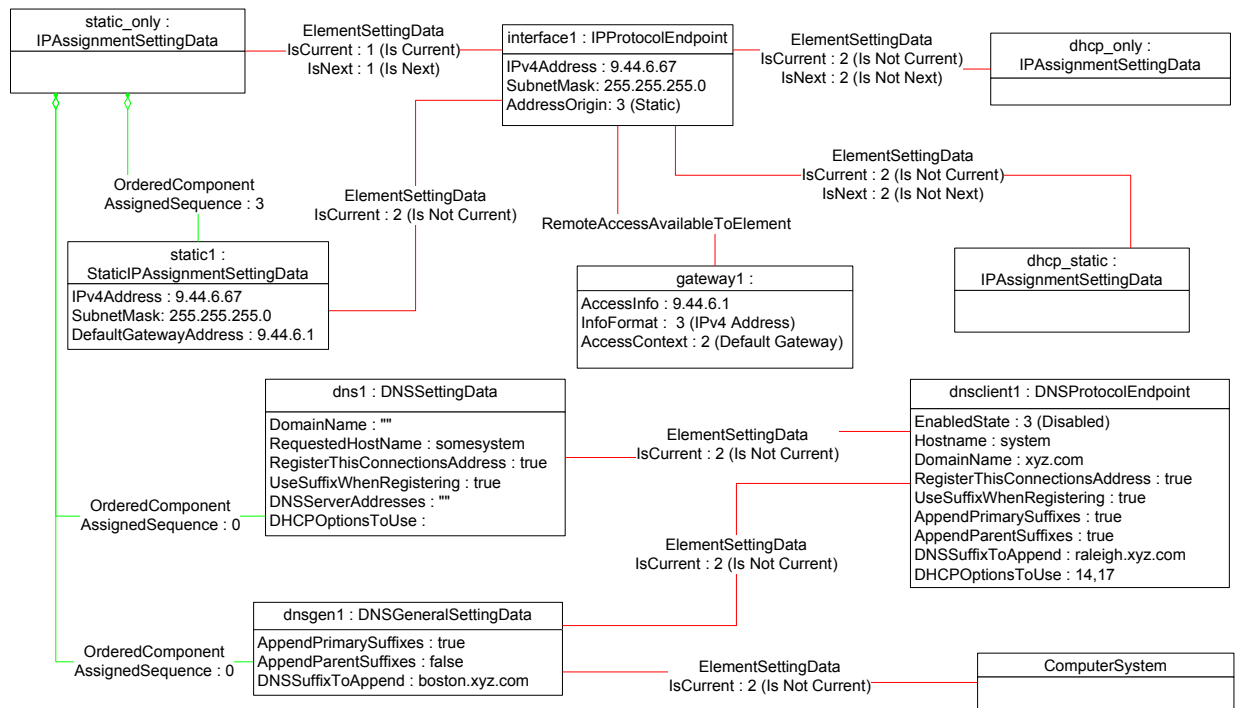
1093

Figure 14 – Static without DNS configuration — Two

1094

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

1095



1096

1097

Figure 15 – Static without DNS configuration — Three

1098 9.2 Determine supported configuration methods

1099 A client can determine which configuration methods are supported for a given interface as follows:

- 1100 1) Find all instances of CIM_IPAssignmentSettingData that are associated with the
1101 CIM_IPProtocolEndpoint instance.
- 1102 2) For each instance of CIM_IPAssignmentSettingData:
 - 1103 a) Find all instances of subclasses of CIM_IPAssignmentSettingData that are associated with
1104 the CIM_IPAssignmentSettingData instance through an instance of
1105 CIM_OrderedComponent.
 - 1106 b) Query the value of the AddressOrigin property to determine the supported identified
1107 configuration method.

1108 9.3 Determine gateway address

1109 A client can find the default gateway in use for an IP interface as follows:

- 1110 1) Find all instances of CIM_RemoteServiceAccessPoint that are associated with the
1111 CIM_IPProtocolEndpoint instance through an instance of
1112 CIM_RemoteAccessAvailableToElement.
- 1113 2) For each instance of CIM_RemoteServiceAccessPoint, determine if the value of the
1114 AccessContext property is "Default Gateway". If so, query the value of the AccessInfo property.

1115 9.4 Determine method used for current configuration

1116 A client can determine the method by which the IP configuration was assigned by querying the
1117 AddressOrigin property of the CIM_IPProtocolEndpoint instance.

1118 9.5 Determine whether DHCP then static is supported

1119 An implementation may support attempting to acquire its IP configuration through a DHCP client and
1120 defaulting to a static configuration if the client fails to acquire a configuration from a DHCP server. A client
1121 can determine whether this functionality is supported as follows:

- 1122 1) Find all instances of CIM_IPAssignmentSettingData (the parent class and not subclasses) that
1123 are associated with the CIM_IPProtocolEndpoint instance.
- 1124 2) For each instance of CIM_IPAssignmentSettingData:
 - 1125 a) Find all instances of CIM_DHCPSettingData that are associated through an instance of
1126 CIM_OrderedComponent.
 - 1127 b) Find all instances of CIM_StaticIPAssignmentSettingData that are associated through an
1128 instance of CIM_OrderedComponent.
- 1129 3) Determine if there is an instance of CIM_DHCPSettingData such that the value of the
1130 AssignedSequence property of the CIM_OrderedComponent that associates the instance of
1131 CIM_DHCPSettingData with the instance of CIM_IPAssignmentSettingData is less than the
1132 value of the AssignedSequence property of an instance of CIM_OrderedComponent that
1133 associates the CIM_StaticIPAssignmentSettingData with the instance of
1134 CIM_IPAssignmentSettingData. If so, DHCP then static is supported.

1135 9.6 View default configuration

1136 A client can view the default configuration for an IP interface as follows:

- 1137 1) Find all instances of CIM_ElementSettingData that associate an instance of
1138 CIM_IPAssignmentSettingData (the parent class and not subclasses) with the
1139 CIM_IPProtocolEndpoint instance.

- 1140 2) For each instance of CIM_ElementSettingData, see if the value of the IsDefault property is 1 (Is
1141 Default).

1142 9.7 Configure the interface to use DHCP

1143 An implementation may support attempting to acquire its IP configuration through a DHCP client. A client
1144 can determine whether this functionality is supported and configure the interface to use it as follows:

- 1145 1) Find all instances of CIM_IPAssignmentSettingData (the parent class and not subclasses) that
1146 are associated with the CIM_IPProtocolEndpoint instance.
- 1147 2) For each instance of CIM_IPAssignmentSettingData:
- 1148 a) Find an instance of CIM_DHCPSettingData that is associated through an instance of
1149 CIM_OrderedComponent.
- 1150 b) Verify that no instances of CIM_StaticIPAssignmentSettingData are associated with the
1151 instance of CIM_IPAssignmentSettingData.
- 1152 This instance of CIM_IPAssignmentSettingData represents a DHCP configuration.
- 1153 3) Find an instance of CIM_IPConfigurationService that is associated with the
1154 CIM_IPProtocolEndpoint instance through an instance of CIM_ServiceAffectsElement.
- 1155 4) Invoke the ApplySettingToIPProtocolEndpoint() method of the CIM_IPConfigurationService
1156 instance, specifying the instances of CIM_IPProtocolEndpoint and
1157 CIM_IPAssignmentSettingData.

1158 9.8 Establish a static IP configuration for an interface

1159 A client can manually assign an IP configuration to an interface as follows:

- 1160 1) Find all instances of CIM_IPAssignmentSettingData (the parent class and not subclasses) that
1161 are associated with the CIM_IPProtocolEndpoint instance.
- 1162 2) For each instance of CIM_IPAssignmentSettingData:
- 1163 a) Find an instance of CIM_StaticIPAssignmentSettingData that is associated through an
1164 instance of CIM_OrderedComponent.
- 1165 b) Verify that no other instances of CIM_StaticIPAssignmentSettingData or instances of
1166 CIM_DHCPSettingData are associated with the instance of CIM_IPAssignmentSettingData
1167 through an instance of CIM_OrderedComponent.
- 1168 c) For the instance of CIM_ElementSettingData that associates the
1169 CIM_IPAssignmentSettingData instance with the instance of CIM_IPProtocolEndpoint, verify
1170 that the value of the IsDefault property is 2 (Is Not Default).
- 1171 This instance of CIM_IPAssignmentSettingData represents a modifiable, static configuration for
1172 the IP interface.
- 1173 3) Modify the properties of the CIM_StaticIPAssignmentSettingData instance to contain the
1174 appropriate configuration for the IP interface.
- 1175 4) Apply the pending configuration using the steps in 9.9 or 9.10.

1176 9.9 Apply a pending configuration — Synchronously

1177 Some implementations may support modifying the configuration of an IP interface without requiring a
1178 restart of the underlying network interface. If this behavior is supported by the implementation, then given
1179 an instance of CIM_IPProtocolEndpoint for which the configuration should be modified and an instance of
1180 CIM_IPAssignmentSettingData that represents the new configuration, a client can:

- 1181 1) Find an instance of CIM_IPConfigurationService that is associated with the
1182 CIM_IPProtocolEndpoint instance through an instance of CIM_ServiceAffectsElement.
- 1183 2) Invoke the ApplySettingToIPProtocolEndpoint() method of the CIM_IPConfigurationService,
1184 specifying the instances of CIM_IPProtocolEndpoint and CIM_IPAssignmentSettingData.

1185 **9.10 Apply a pending configuration — Upon restart**

1186 Some implementations may require that the IP interface be restarted in order for a new configuration that
1187 is bound to the interface to take effect. If an implementation requires that the IP interface be restarted,
1188 then given an instance of CIM_IPProtocolEndpoint for which the configuration should be modified and an
1189 instance of CIM_IPAssignmentSettingData that represents the new configuration, a client can:

- 1190 1) Find an instance of CIM_ElementSettingData that associates the CIM_IPAssignmentSettingData
1191 instance with the CIM_IPProtocolEndpoint instance.
- 1192 2) Set the IsNext property of the CIM_ElementSettingData instance to a value of 1 (Is Next).
- 1193 3) Invoke the RequestStateChange() method of the CIM_IPProtocolEndpoint instance, with a
1194 RequestedState of 11 (Reset).

1195 **9.11 Determine whether DNS configuration was DHCP assigned**

1196 Starting at the CIM_DNSProtocolEndpoint instance, a client can determine if any elements of the DNS
1197 configuration were assigned through DHCP as follows:

- 1198 1) Find the instance of CIM_IPProtocolEndpoint that is associated through an instance of
1199 CIM_SAPSAPDependency.
- 1200 2) Find the instance of CIM_DHCPProtocolEndpoint that is associated with the
1201 CIM_IPProtocolEndpoint instance through an instance of CIM_SAPSAPDependency.
- 1202 3) Query the EnabledState property of the CIM_DHCPProtocolEndpoint instance for the value 2
1203 (Enabled) to ensure that the DHCP client was used.
- 1204 4) Query the OptionsReceived property of the CIM_DHCPProtocolEndpoint instance to determine
1205 if one of the DNS-related options (8, 14, or 17) was received.

1206 **9.12 Determine whether ElementName can be modified**

1207 A client can determine whether it can modify the ElementName property of an instance of
1208 CIM_IPProtocolEndpoint as follows:

- 1209 1) Find the CIM_EnabledLogicalElementCapabilities instance that is associated with the
1210 CIM_IPProtocolEndpoint instance.
- 1211 2) Query the value of the ElementNameEditSupported property of the
1212 CIM_EnabledLogicalElementCapabilities instance. If the value is TRUE, the client can modify
1213 the ElementName property of the target instance.

1214 **9.13 Determine whether state management is supported**

1215 A client can determine whether state management is supported for an instance of
1216 CIM_IPProtocolEndpoint as follows:

- 1217 1) Find the CIM_EnabledLogicalElementCapabilities instance that is associated with the
1218 CIM_IPProtocolEndpoint instance.
- 1219 2) Query the value of the RequestedStatesSupported property. If at least one value is specified,
1220 state management is supported.

1221 **10 CIM Elements**

1222 Table 15 shows the instances of CIM Elements for this profile. Instances of the CIM Elements shall be
 1223 implemented as described in Table 15. Clauses 7 (“Implementation”) and 8 (“Methods”) may impose
 1224 additional requirements on these elements.

1225 **Table 15 – 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 | | |

1226 **10.1 CIM_BindsToLANEndpoint**

1227 CIM_BindsToLANEndpoint relates the CIM_IPProtocolEndpoint instance with the CIM_LANEndpoint
 1228 instance on which it depends. Table 16 provides information about the properties of
 1229 CIM_BindsToLANEndpoint.

1230 **Table 16 – 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 |

1231 **10.2 CIM_ElementCapabilities**

1232 CIM_ElementCapabilities associates an instance of CIM_EnabledLogicalElementCapabilities with the
 1233 CIM_IPProtocolEndpoint instance. Table 17 provides information about the properties of
 1234 CIM_ElementCapabilities.

1235 **Table 17 – 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 |

1236 **10.3 CIM_ElementSettingData — CIM_IPAssignmentSettingData Reference**

1237 CIM_ElementSettingData associates instances of CIM_IPAssignmentSettingData with the
 1238 CIM_IPProtocolEndpoint instance. Table 18 provides information about the properties of
 1239 CIM_ElementSettingData.

1240 **Table 18 – 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) |

1241 **10.4 CIM_ElementSettingData — CIM_StaticIPAssignmentSettingData Reference**

1242 CIM_ElementSettingData associates instances of CIM_StaticIPAssignmentSettingData with the
 1243 CIM_IPProtocolEndpoint instance. Table 19 provides information about the properties of
 1244 CIM_ElementSettingData.

1245 **Table 19 – 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) |

1246 **10.5 CIM_EnabledLogicalElementCapabilities**

1247 CIM_EnabledLogicalElementCapabilities indicates support for managing the IP interface. Table 20
1248 provides information about the properties of CIM_EnabledLogicalElementCapabilities.

1249 **Table 20 – 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. |

1250 **10.6 CIM_HostedAccessPoint — CIM_RemoteServiceAccessPoint Reference**

1251 An instance of CIM_HostedAccessPoint Association between an instance of CIM_ProtocolEndpoint and
1252 CIM_RemoteServiceAccessPoint shall only be instantiated if CIM_RemoteServiceAccessPoint is
1253 supported.

1254 CIM_HostedAccessPoint relates the CIM_RemoteServiceAccessPoint instance that represents the
1255 default gateway with its scoping CIM_ComputerSystem instance. Table 21 provides information about the
1256 properties of CIM_HostedAccessPoint.

1257 **Table 21 – 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 * |

1258 **10.7 CIM_HostedAccessPoint — CIM_IPProtocolEndpoint Reference**

1259 CIM_HostedAccessPoint relates the Central Instance with its Scoping Instance. Table 22 provides
 1260 information about the properties of CIM_HostedAccessPoint.

1261 **Table 22 – Class: CIM_HostedAccessPoint — CIM_IPProtocolEndpoint**

| 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_IPProtocolEndPoint. Cardinality 1..* |

1262 **10.8 CIM_HostedService**

1263 CIM_HostedService relates the CIM_IPConfigurationService instance to its scoping
 1264 CIM_ComputerSystem instance. Table 23 provides information about the properties of
 1265 CIM_HostedService.

1266 **Table 23 – 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 * |

1267 **10.9 CIM_IPAssignmentSettingData**

1268 CIM_IPAssignmentSettingData is the aggregation point for the SettingData instances that define a
 1269 configuration that can be applied to an IP interface. Table 24 provides information about the properties of
 1270 CIM_IPAssignmentSettingData.

1271 **Table 24 – Class: CIM_IPAssignmentSettingData**

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

1272 **10.10 CIM_IPConfigurationService**

1273 CIM_IPConfigurationService represents the ability to configure an IP interface. Table 25 provides
 1274 information about the properties of CIM_IPConfigurationService.

1275 **Table 25 – 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. |

1276 **10.11 CIM_IPProtocolEndpoint**

1277 CIM_IPProtocolEndpoint represents an IP interface that is associated with an Ethernet interface. Table 26
 1278 provides information about the properties of CIM_IPProtocolEndpoint.

1279 **Table 26 – Class: CIM_IPProtocolEndpoint**

| Elements | Requirement | Description |
|-------------------------|-------------|--------------------------|
| SystemCreationClassName | Mandatory | Key |
| CreationClassName | Mandatory | Key |
| SystemName | Mandatory | Key |
| Name | Mandatory | Key |
| NameFormat | Mandatory | Pattern ".*" |
| ProtocolIFType | 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. |

1280 **10.12 CIM_OrderedComponent**

1281 CIM_OrderedComponent associates an instance of CIM_IPAssignmentSettingData to the instances of
 1282 CIM_StaticIPAssignmentSettingData, CIM_DHCPSettingData, CIM_DNSSettingData, and

1283 CIM_DNSGeneralSettingData that compose a configuration. Table 27 provides information about the
 1284 properties of CIM_OrderedComponent.

1285 **Table 27 – 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. |

1286 10.13 CIM_RegisteredProfile

1287 CIM_RegisteredProfile identifies the *IP Interface Profile* in order for a client to determine whether an
 1288 instance of CIM_IPProtocolEndpoint is conformant with this profile. The CIM_RegisteredProfile class is
 1289 defined by the [Profile Registration Profile](#). With the exception of the mandatory values specified for the
 1290 properties in Table 28, the behavior of the CIM_RegisteredProfile instance is in accordance with the
 1291 [Profile Registration Profile](#).

1292 **Table 28 – 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 "1.1.1". |
| RegisteredOrganization | Mandatory | This property shall have a value of "DMTF". |

1293 NOTE: Previous versions of this document included the suffix "Profile" for the RegisteredName value. If
 1294 implementations querying for the RegisteredName value find the suffix "Profile", they should ignore the suffix, with
 1295 any surrounding white spaces, before any comparison is done with the value as specified in this document.

1296 10.14 CIM_RemoteAccessAvailableToElement

1297 CIM_RemoteAccessAvailableToElement associates the CIM_IPProtocolEndpoint instance with the
 1298 CIM_RemoteServiceAccessPoint instance that represents the network gateway. Table 29 provides
 1299 information about the properties of CIM_RemoteAccessAvailableToElement.

1300 **Table 29 – 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. |

1301 **10.15 CIM_RemoteServiceAccessPoint**

1302 CIM_RemoteServiceAccessPoint represents the managed system's view of the default gateway. Table
 1303 30 provides information about the properties of CIM_RemoteServiceAccessPoint.

1304 **Table 30 – 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 ".*" |

1305 **10.16 CIM_ServiceAffectsElement**

1306 CIM_ServiceAffectsElement associates an instance of CIM_IPConfigurationService with an instance of
 1307 CIM_IPProtocolEndpoint that the service is able to configure. Table 31 provides information about the
 1308 properties of CIM_ServiceAffectsElement.

1309 **Table 31 – 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..* |
| ElementEffects | Mandatory | Matches 5 (Manages) |

1310 **10.17 CIM_StaticIPAssignmentSettingData**

1311 CIM_StaticIPAssignmentSettingData represents a static configuration that can be applied to an instance
 1312 of CIM_IPProtocolEndpoint. Table 32 provides information about the properties of
 1313 CIM_StaticIPAssignmentSettingData.

1314 **Table 32 – 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 | |

1315

**ANNEX A
(informative)**

Change log

1316
1317
1318
1319

| 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. |
| 1.1.1 | 2012-02-23 | Errata 1.1.1 Section 8 - Removed CIM_SystemDevice. Section 9 - Correction in association for CIM_RemoteServiceAccessPoint. Section 10 - Spelling for CIM_ServiceAffectsElement.ElementEffects |

1320
1321