



1

2

3

4

**Document Number: DSP1036**

**Date: 2008-09-04**

**Version: 1.0.0**

5 **IP Interface Profile**

6 **Document Type: Specification**

7 **Document Status: Final Standard**

8 **Document Language: E**

9

## 10 Copyright Notice

11 Copyright © 2008 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 for uses consistent with this purpose, provided that correct attribution is given. As DMTF  
15 specifications may be revised from time to time, the particular version and release date should always be  
16 noted.

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27 withdrawn or modified after publication, and shall be indemnified and held harmless by any party  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

# CONTENTS

|    |  |    |
|----|--|----|
| 31 | Foreword .....   | 6  |
| 32 | Introduction .....   | 7  |
| 33 | 1 Scope .....  | 9  |
| 34 | 2 Normative References.....  | 9  |
| 35 | 2.1 Approved References .....  | 9  |
| 36 | 2.2 Other References.....  | 9  |
| 37 | 3 Terms and Definitions .....  | 9  |
| 38 | 4 Symbols and Abbreviated Terms .....                                    | 11 |
| 39 | 5 Synopsis .....   | 11 |
| 40 | 6 Description .....  | 12 |
| 41 | 6.1 Pending and Alternate Configuration Management.....                  | 13 |
| 42 | 7 Implementation.....  | 13 |
| 43 | 7.1 Basic IP Configuration .....   | 13 |
| 44 | 7.2 DHCP Client Is Supported .....                                       | 18 |
| 45 | 7.3 DNS Client Is Supported.....   | 18 |
| 46 | 7.4 Managing Alternate Configurations—Optional .....                     | 18 |
| 47 | 7.5 Applying an Alternate Configuration .....                            | 21 |
| 48 | 7.6 Relationship with a Network Interface .....                          | 23 |
| 49 | 8 Methods.....   | 23 |
| 50 | 8.1 CIM_IPProtocolEndpoint.RequestStateChange( ) .....                   | 23 |
| 51 | 8.2 CIM_IPConfigurationService.ApplySettingToIPProtocolEndpoint( ) ..... | 24 |
| 52 | 8.3 Profile Conventions for Operations.....                              | 25 |
| 53 | 8.4 CIM_BindsToLANEndpoint.....  | 26 |
| 54 | 8.5 CIM_ElementSettingData .....   | 26 |
| 55 | 8.6 CIM_HostedAccessPoint .....  | 27 |
| 56 | 8.7 CIM_HostedService .....  | 27 |
| 57 | 8.8 CIM_IPAssignmentSettingData .....                                    | 28 |
| 58 | 8.9 CIM_IPConfigurationService .....                                     | 28 |
| 59 | 8.10 CIM_IPProtocolEndpoint .....  | 28 |
| 60 | 8.11 CIM_OrderedComponent .....  | 29 |
| 61 | 8.12 CIM_RemoteAccessAvailableToElement.....                             | 29 |
| 62 | 8.13 CIM_RemoteServiceAccessPoint.....                                   | 29 |
| 63 | 8.14 CIM_ServiceAffectsElement .....                                     | 30 |
| 64 | 8.15 CIM_StaticIPAssignmentSettingData .....                             | 30 |
| 65 | 8.16 CIM_SystemDevice .....  | 30 |
| 66 | 9 Use Cases.....   | 31 |
| 67 | 9.1 Miscellaneous Object Diagrams .....                                  | 31 |
| 68 | 9.2 Determine Supported Configuration Methods .....                      | 43 |
| 69 | 9.3 Determine Gateway Address.....                                       | 43 |
| 70 | 9.4 Determine Method Used for Current Configuration .....                | 43 |
| 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.....                             | 44 |
| 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.....                   | 45 |
| 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.....   | 46 |
| 81 | 10.1 CIM_BindsToLANEndpoint.....   | 47 |
| 82 | 10.2 CIM_ElementCapabilities .....                                       | 47 |

|    |       |  |    |
|----|-------|--|----|
| 83 | 10.3  | CIM_ElementSettingData—CIM_IPAssignmentSettingData Reference       | 48 |
| 84 | 10.4  | CIM_ElementSettingData—CIM_StaticIPAssignmentSettingData Reference | 48 |
| 85 | 10.5  | CIM_EnabledLogicalElementCapabilities                              | 48 |
| 86 | 10.6  | CIM_HostedAccessPoint—CIM_RemoteServiceAccessPoint Reference       | 49 |
| 87 | 10.7  | CIM_HostedAccessPoint—CIM_IPProtocolEndpoint Reference             | 49 |
| 88 | 10.8  | CIM_HostedService  | 49 |
| 89 | 10.9  | CIM_IPAssignmentSettingData  | 50 |
| 90 | 10.10 | CIM_IPConfigurationService   | 50 |
| 91 | 10.11 | CIM_IPProtocolEndpoint   | 50 |
| 92 | 10.12 | CIM_OrderedComponent   | 51 |
| 93 | 10.13 | CIM_RegisteredProfile  | 51 |
| 94 | 10.14 | CIM_RemoteAccessAvailableToElement                                 | 52 |
| 95 | 10.15 | CIM_RemoteServiceAccessPoint                                       | 52 |
| 96 | 10.16 | CIM_ServiceAffectsElement  | 52 |
| 97 | 10.17 | CIM_StaticIPAssignmentSettingData                                  | 53 |
| 98 |       | ANNEX A (informative) Change Log                                   | 54 |
| 99 |       | ANNEX B (informative) Acknowledgments                              | 55 |

100

## 101 Figures

|     |  |    |
|-----|--|----|
| 102 | Figure 1 – IP Interface Profile: Class Diagram       | 12 |
| 103 | Figure 2 – Registered Profile                        | 31 |
| 104 | Figure 3 – Basic Configuration – IPv4                | 32 |
| 105 | Figure 4 – Basic Configuration – IPv6                | 32 |
| 106 | Figure 5 – Basic Configuration – IPv4 and IPv6       | 33 |
| 107 | Figure 6 – Static Current and Pending Configuration  | 34 |
| 108 | Figure 7 – Static and DHCP Pending Configurations    | 35 |
| 109 | Figure 8 – DHCP Timed Out to a Static Configuration  | 36 |
| 110 | Figure 9 – Service Processor and Server Share a NIC  | 37 |
| 111 | Figure 10 – Configuration Choices                    | 38 |
| 112 | Figure 11 – DHCP Assigned Partial DNS                | 39 |
| 113 | Figure 12 – DHCP with DNS Statically Configured      | 40 |
| 114 | Figure 13 – Static without DNS Configuration – One   | 41 |
| 115 | Figure 14 – Static without DNS Configuration – Two   | 42 |
| 116 | Figure 15 – Static without DNS Configuration – Three | 43 |

117

## 118 Tables

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

130 Table 11 – Operations: CIM\_OrderedComponent..... 29

131 Table 12 – Operations: CIM\_RemoteAccessAvailableToElement ..... 29

132 Table 13 – Operations: CIM\_ServiceAffectsElement ..... 30

133 Table 14 – Operations: CIM\_StaticIPAssignmentSettingData..... 30

134 Table 15 – Operations: CIM\_SystemDevice..... 30

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

136 Table 17 – Class: CIM\_BindsToLANEndpoint..... 47

137 Table 18 – Class: CIM\_ElementCapabilities..... 47

138 Table 19 – Class: CIM\_ElementSettingData – CIM\_IPAssignmentSettingData ..... 48

139 Table 20 – Class: CIM\_ElementSettingData – CIM\_StaticIPAssignmentSettingData ..... 48

140 Table 21 – Class: CIM\_EnabledLogicalElementCapabilities..... 48

141 Table 22 – Class: CIM\_HostedAccessPoint – CIM\_RemoteServiceAccessPoint..... 49

142 Table 23 – Class: CIM\_HostedAccessPoint – CIM\_IPProtocolEndpoint ..... 49

143 Table 24 – Class: CIM\_HostedService ..... 49

144 Table 25 – Class: CIM\_IPAssignmentSettingData ..... 50

145 Table 26 – Class: CIM\_IPConfigurationService..... 50

146 Table 27 – Class: CIM\_IPProtocolEndpoint..... 50

147 Table 28 – Class: CIM\_OrderedComponent..... 51

148 Table 29 – Class: CIM\_RegisteredProfile..... 51

149 Table 30 – Class: CIM\_RemoteAccessAvailableToElement..... 52

150 Table 31 – Class: CIM\_RemoteServiceAccessPoint..... 52

151 Table 32 – Class: CIM\_ServiceAffectsElement ..... 52

152 Table 33 – Class: CIM\_StaticIPAssignmentSettingData ..... 53

153

154

## Foreword

155 The *IP Interface Profile* (DSP1036) was prepared by the Server Management Working Group.

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

158

## Introduction

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





164

# IP Interface Profile

## 165 1 Scope

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

## 170 2 Normative References

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

### 174 2.1 Approved References

- 175 DMTF [DSP0200](#), *CIM Operations over HTTP 1.2.0*  
176 DMTF [DSP0004](#), *CIM Infrastructure Specification 2.3.0*  
177 DMTF [DSP1037](#), *DHCP Client Profile*  
178 DMTF [DSP1038](#), *DNS Client Profile*  
179 DMTF [DSP1035](#), *Host LAN Network Port Profile*  
180 DMTF [DSP1000](#), *Management Profile Specification Template 1.0.0*  
181 DMTF [DSP1001](#), *Management Profile Specification Usage Guide 1.0.0*  
182 DMTF [DSP1033](#), *Profile Registration Profile*

### 183 2.2 Other References

- 184 [ISO/IEC Directives, Part 2](#), *Rules for the structure and drafting of International Standards*  
185 [Unified Modeling Language \(UML\) from the Open Management Group \(OMG\)](#)  
186 IETF, [RFC 2131](#), *Dynamic Host Configuration Protocol*, March 1997  
187 IETF, [RFC 1541](#), *Dynamic Host Configuration Protocol*, October 1993  
188 IETF, [RFC 1208](#), *A Glossary of Networking Terms*, March 1991  
189 IETF, [RFC 4291](#), *IP Version 6 Addressing Architecture*, February 2006

## 190 3 Terms and Definitions

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

### 193 3.1

#### 194 **can**

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

- 196 **3.2**  
197 **cannot**  
198 used for statements of possibility and capability, whether material, physical, or causal
- 199 **3.3**  
200 **conditional**  
201 indicates requirements to be followed strictly to conform to the document when the specified conditions  
202 are met
- 203 **3.4**  
204 **mandatory**  
205 indicates requirements to be followed strictly to conform to the document and from which no deviation is  
206 permitted
- 207 **3.5**  
208 **may**  
209 indicates a course of action permissible within the limits of the document
- 210 **3.6**  
211 **need not**  
212 indicates a course of action permissible within the limits of the document
- 213 **3.7**  
214 **optional**  
215 indicates a course of action permissible within the limits of the document
- 216 **3.8**  
217 **pending configuration**  
218 the configuration that will be applied to an IP interface the next time the interface accepts a configuration
- 219 **3.9**  
220 **referencing profile**  
221 indicates a profile that owns the definition of this class and can include a reference to this profile in its  
222 "Referenced Profiles" table
- 223 **3.10**  
224 **shall**  
225 indicates requirements to be followed strictly to conform to the document and from which no deviation is  
226 permitted
- 227 **3.11**  
228 **shall not**  
229 indicates requirements to be followed strictly to conform to the document and from which no deviation is  
230 permitted
- 231 **3.12**  
232 **should**  
233 indicates that among several possibilities, one is recommended as particularly suitable, without  
234 mentioning or excluding others, or that a certain course of action is preferred but not necessarily required
- 235 **3.13**  
236 **should not**  
237 indicates that a certain possibility or course of action is deprecated but not prohibited

238 **3.14**  
 239 **unspecified**  
 240 indicates that this profile does not define any constraints for the referenced CIM element or operation

## 241 **4 Symbols and Abbreviated Terms**

### 242 **Experimental Maturity Level**

243  
 244 Some of the content considered for inclusion in *IP Interface Profile* has yet to receive sufficient review to  
 245 satisfy the adoption requirements set forth by the Technical Committee within the DMTF. This content is  
 246 presented here as an aid to implementers who are interested in likely future developments within this  
 247 specification. The content marked experimental may change as implementation experience is gained.  
 248 There is a high likelihood that it will be included in an upcoming revision of the specification. Until that  
 249 time, it is purely informational, and is clearly marked within the text.  
 250 A sample of the typographical convention for experimental content is included here:

---

### 251 **EXPERIMENTAL**

252 Experimental content appears here

---

### 253 **EXPERIMENTAL**

---

254 The following abbreviations are used in this document.

255 **4.1**  
 256 **DHCP**  
 257 Dynamic Host Configuration Protocol

258 **4.2**  
 259 **DNS**  
 260 Domain Name System

261 **4.3**  
 262 **IP**  
 263 Internet Protocol

## 264 **5 Synopsis**

265 **Profile Name:** IP Interface  
 266 **Version:** 1.0.0  
 267 **Organization:** DMTF  
 268 **CIM Schema Version:** 2.19  
 269 **Central Class:** CIM\_IPProtocolEndpoint  
 270 **Scoping Class:** CIM\_ComputerSystem

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

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

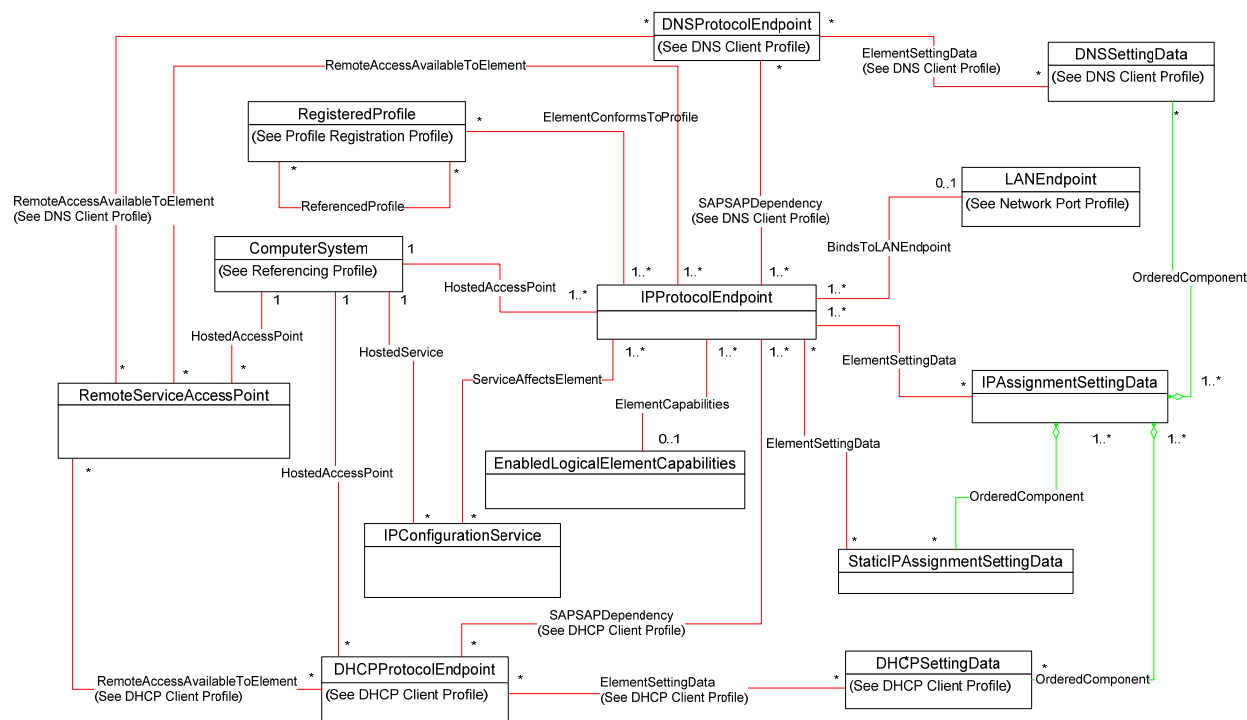
276 **Table 1 – Referenced Profiles**

| Profile Name                 | Organization | Version | Relationship | Behavior         |
|------------------------------|--------------|---------|--------------|------------------|
| <i>Profile Registration</i>  | DMTF         | 1.0.0   | Mandatory    | None             |
| <i>DNS Client</i>            | DMTF         | 1.0.0   | Optional     | See section 7.3. |
| <i>DHCP Client</i>           | DMTF         | 1.0.0   | Optional     | See section 7.2. |
| <i>Host LAN Network Port</i> | DMTF         | 1.0.0   | Optional     | See section 7.6. |

277 **6 Description**

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

280 Figure 1 represents the class schema for the *IP Interface Profile*. For simplicity, the CIM\_ prefix has been  
 281 removed from the names of the classes. Note that this class diagram is meant to be used in conjunction  
 282 with the class diagrams from the *DHCP Client Profile* (DSP1037) and the *DNS Client Profile* (DSP1038).



283

284 **Figure 1 – IP Interface Profile: Class Diagram**

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

- 288 • IPv4 interface (optionally associated with a network interface)
- 289 • optional relationship with a DNS client

290       • optional relationship with a DHCP client

291       • current and pending configurations

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

293       • modeling of the network gateway

294       • modeling of TCP/UDP ports

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

298   This profile represents the current configuration of an IP interface, associated configurations that could be  
299   applied, the DNS client, and the DHCP client. Support for the DNS and DHCP clients is not required. In  
300   general, the various subclasses of CIM\_ProtocolEndpoint reflect the current configuration and status of  
301   their respective elements.

302   Functionality provided by other systems (Gateway, DHCP server, and DNS server) is modeled from the  
303   IP interface view and is therefore represented by instances of CIM\_RemoteServiceAccessPoint.

## 304   **6.1 Pending and Alternate Configuration Management**

305   Pending configurations, which are associated with the IP interface and could be applied in the future, are  
306   represented by instances of CIM\_IPAssignmentSettingData and its subclasses. Each pending  
307   configuration can include multiple settings that will be applied to the different elements of the endpoint  
308   configuration. Settings for a particular element of the configuration are represented with the appropriate  
309   subclass of CIM\_IPAssignmentSettingData and aggregated into one or more instances of  
310   CIM\_IPAssignmentSettingData that represent the configuration.

311   The management of DNS and DHCP clients as part of an alternate configuration is handled differently for  
312   the two clients. DHCP and static IP configuration management are generally treated as alternatives to  
313   each other. For the basic configuration of an IP interface, the information is assigned either statically or  
314   through DHCP. DNS configuration occurs differently. When DNS and static configuration occur together,  
315   there is no overlap. Thus the DNS settings that are part of the configuration are applied to the DNS client.  
316   When DHCP and DNS settings are used together, portions of the DNS configuration can potentially be  
317   assigned through DHCP.

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

## 324   **7 Implementation**

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

### 327   **7.1 Basic IP Configuration**

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

#### 329   **7.1.1 CIM\_IPProtocolEndpoint**

330   An instance of CIM\_IPProtocolEndpoint shall represent the IP interface. The properties of the instance of  
331   CIM\_IPProtocolEndpoint shall reflect the current configuration of an IP interface.

**332 7.1.1.1 CIM\_IPProtocolEndpoint.AddressOrigin**

333 The AddressOrigin property indicates the configuration method that resulted in the configuration being  
334 assigned to the CIM\_IPProtocolEndpoint.

**335 7.1.1.1.1 AddressOrigin – Static**

336 A value of 3 (Static) shall indicate that the configuration was assigned statically. The AddressOrigin  
337 property shall have a value of 3 (Static) when the configuration is the result of an instance of  
338 CIM\_StaticIPAssignmentSettingData being successfully applied. Section 7.5.3.3 explains what it means  
339 for settings to be successfully applied.

**340 7.1.1.1.2 AddressOrigin – DHCP**

341 A value of 4 (DHCP) shall indicate that the configuration was obtained through an associated DHCP  
342 client. The AddressOrigin property shall have a value of 4 (DHCP) when the configuration is the result of  
343 an instance of CIM\_DHCPSettingData being successfully applied.

**344 7.1.1.2 CIM\_IPProtocolEndpoint.ProtocolIFType**

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

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

347 The value of CIM\_IPProtocolEndpoint.ProtocolIFType shall be 4096

---

**348 EXPERIMENTAL**

349 , 4097, or 4098.

350 If the value is 4097 (Ipv6) the IPv6Address, IPv6AddressType, and IPv6SubnetPrefixLength properties  
351 shall be implemented.

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

---

**354 EXPERIMENTAL****355 7.1.1.3 CIM\_IPProtocolEndpoint.IPv4Address**

356 If the value of CIM\_IPProtocolEndpoint.ProtocolIFType is 4096 (IPv4), the IPv4Address property shall  
357 indicate the current IPv4 address assigned to this IP endpoint. The value of the property shall be  
358 specified in dotted decimal notation as defined in IETF [RFC 1208](#). A value of 0.0.0.0 shall indicate that a  
359 valid IP address is not assigned to this IP endpoint.

**360 7.1.1.4 CIM\_IPProtocolEndpoint.SubnetMask**

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

---

**364 EXPERIMENTAL****365 7.1.1.5 CIM\_IPProtocolEndpoint.Ipv6Address**

366 If the value of CIM\_IPProtocolEndpoint.ProtocolIFType is 4097 (IPv6) or 4098 (IPv4/IPv6), the  
367 Ipv6Address property shall indicate the current IPv6 address assigned to this IP endpoint. The value of  
368 the property shall be specified in the notation specified in IETF [RFC 4291](#), section 2.2.

**369 EXPERIMENTAL**

---

**370 7.1.2 IP Interface State Management Is Supported—Conditional**

371 When management of the state of an IP interface is supported, exactly one instance of  
372 CIM\_EnabledLogicalElementCapabilities shall be associated with the CIM\_IPProtocolEndpoint instance  
373 through an instance of CIM\_ElementCapabilities. The existence of the CIM\_ElementCapabilities instance  
374 is conditional on the existence of the CIM\_EnabledLogicalElementCapabilities instance.

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

**377 7.1.2.1 CIM\_EnabledLogicalElementCapabilities**

378 The instance of CIM\_EnabledLogicalElementCapabilities is used to advertise the state management  
379 supported for the IP interface.

**380 7.1.2.1.1 CIM\_EnabledLogicalElementCapabilities.RequestedStatesSupported**

381 The RequestedStatesSupported property may contain zero or more of the following values: 2 (Enabled),  
382 3 (Disabled), or 11 (Reset).

**383 7.1.2.2 CIM\_IPProtocolEndpoint.RequestedState**

384 When the CIM\_IPProtocolEndpoint.RequestStateChange() method is successfully invoked, the value of  
385 the RequestedState property shall be the value of the RequestedState parameter. If the method is not  
386 successfully invoked, the value of the RequestedState property is indeterminate.

387 The CIM\_IPProtocolEndpoint.RequestedState property shall have one of the values specified in the  
388 CIM\_EnabledLogicalElementCapabilities.RequestedStatesSupported property or a value of 5 (No  
389 Change).

**390 7.1.2.3 CIM\_IPProtocolEndpoint.EnabledState**

391 When the RequestedState parameter has a value of 2 (Enabled) or 3 (Disabled) and the  
392 CIM\_IPProtocolEndpoint.RequestStateChange() method completes successfully, the value of the  
393 EnabledState property shall equal the value of the CIM\_IPProtocolEndpoint.RequestedState property.

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

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

**397 7.1.3 IP Interface State Management Is Not Supported**

398 This section describes the CIM elements and behaviors that shall be implemented when management of  
399 the IP Interface state is not supported.

**400 7.1.3.1 CIM\_EnabledLogicalElementCapabilities**

401 When state management is not supported, exactly one instance of  
402 CIM\_EnabledLogicalElementCapabilities may be associated with the CIM\_IPProtocolEndpoint instance  
403 through an instance of CIM\_ElementCapabilities.

**404 7.1.3.1.1 CIM\_EnabledLogicalElementCapabilities.RequestedStatesSupported**

405 The CIM\_EnabledLogicalElementCapabilities.RequestedStatesSupported property shall not contain any  
406 values.

**407 7.1.3.2 CIM\_IPProtocolEndpoint.RequestedState**

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

**409 7.1.3.3 CIM\_IPProtocolEndpoint.EnabledState**

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

**412 7.1.4 Modifying ElementName Is Supported—Conditional**

413 The CIM\_IPProtocolEndpoint.ElementName property may support being modified by the ModifyInstance  
414 operation. See section 8.10.1.1.

415 This behavior is conditional. This section describes the CIM elements and behavior requirements when  
416 an implementation supports client modification of the CIM\_IPProtocolEndpoint.ElementName property.

**417 7.1.4.1 CIM\_EnabledLogicalElementCapabilities**

418 An instance of CIM\_EnabledLogicalElementCapabilities shall be associated with the  
419 CIM\_IPProtocolEndpoint instance through an instance of CIM\_ElementCapabilities.

**420 7.1.4.1.1 CIM\_EnabledLogicalElementCapabilities.ElementNameEditSupported**

421 The ElementNameEditSupported property shall have a value of TRUE.

**422 7.1.4.1.2 CIM\_EnabledLogicalElementCapabilities.MaxElementNameLen**

423 The MaxElementNameLen property shall be implemented.

**424 7.1.5 Modifying ElementName Is Not Supported**

425 This section describes the CIM elements and behaviors that shall be implemented when the  
426 CIM\_IPProtocolEndpoint.ElementName property does not support being modified by the ModifyInstance  
427 operation.

**428 7.1.5.1 CIM\_EnabledLogicalElementCapabilities**

429 An instance of CIM\_EnabledLogicalElementCapabilities may be associated with the  
430 CIM\_IPProtocolEndpoint instance through an instance of CIM\_ElementCapabilities.

**431 7.1.5.1.1 CIM\_EnabledLogicalElementCapabilities.ElementNameEditSupported**

432 The ElementNameEditSupported property shall have a value of FALSE.

**433 7.1.5.1.2 CIM\_EnabledLogicalElementCapabilities.MaxElementNameLen**

434 The MaxElementNameLen property may be implemented. The MaxElementNameLen property is  
435 irrelevant in this context.



### 436 7.1.6 Default Gateway

437 An IP interface can be configured with the address of a network gateway. Modeling of the default gateway  
438 is optional. When the IP interface is configured with the address of a default gateway, an instance of  
439 CIM\_RemoteServiceAccessPoint shall represent the default gateway. The instance of  
440 CIM\_RemoteServiceAccessPoint shall be associated with the instance of CIM\_IPProtocolEndpoint  
441 through an instance of CIM\_RemoteAccessAvailableToElement. An instance of  
442 CIM\_RemoteServiceAccessPoint may represent the default gateway even when a valid default gateway  
443 has not been configured for the IP interface. It can be more convenient for an implementation to always  
444 instantiate the instance of CIM\_RemoteServiceAccessPoint even if a default gateway has not been  
445 assigned to the IP interface rather than conditionally provide the relevant instances. For IPv4, this will  
446 result in a single instance of CIM\_RemoteServiceAccessPoint associated with the instance of  
447 CIM\_IPProtocolEndpoint.

---

#### 448 EXPERIMENTAL

449 For IPv6 or IPv4/IPv6 there may be one or more instances of CIM\_RemoteServiceAccessPoint  
450 associated with the instance of CIM\_IPProtocolEndpoint, since there may be more than one default  
451 gateway. In this case, the use of CIM\_RemoteAccessAvailableToElement.OrderOfAccess can be used to  
452 represent the list of default gateways in priority order.

#### 453 EXPERIMENTAL

---

454 For IPv6 or IPv4/IPv6 there may be one or more instances of CIM\_RemoteServiceAccessPoint  
455 associated with the instance of CIM\_IPProtocolEndpoint, since there may be more than one default  
456 gateway. In this case, the use of CIM\_RemoteAccessAvailableToElement.OrderOfAccess can be used to  
457 represent the list of default gateways in priority order.

#### 458 7.1.6.1 CIM\_RemoteServiceAccessPoint.AccessInfo

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

---

#### 463 EXPERIMENTAL

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

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

#### 472 EXPERIMENTAL

---

#### 473 7.1.6.2 CIM\_RemoteAccessAvailableToElement.Antecedent

474 The value of the Antecedent reference shall be the instance of CIM\_RemoteServiceAccessPoint.  
475 Cardinality \*.

#### 476 7.1.6.3 CIM\_RemoteAccessAvailableToElement.Dependent

477 The value of the Dependent reference shall be the instance of CIM\_IPProtocolEndpoint. Cardinality \*.

---

#### 478 **7.1.6.4 CIM\_RemoteAccessAvailableToElement.OrderOfAccess**

479 If the associated value of CIM\_IPProtocolEndpoint.ProtocolIFType = 4096 (IPv4), then the  
480 OrderOfAccess property shall have a value of 0 (Zero).

### 481 **7.2 DHCP Client Is Supported**

482 When a DHCP client is supported for the IP interface, the *DHCP Client Profile* shall be supported. This  
483 behavior is optional.

### 484 **7.3 DNS Client Is Supported**

485 When a DNS client is supported for the IP interface, the *DNS Client Profile* shall be supported. This  
486 behavior is optional.

### 487 **7.4 Managing Alternate Configurations—Optional**

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

#### 491 **7.4.1 Configuration Management Is Supported**

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

498 The CIM\_IPConfigurationService instance shall be associated with a CIM\_ComputerSystem instance  
499 through an instance of CIM\_HostedService. The existence of the CIM\_HostedService association is  
500 conditional on the existence of the CIM\_IPConfigurationService instance.

#### 501 **7.4.2 Representing an Alternate Configuration Using CIM\_IPAssignmentSettingData**

502 Each instance of CIM\_IPAssignmentSettingData shall represent a possible configuration for an IP  
503 interface. The detailed settings for the IP interface shall be contained in the instances of subclasses of  
504 CIM\_IPAssignmentSettingData, which are associated with the instance of CIM\_IPAssignmentSettingData  
505 through instances of CIM\_OrderedComponent.

506 The existence of one or more instances of CIM\_IPAssignmentSettingData is conditional on the existence  
507 of the CIM\_IPConfigurationService instance. The existence of one or more instances of  
508 CIM\_ElementSettingData is conditional on the existence of one or more instances of  
509 CIM\_IPAssignmentSettingData.

##### 510 **7.4.2.1 Associating an Alternate Configuration with an IP Interface**

511 The instance of CIM\_IPAssignmentSettingData shall be associated with the instance of  
512 CIM\_IPProtocolEndpoint through an instance of CIM\_ElementSettingData.

###### 513 **7.4.2.1.1 CIM\_ElementSettingData.IsCurrent**

514 When an instance of CIM\_ElementSettingData associates an instance of CIM\_IPAssignmentSettingData  
515 with an instance of CIM\_IPProtocolEndpoint, the CIM\_ElementSettingData.IsCurrent property shall have  
516 a value of 1 (Is Current) when the configuration represented by the referenced instance of  
517 CIM\_IPAssignmentSettingData is the last configuration applied to the IP interface represented by the  
518 referenced instance of CIM\_IPProtocolEndpoint.

519 When an instance of CIM\_ElementSettingData associates an instance of CIM\_IPAssignmentSettingData  
520 with an instance of CIM\_IPProtocolEndpoint, the CIM\_ElementSettingData.IsCurrent property shall have  
521 a value of 2 (Is Not Current) when the configuration represented by the referenced instance of  
522 CIM\_IPAssignmentSettingData is not the last configuration applied to the IP interface represented by the  
523 referenced instance of CIM\_IPProtocolEndpoint.

### 524 **7.4.3 Associating Settings Using CIM\_OrderedComponent**

525 Instances of the subclasses of CIM\_IPAssignmentSettingData contain the details of the IP configuration.  
526 The CIM\_OrderedComponent association aggregates these instances into instances of  
527 CIM\_IPAssignmentSettingData. An instance of CIM\_IPAssignmentSettingData will have one or more  
528 instances of its subclasses associated with it through an instance of CIM\_OrderedComponent. An  
529 instance of a subclass of CIM\_IPAssignmentSettingData will be associated with one or more instances of  
530 CIM\_IPAssignmentSettingData.

#### 531 **7.4.3.1 CIM\_OrderedComponent.GroupComponent**

532 An instance of CIM\_IPAssignmentSettingData shall be the value of the GroupComponent property of an  
533 instance of CIM\_OrderedComponent. Cardinality 1..\*

#### 534 **7.4.3.2 CIM\_OrderedComponent.PartComponent**

535 An instance of a subclass of CIM\_IPAssignmentSettingData shall be the value of the PartComponent  
536 property of an instance of CIM\_OrderedComponent. Cardinality \*

#### 537 **7.4.3.3 Interpretation of CIM\_OrderedComponent.AssignedSequence**

538 The relative value of the CIM\_OrderedComponent.AssignedSequence property shall indicate the order in  
539 which aggregated instances of subclasses of CIM\_IPAssignmentSettingData are applied to their  
540 associated CIM\_ProtocolEndpoint instances.

##### 541 **7.4.3.3.1 Use of 0 (Zero)**

542 When the CIM\_OrderedComponent.AssignedSequence property has a value of 0 (zero), the instance of  
543 CIM\_SettingData referenced by the CIM\_OrderedComponent.PartComponent property shall not be  
544 applied when the configuration represented by the CIM\_IPAssignmentSettingData instance that is the  
545 value of the CIM\_OrderedComponent.GroupComponent property is applied. The  
546 CIM\_OrderedComponent.AssignedSequence property may have the value 0 (zero) when the instance of  
547 CIM\_OrderedComponent references an instance of CIM\_DNSSettingData or  
548 CIM\_DNSGeneralSettingData. The CIM\_OrderedComponent.AssignedSequence property shall not have  
549 the value 0 (zero) when the instance of CIM\_OrderedComponent does not reference an instance of  
550 CIM\_DNSSettingData or CIM\_DNSGeneralSettingData.

##### 551 **7.4.3.3.2 Discreteness**

552 Two instances of CIM\_OrderedComponent that reference the same instance of  
553 CIM\_IPAssignmentSettingData shall not have the same value for their AssignedSequence properties  
554 unless the value is 0 (zero).

### 555 **7.4.4 Alternate Static Configuration**

556 When an implementation supports the manual assignment of an IP configuration to the IP endpoint, an  
557 instance of CIM\_StaticIPAssignmentSettingData shall be associated with the CIM\_IPProtocolEndpoint  
558 through an instance of CIM\_ElementSettingData. This instance of CIM\_StaticIPAssignmentSettingData  
559 shall be associated with at least one instance of CIM\_IPAssignmentSettingData through an instance of  
560 CIM\_OrderedComponent. When the aggregating IP configuration has been applied to the IP interface  
561 and the IP interface is using the settings contained in the instance of  
562 CIM\_StaticIPAssignmentSettingData, the IsCurrent property of the CIM\_ElementSettingData instance

563 has the value 1 (Is Current). Otherwise, the CIM\_ElementSettingData.IsCurrent property shall have the  
564 value 2 (Is Not Current).

#### 565 **7.4.5 Alternate DHCP Configuration**

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

#### 568 **7.4.6 DNS Client Alternate Configuration**

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

#### 571 **7.4.7 Relationship between DHCP and DNS Configuration**

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

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

##### 579 **7.4.7.1 Relationship between DHCP Options and the DNS Configuration**

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

583 DHCPPE – the instance of CIM\_DHCPProtocolEndpoint that represents the DHCP client for an IP  
584 interface

585 DNSPE – the instance of CIM\_DNSProtocolEndpoint that represents the DNS client that is associated  
586 through an instance of CIM\_SAPSAPDependency with the same instance of CIM\_IPProtocolEndpoint  
587 with which the DHCPPE is associated through an instance of CIM\_SAPSAPDependency

588 DNS Pending – the instance of CIM\_DNSSettingData that is associated through an instance of  
589 CIM\_OrderedComponent with the instance of CIM\_IPAssignmentSettingData that is being applied to the  
590 CIM\_IPProtocolEndpoint instance

591 DHCP Pending – the instance of CIM\_DHCPSettingData that is associated through an instance of  
592 CIM\_OrderedComponent with the instance of CIM\_IPAssignmentSettingData that is being applied to the  
593 CIM\_IPProtocolEndpoint instance

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

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

- When the OptionsReceived property of the DHCPPE instance and the DHCPOptionsToUse property of the DNSPE instance both contain the value 17 (Domain Name), the value of the DomainName property of the DNSPE instance shall be the domain name specified by the DHCP server as the data for the Domain Name DHCP option.

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

## 7.4.8 Representing a Pending Configuration

When an implementation supports alternate configurations, exactly one instance of CIM\_IPAssignmentSettingData shall be associated with the Central Instance through an instance of CIM\_ElementSettingData whose IsNext property has the value 1 (Is Next).

Exactly one instance of CIM\_IPAssignmentSettingData may be associated with the Central Instance through an instance of CIM\_ElementSettingData whose IsNext property has the value 3 (Is Next For Single Use).

If an instance of CIM\_IPAssignmentSettingData is associated with the Central Instance through an instance of CIM\_ElementSettingData whose IsNext property has the value 3 (Is Next For Single Use), this instance of CIM\_IPAssignmentSettingData shall represent the pending configuration. If no instance of CIM\_IPAssignmentSettingData is associated with the Central Instance through an instance of CIM\_ElementSettingData whose IsNext property has the value 3 (Is Next For Single Use), the instance of CIM\_IPAssignmentSettingData that is associated with the Central Instance through an instance of CIM\_ElementSettingData whose IsNext property has the value 1 (Is Next) shall represent the pending configuration.

## 7.5 Applying an Alternate Configuration

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

### 7.5.1 Applying the Pending Configuration upon Transition to Enabled

When the value of the EnabledState property of the CIM\_IPProtocolEndpoint instance has a value other than 2 (Enabled) and the value of the EnabledState property transitions to 2 (Enabled), the implementation shall apply the pending configuration.

### 7.5.2 Determining the Target CIM\_ProtocolEndpoint Instance

An instance of CIM\_IPAssignmentSettingData or its subclasses may be associated with more than one instance of a subclass of CIM\_ProtocolEndpoint through instances of CIM\_ElementSettingData. Instances of subclasses of CIM\_IPAssignmentSettingData may be aggregated into one or more instances of CIM\_IPAssignmentSettingData where the aggregating CIM\_IPAssignmentSettingData instances are associated with different instances of CIM\_IPProtocolEndpoint. This is allowed as a convenience for instrumentation to reduce the number of instances required when multiple IP interfaces share a common configuration.

The following rules unambiguously identify the instance of a subclass of CIM\_ProtocolEndpoint that will have an instance of a subclass of CIM\_SettingData applied to it when a pending configuration is applied to an instance of CIM\_IPProtocolEndpoint. Note that the DNS and DHCP related classes are owned by

650 the [DNS Client Profile](#) and [DHCP Client Profile](#), respectively. The algorithm for determining their use is  
651 provided here because it is part of the behavior of applying a configuration.

652 When a pending IP configuration is applied, each instance of CIM\_StaticIPAssignmentSettingData that is  
653 associated with the CIM\_IPAssignmentSettingData instance through an instance of  
654 CIM\_OrderedComponent shall be applied to the CIM\_IPProtocolEndpoint instance that is identified as  
655 follows:

- 656 1) The CIM\_IPProtocolEndpoint instance shall be associated with the  
657 CIM\_StaticIPAssignmentSettingData instance through an instance of CIM\_ElementSettingData.
- 658 2) The CIM\_IPProtocolEndpoint instance shall be the CIM\_IPProtocolEndpoint instance to which  
659 the aggregating CIM\_IPAssignmentSettingData is being applied.

660 When a pending IP configuration is applied, each instance of CIM\_DHCPSettingData that is associated  
661 with the CIM\_IPAssignmentSettingData instance through an instance of CIM\_OrderedComponent shall  
662 be applied to the CIM\_DHCPProtocolEndpoint instance that is identified as follows:

- 663 1) The CIM\_DHCPProtocolEndpoint instance shall be associated with the CIM\_DHCPSettingData  
664 instance through an instance of CIM\_ElementSettingData.
- 665 2) The CIM\_DHCPProtocolEndpoint instance shall be associated through an instance of  
666 CIM\_SAPSAPDependency with the CIM\_IPProtocolEndpoint instance to which the aggregating  
667 CIM\_IPAssignmentSettingData is being applied.

668 When a pending IP configuration is applied, each instance of CIM\_DNSSettingData that is associated  
669 with the CIM\_IPAssignmentSettingData instance through an instance of CIM\_OrderedComponent shall  
670 be applied to the CIM\_DNSProtocolEndpoint instance that is identified as follows:

- 671 1) The CIM\_DNSProtocolEndpoint instance shall be associated with the CIM\_DNSSettingData  
672 instance through an instance of CIM\_ElementSettingData.
- 673 2) The CIM\_DNSProtocolEndpoint instance shall be associated through an instance of  
674 CIM\_SAPSAPDependency with the CIM\_IPProtocolEndpoint instance to which the aggregating  
675 CIM\_IPAssignmentSettingData is being applied.

### 676 7.5.3 Applying Static IP Settings

677 When an instance of CIM\_StaticIPAssignmentSettingData is applied to the CIM\_IPProtocolEndpoint  
678 instance, the values of the properties of the CIM\_IPProtocolEndpoint instance shall be the values of the  
679 properties of the CIM\_StaticIPAssignmentSettingData instance.

#### 680 7.5.3.1 CIM\_StaticIPAssignmentSettingData.GatewayIPv4Address

681 If the associated value of CIM\_IPProtocolEndpoint.ProtocolIFType = 4096 (IPv4), then the value of the  
682 AccessInfo property of the CIM\_RemoteServiceAccessPoint that represents the default gateway shall be  
683 the value of the CIM\_StaticIPAssignmentSettingData.GatewayIPv4Address property.

---

### 684 EXPERIMENTAL

685 If the associated value of CIM\_IPProtocolEndpoint.ProtocolIFType = 4098 (IPv4/IPv6), then the value of  
686 the AccessInfo property of the CIM\_RemoteServiceAccessPoint that represents the default IPv4 gateway  
687 shall be the value of the CIM\_StaticIPAssignmentSettingData.GatewayIPv4Address property.

#### 688 7.5.3.2 CIM\_StaticIPAssignmentSettingData.GatewayIPv6Address

689 If the associated value of CIM\_IPProtocolEndpoint.ProtocolIFType = 4097 (IPv6), then the value of the  
690 AccessInfo property of the CIM\_RemoteServiceAccessPoint that represents the default IPv6 gateway  
691 shall be the value of the CIM\_StaticIPAssignmentSettingData.GatewayIPv6Address property.

---

### 692 EXPERIMENTAL

### 693 7.5.3.3 Successful Application of Settings

694 An instance of CIM\_StaticIPAssignmentSettingData shall be considered successfully applied when the  
695 properties of the associated instance of CIM\_IPProtocolEndpoint to which the instance of  
696 CIM\_StaticIPAssignmentSettingData has been applied have the values of the relevant properties of the  
697 CIM\_StaticIPAssignmentSettingData instance.

### 698 7.5.4 Applying DHCP Settings

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

### 701 7.5.5 Applying DNS Settings

702 When a pending configuration includes DNS client configuration, the DNS configuration is applied as  
703 defined in the [DNS Client Profile](#). When the AssignedSequence property of the CIM\_OrderedComponent  
704 association that references an instance of CIM\_DNSSettingData or CIM\_DNSGeneralSettingData has a  
705 non-zero value, the referenced instance of CIM\_DNSSettingData or CIM\_DNSGeneralSettingData shall  
706 be applied, regardless of whether the application of a preceding CIM\_SettingData instance was  
707 successful.

### 708 7.5.6 Resolving Overlapped Settings

709 When more than one instance of CIM\_StaticIPAssignmentSettingData or CIM\_DHCPSettingData is  
710 associated with the same instance of CIM\_IPAssignmentSettingData, each CIM\_SettingData instance  
711 shall be applied in order (as described in section 7.4.3.3) until the implementation determines that the  
712 resultant configuration is valid. The amount of time an implementation waits after applying an instance of  
713 CIM\_SettingData before deciding whether the resultant configuration is valid is implementation specific  
714 and outside the scope of this specification. The criterion for determining whether a configuration that is  
715 represented by a specific CIM\_SettingData instance is valid is implementation specific and outside the  
716 scope of this specification.

## 717 7.6 Relationship with a Network Interface

718 An IP interface is generally bound to an underlying network interface. The underlying network interface  
719 might participate in a LAN and be modeled using the *Host LAN Network Port Profile* or a specialization  
720 thereof. When the underlying network interface is modeled with instrumentation compliant with the *Host  
721 LAN Network Port Profile*, an instance of CIM\_BindsToLANEndpoint shall associate the Central Instance  
722 of this profile with an instance of CIM\_LANEndpoint that is compliant with the *Host LAN Network Port  
723 Profile*.

## 724 8 Methods

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

### 727 8.1 CIM\_IPProtocolEndpoint.RequestStateChange()

728 Invocation of the RequestStateChange() method changes the element's state to the value specified in the  
729 RequestedState parameter. The 2 (Enabled) and 3 (Disabled) values of the RequestedState parameter  
730 shall correspond to enabling or disabling the IP network interface, respectively. A value of 11 (Reset)  
731 shall correspond to disabling and then enabling the IP interface.

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

733 No standard messages are defined.

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

736 **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 |

737 **Table 3 – CIM\_IPProtocolEndpoint.RequestStateChange() Method: Parameters**

| Qualifiers | Name           | Type                | Description/Values  |
|------------|----------------|---------------------|---|
| IN, REQ    | RequestedState | uint16              | Valid state values:<br>2 (Enabled)<br>3 (Disabled)<br>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:<br>0 or NULL – No time requirements<br><interval> – Maximum time allowed |

#### 738 **8.1.1.1 CIM\_IPProtocolEndpoint.RequestStateChange()—Conditional Support**

739 When an instance of CIM\_EnabledLogicalElementCapabilities is associated with the  
740 CIM\_IPProtocolEndpoint instance and the  
741 CIM\_EnabledLogicalElementCapabilities.RequestedStatesSupported property contains at least one  
742 value, the CIM\_IPProtocolEndpoint.RequestStateChange() method shall be implemented and supported.  
743 The CIM\_IPProtocolEndpoint.RequestStateChange() method shall not return a value of 1 (Not  
744 Supported).

#### 745 **8.2 CIM\_IPConfigurationService.ApplySettingToIPProtocolEndpoint()**

746 The CIM\_IPConfigurationService.ApplySettingToIPProtocolEndpoint() method is used to apply a  
747 configuration, as represented by an aggregating instance of CIM\_IPAssignmentSettingData, to an IP  
748 interface, as represented by an instance of CIM\_IPProtocolEndpoint. Implementation of this method is  
749 optional.

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

752 No standard messages are defined.



753  
754

**Table 4 – CIM\_IPConfigurationService.ApplySettingToIPProtocolEndpoint() Method: Return Code 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. |

755

**Table 5 – CIM\_IPConfigurationService.ApplySettingToIPProtocolEndpoint() Method: Parameters**

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

756  
757

The CIM\_IPConfigurationService.ApplySettingToIPProtocolEndpoint() method shall be implemented as follows:

758  
759  
760  
761

- The implementation shall validate that an instance of CIM\_ServiceAffectsElement references the CIM\_IPConfigurationService instance and the CIM\_IPProtocolEndpoint instance that is identified by the Endpoint parameter to the method. If the association does not exist, the return code of the method shall be 4 (Failed).

762  
763  
764  
765

- The implementation shall validate that an instance of CIM\_ElementSettingData associates the instance of CIM\_IPProtocolEndpoint that is identified by the Endpoint parameter with the instance of CIM\_IPAssignmentSettingData that is identified by the Configuration parameter. If the association does not exist, the return code of the method shall be 4 (Failed).

766  
767

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

768

### 8.3 Profile Conventions for Operations

769  
770  
771  
772  
773

Support for operations for each profile class (including associations) is specified in the following subclauses. Each subclause includes either the statement “All operations in the default list in section 8.3 are supported as described by [DSP0200 version 1.2](#)” or a table listing all of the operations that are not supported by this profile or where the profile requires behavior other than that described by [DSP0200 version 1.2](#).

774

The default list of operations is as follows:

775  
776  
777  
778  
779  
780

- GetInstance
- Associators
- AssociatorNames
- References
- ReferenceNames
- EnumerateInstances

- 781       • EnumerateInstanceNames

782 A compliant implementation shall support all the operations in the default list for each class, unless the  
783 “Requirement” column states something other than *Mandatory*.

#### 784 **8.4 CIM\_BindsToLANEndpoint**

785 Table 6 lists operations that either have special requirements beyond those from [DSP0200 version 1.2](#) or  
786 shall not be supported.

787 **Table 6 – Operations: CIM\_BindsToLANEndpoint**

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

#### 788 **8.5 CIM\_ElementSettingData**

789 Table 7 lists operations that either have special requirements beyond those from [DSP0200 version 1.2](#) or  
790 shall not be supported.

791 **Table 7 – Operations: CIM\_ElementSettingData**

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

##### 792 **8.5.1 CIM\_ElementSettingData—ModifyInstance**

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

##### 795 **8.5.1.1 CIM\_ElementSettingData Referencing CIM\_IPAssignmentSettingData**

796 When an instance of CIM\_ElementSettingData associates an instance of CIM\_IPAssignmentSettingData  
797 with an instance of CIM\_IPProtocolEndpoint, the following rules shall govern the behavior of the  
798 ModifyInstance operation:

- 799       • The ModifyInstance operation shall not allow the IsDefault property to be modified.
- 800       • The ModifyInstance operation shall not allow the IsCurrent property to be modified.
- 801       • When the ModifyInstance operation is used to set the IsNext property to a value of 1 (Is Next),  
802 the ModifyInstance operation shall implement the following behavior:
- 803       1) The ModifyInstance operation shall find all other instances of CIM\_ElementSettingData  
804 that associate an instance of CIM\_IPAssignmentSettingData with the instance of

- 805 CIM\_IPProtocolEndpoint referenced by the target instance of CIM\_ElementSettingData  
 806 where the IsNext property has a value of 1 (Is Next).
- 807 2) For each instance of CIM\_ElementSettingData found, the ModifyInstance operation shall  
 808 modify the value of its IsNext property to have a value of 2 (Is Not Next).
- 809 • When the IsNext property has a value of 1 (Is Next), the ModifyInstance operation shall not be  
 810 supported.
  - 811 • When the ModifyInstance operation is used to set the IsNext property to a value of 3 (Is Next for  
 812 Single Use), the ModifyInstance operation shall implement the following behavior:
    - 813 1) The ModifyInstance operation shall find all other instances of CIM\_ElementSettingData  
 814 that associate an instance of CIM\_IPAssignmentSettingData with the instance of  
 815 CIM\_IPProtocolEndpoint referenced by the target instance of CIM\_ElementSettingData  
 816 where the IsNext property has a value of 3 (Is Next For Single Use).
    - 817 2) For each instance of CIM\_ElementSettingData found, the ModifyInstance operation shall  
 818 modify the value of its IsNext property to have a value of 2 (Is Not Next).

819 **8.5.1.2 CIM\_ElementSettingData Referencing CIM\_StaticIPAssignmentSettingData**

820 When an instance of CIM\_ElementSettingData associates an instance of  
 821 CIM\_StaticIPAssignmentSettingData with an instance of CIM\_IPProtocolEndpoint, the ModifyInstance  
 822 operation shall not be supported.

823 **8.6 CIM\_HostedAccessPoint**

824 Table 8 lists operations that either have special requirements beyond those from [DSP0200 version 1.2](#) or  
 825 shall not be supported.

826 **Table 8 – Operations: CIM\_HostedAccessPoint**

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

827 **8.7 CIM\_HostedService**

828 Table 9 lists operations that either have special requirements beyond those from [DSP0200 version 1.2](#) or  
 829 shall not be supported.

830 **Table 9 – Operations: CIM\_HostedService**

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

## 831 **8.8 CIM\_IPAssignmentSettingData**

832 All operations in the default list in section 8.3 are supported as described by [DSP0200 version 1.2](#).

## 833 **8.9 CIM\_IPConfigurationService**

834 All operations in the default list in section 8.3 are supported as described by [DSP0200 version 1.2](#).

## 835 **8.10 CIM\_IPProtocolEndpoint**

836 Table 10 lists operations that either have special requirements beyond those from [DSP0200 version 1.2](#)  
837 or shall not be supported.

838 **Table 10 – Operations: CIM\_IPProtocolEndpoint**

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

### 839 **8.10.1 CIM\_IPProtocolEndpoint—ModifyInstance Operation**

840 This section details the specific requirements for the ModifyInstance operation applied to an instance of  
841 CIM\_IPProtocolEndpoint.

#### 842 **8.10.1.1 CIM\_IPProtocolEndpoint.ElementName Property**

843 When an instance of CIM\_EnabledLogicalElementCapabilities is associated with the  
844 CIM\_IPProtocolEndpoint instance and the  
845 CIM\_EnabledLogicalElementCapabilities.ElementNameEditSupported property has a value of TRUE, the  
846 implementation shall allow the ModifyInstance operation to change the value of the ElementName  
847 property of the CIM\_IPProtocolEndpoint instance. The ModifyInstance operation shall enforce the length  
848 restriction specified in the MaxElementNameLen property of the instance of  
849 CIM\_EnabledLogicalElementCapabilities.

850 When no instance of CIM\_EnabledLogicalElementCapabilities is associated with the  
851 CIM\_IPProtocolEndpoint instance, or the ElementNameEditSupported property of the  
852 CIM\_EnabledLogicalElementCapabilities instance has a value of FALSE, the implementation shall not  
853 allow the ModifyInstance operation to change the value of the ElementName property of the  
854 CIM\_IPProtocolEndpoint instance.

855 **8.11 CIM\_OrderedComponent**

856 Table 11 lists operations that either have special requirements beyond those from [DSP0200 version 1.2](#)  
 857 or shall not be supported.

858 **Table 11 – Operations: CIM\_OrderedComponent**

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

859 **8.11.1 CIM\_OrderedComponent—ModifyInstance**

860 The ModifyInstance operation may be supported for CIM\_OrderedComponent. When an instance of  
 861 CIM\_OrderedComponent references an instance of CIM\_DNSSettingData or an instance of  
 862 CIM\_DNSGeneralSettingData, the AssignedSequence property may be modified. When an instance of  
 863 CIM\_OrderedComponent references an instance of CIM\_StaticIPAssignmentSettingData or an instance of  
 864 CIM\_DHCPSettingData, the AssignedSequence property shall not be modified.

865 **8.12 CIM\_RemoteAccessAvailableToElement**

866 Table 12 lists operations that either have special requirements beyond those from [DSP0200 version 1.2](#)  
 867 or shall not be supported.

868 **Table 12 – Operations: CIM\_RemoteAccessAvailableToElement**

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

869 **8.13 CIM\_RemoteServiceAccessPoint**

870 All operations in the default list in section 8.3 are supported as described by [DSP0200 version 1.2](#).

871 **8.14 CIM\_ServiceAffectsElement**

872 Table 13 lists operations that either have special requirements beyond those from [DSP0200 version 1.2](#)  
 873 or shall not be supported.

874 **Table 13 – Operations: CIM\_ServiceAffectsElement**

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

875 **8.15 CIM\_StaticIPAssignmentSettingData**

876 Table 14 lists operations that either have special requirements beyond those from [DSP0200 version 1.2](#)  
 877 or shall not be supported.

878 **Table 14 – Operations: CIM\_StaticIPAssignmentSettingData**

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

879 **8.16 CIM\_SystemDevice**

880 Table 15 lists operations that either have special requirements beyond those from [DSP0200 version 1.2](#)  
 881 or shall not be supported.

882 **Table 15 – Operations: CIM\_SystemDevice**

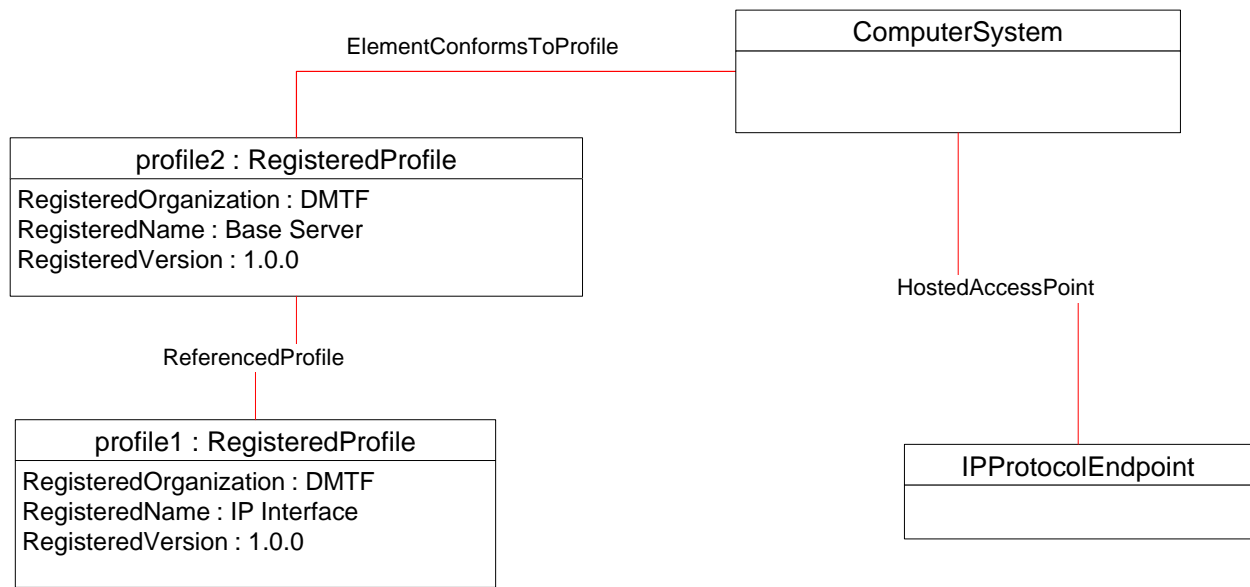
| Operation       | Requirement | Messages |
|-----------------|-------------|----------|
| Associators     | Unspecified | None     |
| AssociatorNames | Unspecified | None     |
| References      | Unspecified | None     |
| ReferenceNames  | Unspecified | None     |

883 **9 Use Cases**

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

885 **9.1 Miscellaneous Object Diagrams**

886 The object diagram in Figure 2 shows one possible method for advertising profile conformance. The  
 887 instances of CIM\_RegisteredProfile are used to identify the version of the *IP Interface Profile* with which  
 888 an instance of CIM\_IPProtocolEndpoint and its associated instances are conformant. An instance of  
 889 CIM\_RegisteredProfile exists for each profile that is instrumented in the system. One instance of  
 890 CIM\_RegisteredProfile identifies the "DMTF Base Server Profile version 1.0.0". The other instance  
 891 identifies the "DMTF IP Interface Profile version 1.0.0". The CIM\_IPProtocolEndpoint instance is scoped  
 892 to an instance of CIM\_ComputerSystem. This instance of CIM\_ComputerSystem is conformant with the  
 893 DMTF *Base Server Profile* version 1.0.0 as indicated by the CIM\_ElementConformsToProfile association  
 894 to the CIM\_RegisteredProfile instance.

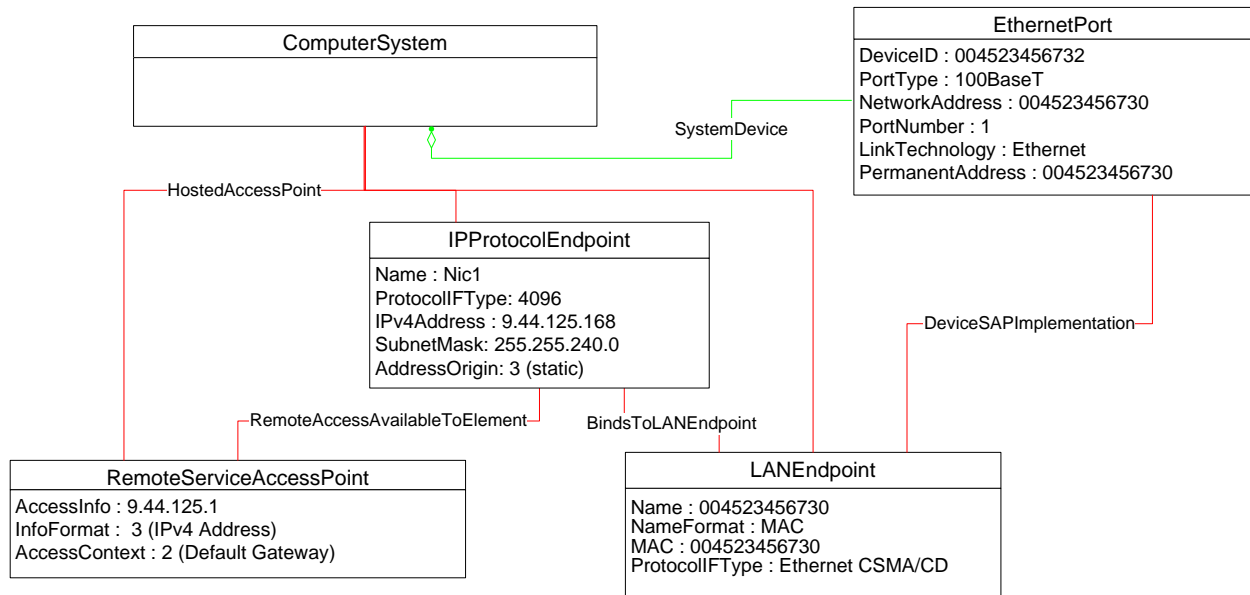


895

896 **Figure 2 – Registered Profile**

897 The object diagram shown in Figure 3 contains the basic elements used to model the current  
 898 configuration of an IP interface when the CIM\_IPProtocolEndpoint.ProtocolIFType is 4096 (Ipv4). The IP  
 899 interface is bound to an Ethernet NIC, as illustrated by the CIM\_BindsToLANEndpoint association  
 900 between the CIM\_IPProtocolEndpoint instance and the CIM\_LANEndpoint instance. The AddressOrigin  
 901 property of the CIM\_IPProtocolEndpoint has a value of "static", indicating that the configuration was  
 902 statically assigned. In this diagram, the *Ethernet Port Profile* and *IP Interface Profile* have been  
 903 implemented.

904 The default gateway used by the IP interface is represented by the instance of  
 905 CIM\_RemoteServiceAccessPoint that is associated with the CIM\_IPProtocolEndpoint instance through an  
 906 instance of CIM\_RemoteAccessAvailableToElement.



907

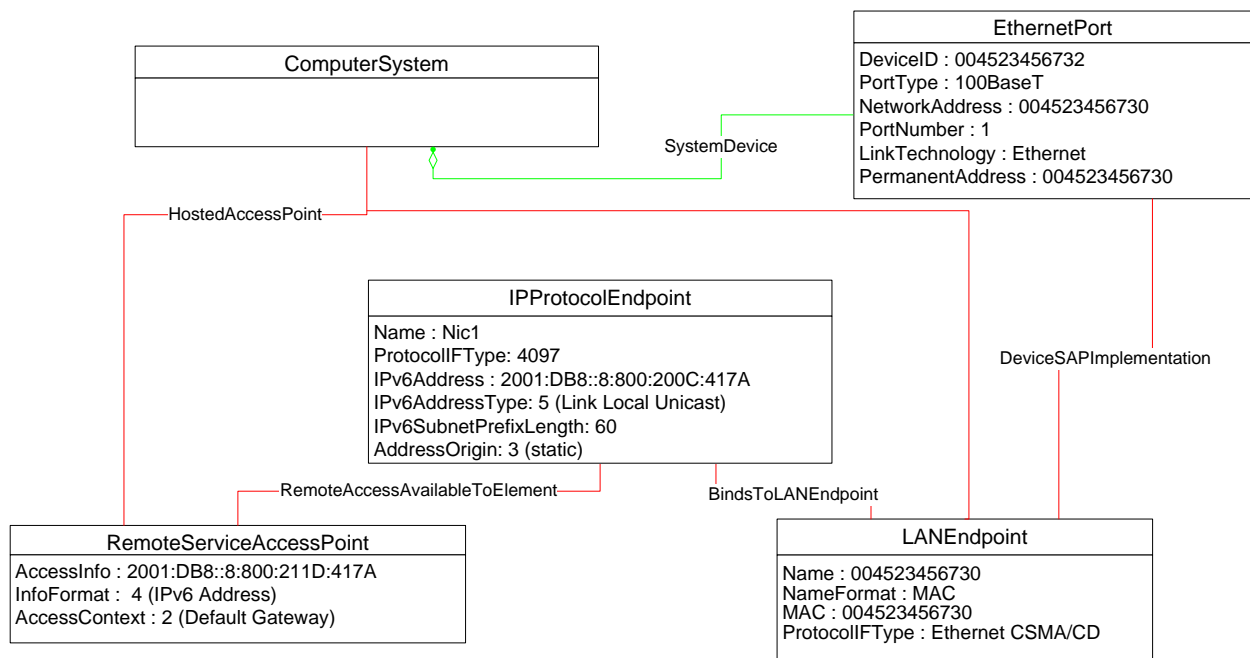
908

Figure 3 – Basic Configuration – IPv4

909 **EXPERIMENTAL**

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

914



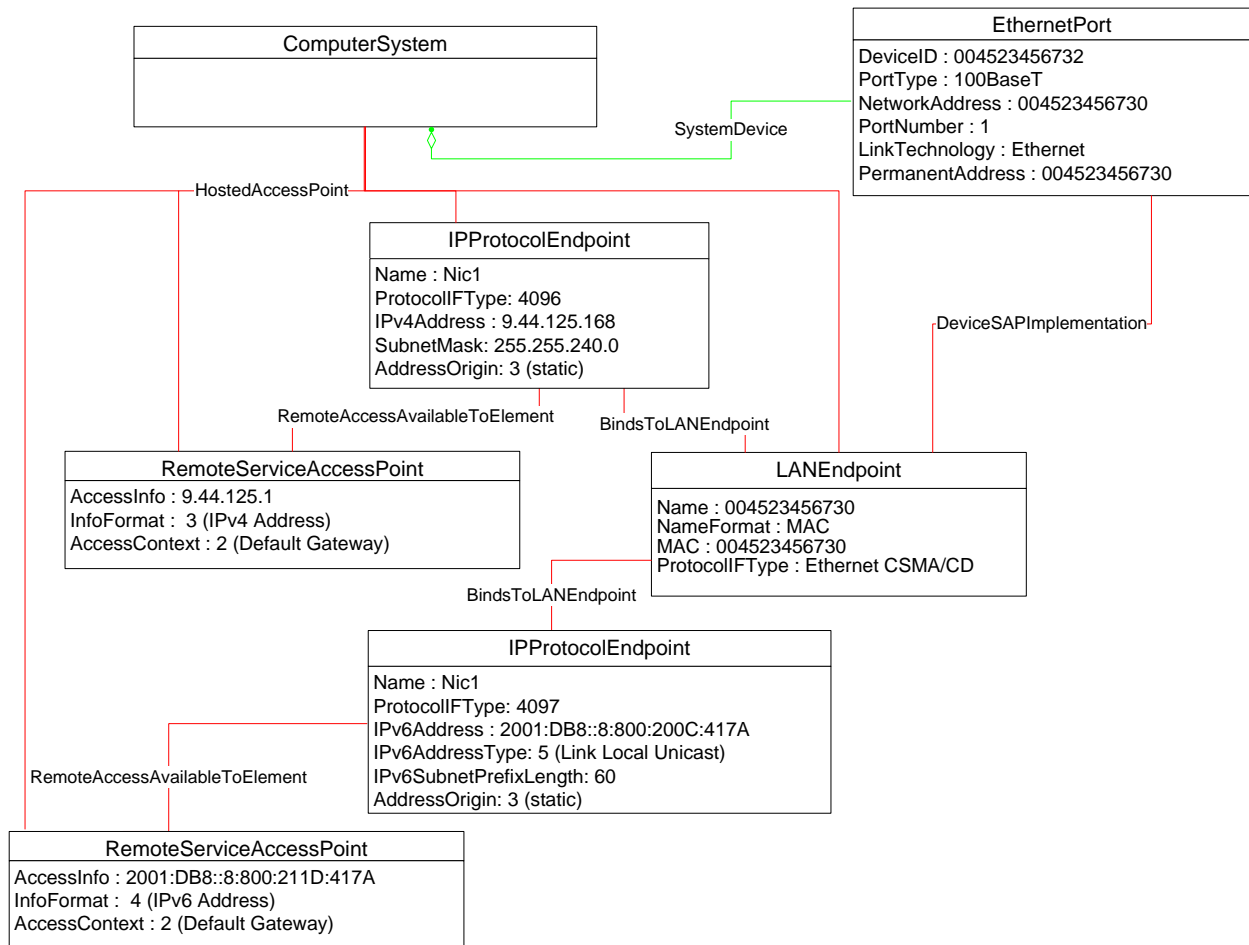
915

916

Figure 4 – Basic Configuration – IPv6



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



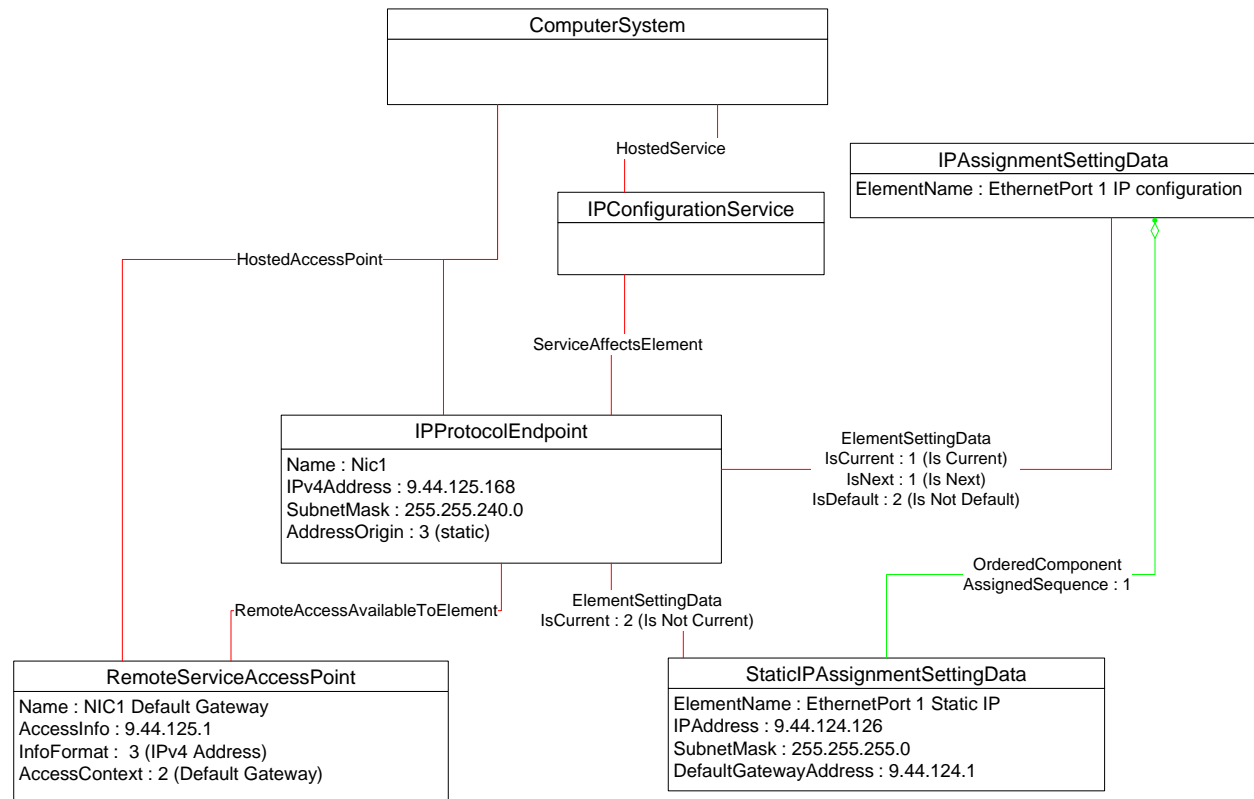
921

922

**Figure 5 – Basic Configuration – IPv4 and IPv6**

923 **EXPERIMENTAL**

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



930

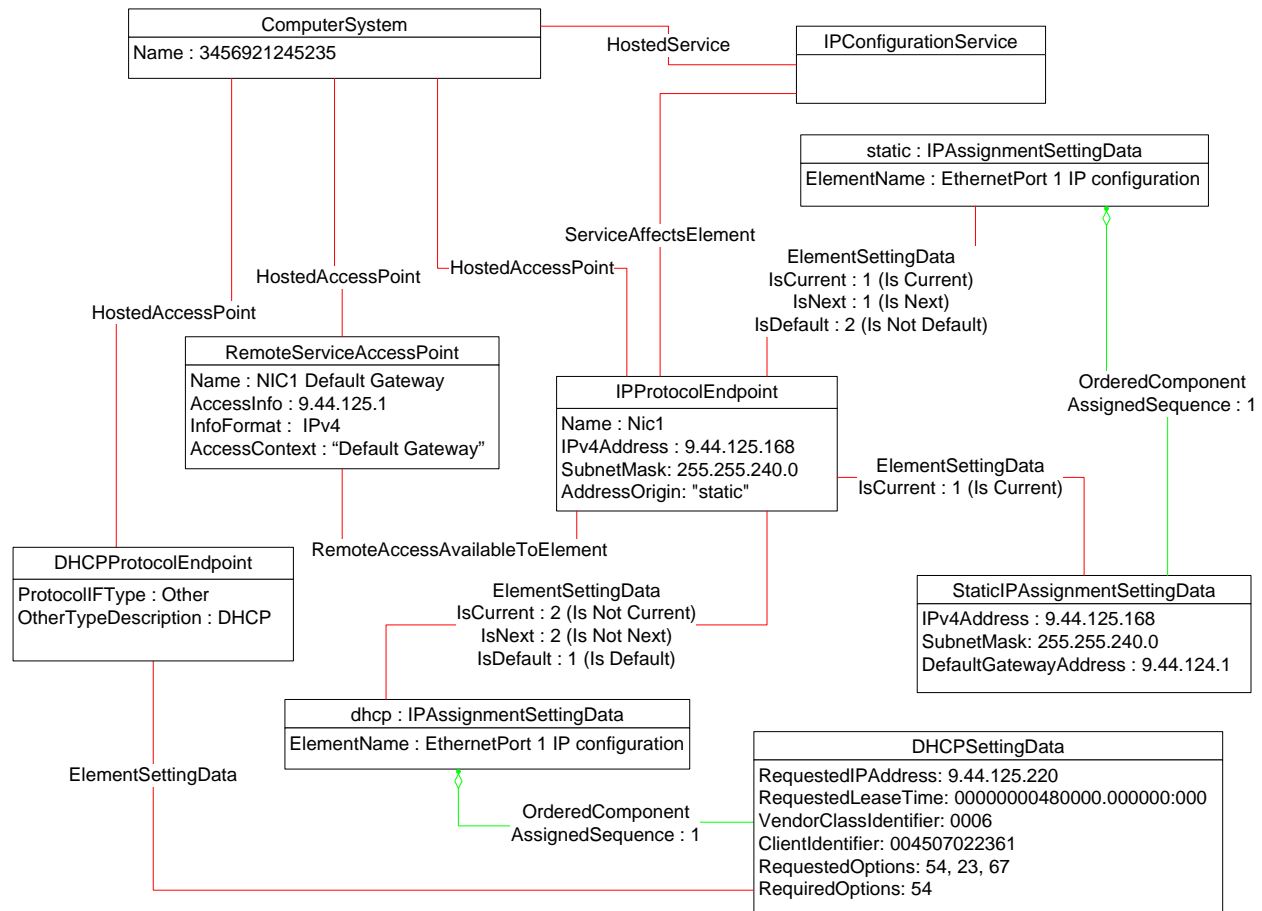
931

**Figure 6 – Static Current and Pending Configuration**

932 The object diagram in Figure 7 illustrates an IP interface with two supported alternate configurations. Two  
 933 discrete IP configuration options are available for the IP interface. Each option is represented by an  
 934 instance of CIM\_IPAssignmentSettingData. One configuration option represents the ability to statically  
 935 assign the IP configuration. This option is indicated by the instance of CIM\_OrderedComponent that  
 936 associates the CIM\_IPAssignmentSettingData instance with an instance of  
 937 CIM\_StaticIPAssignmentSettingData. The other configuration option is to obtain the configuration through  
 938 a DHCP client. This option is indicated by the instance of CIM\_OrderedComponent that associates the  
 939 CIM\_IPAssignmentSettingData with an instance of CIM\_DHCPSettingData.

940 In this example, each configuration option consists of a single instance of a subclass of  
 941 CIM\_IPAssignmentSettingData. Therefore, the value of the AssignedSequence property of the  
 942 CIM\_OrderedComponent instances is irrelevant.

943 The default configuration is to attempt to obtain a configuration through DHCP. This default is indicated  
 944 by the IsDefault property having a value of 1 (Is Default) on the CIM\_ElementSettingData instance that  
 945 associates the CIM\_IPAssignmentSettingData instance with the CIM\_IPProtocolEndpoint instance.  
 946 However, the current configuration of the IP interface was statically assigned using the configuration  
 947 identified by the CIM\_IPAssignmentSettingData instance *static*. This configuration is indicated by the  
 948 value of the IsCurrent property on the instance of CIM\_ElementSettingData that associates the  
 949 CIM\_IPAssignmentSettingData instance *static* with the CIM\_IPProtocolEndpoint instance, and by the  
 950 value of the AddressOrigin property on the CIM\_IPProtocolEndpoint instance. When the interface is  
 951 restarted, the static configuration will be used again for the IP interface. This behavior is indicated by the  
 952 value of the IsNext property on the instance of CIM\_ElementSettingData that associates the  
 953 CIM\_IPAssignmentSettingData instance *static* to the CIM\_IPProtocolEndpoint instance.



954

955

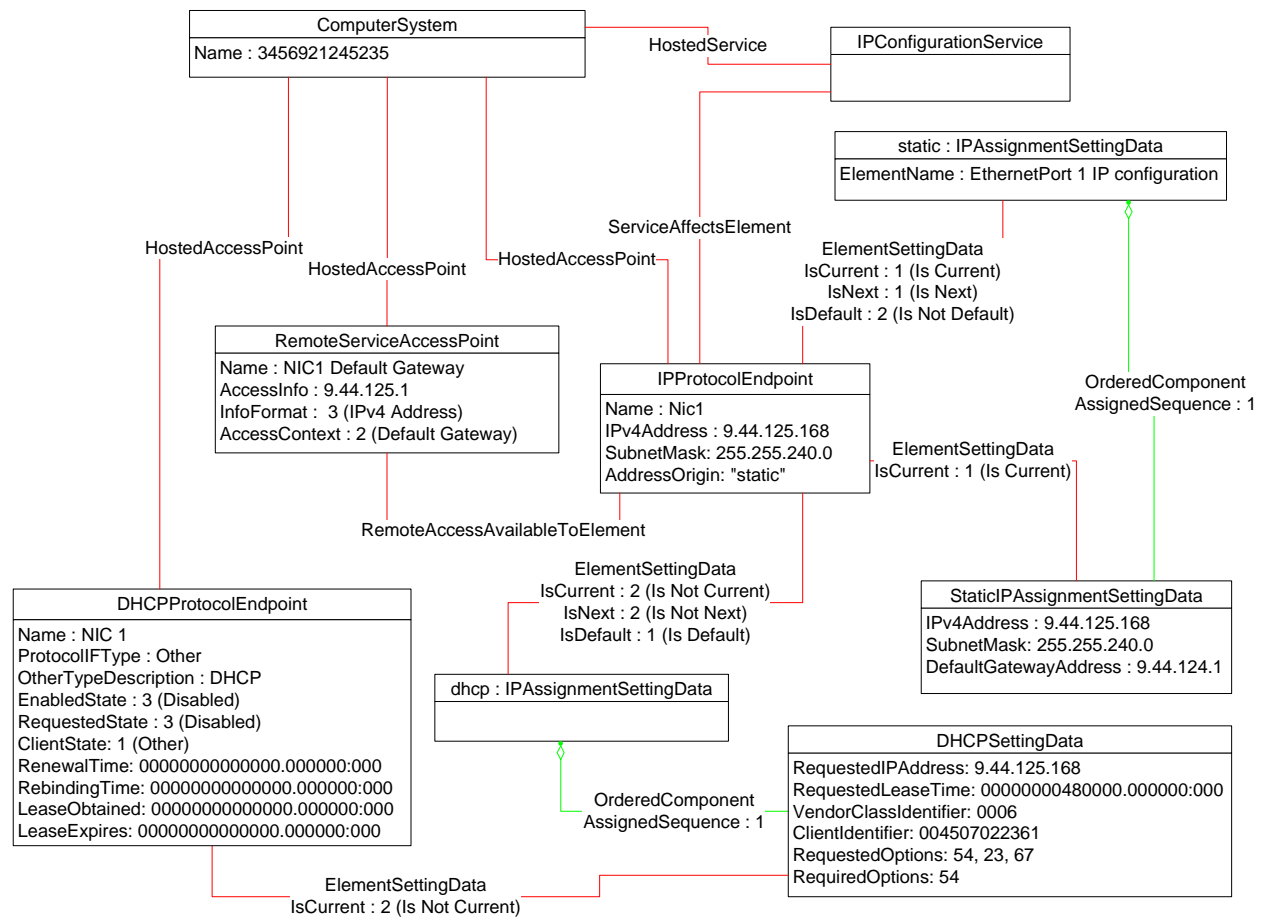
**Figure 7 – Static and DHCP Pending Configurations**

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

958 The EnabledState and ClientState properties of the CIM\_DHCPProtocolEndpoint instance indicate that  
 959 the DHCP client is not disabled but neither is it actively attempting to obtain a configuration any longer.  
 960 No instance of CIM\_RemoteServiceAccessPoint is associated with the CIM\_DHCPProtocolEndpoint  
 961 instance because the DHCP client failed to communicate with a DHCP server.

962 The AddressOrigin property of the CIM\_IPProtocolEndpoint instance reflects that the address was  
 963 assigned statically.

964

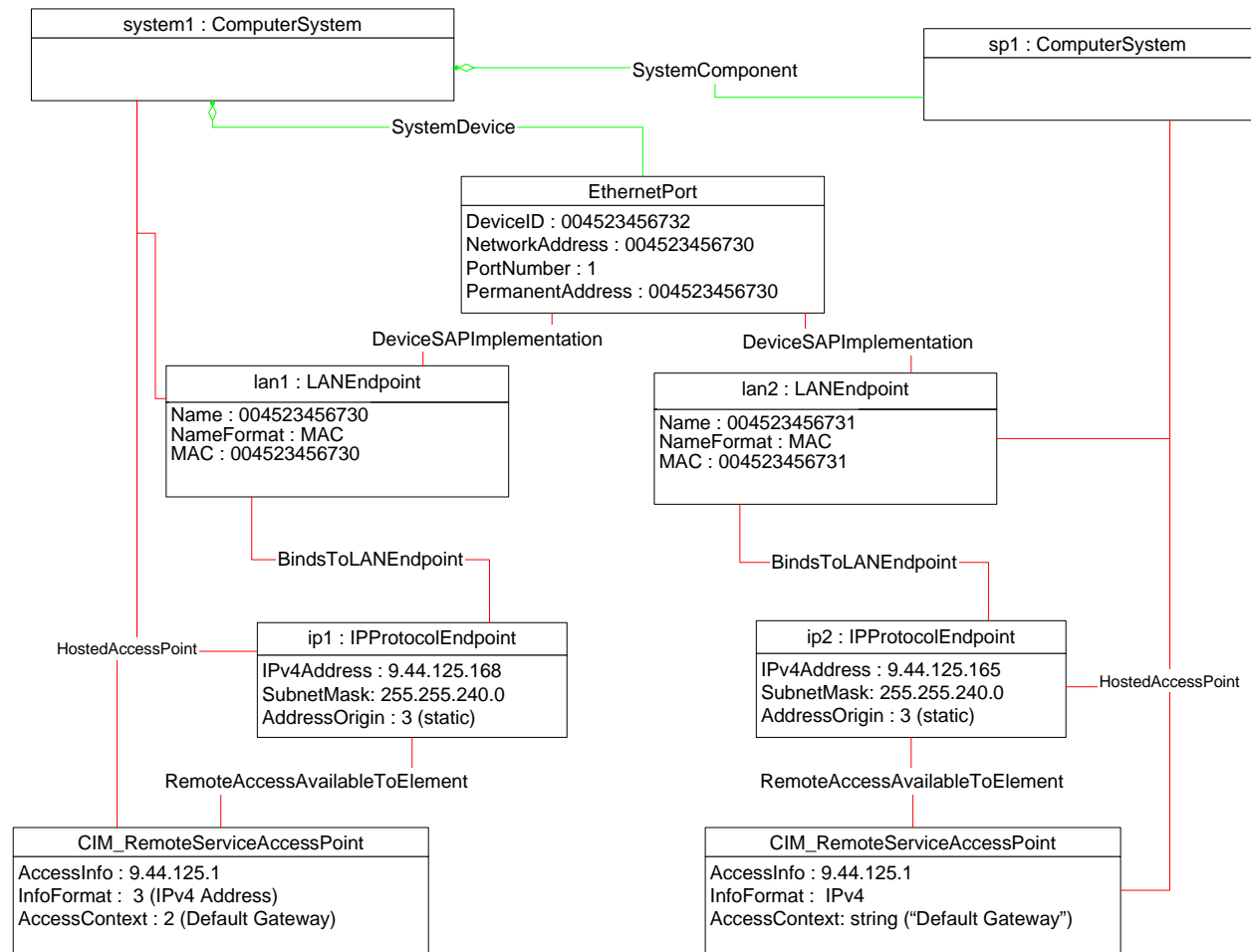


965

966

**Figure 8 – DHCP Timed Out to a Static Configuration**

967 The object diagram in Figure 9 illustrates a configuration in which a system contains an integrated service  
 968 processor and they share the network interface of the system. The CIM\_EthernetPort instance is  
 969 associated with the system1 instance, which indicates that the network device is owned by the server.  
 970 The MAC property of the lan1 instance matches the PermanentAddress property of the CIM\_EthernetPort  
 971 instance, which indicates that the server is using the hardware MAC. The MAC property of the lan2  
 972 instance is different, which indicates that the service processor has been assigned a logical MAC. The  
 973 system and service processor each have a unique IP interface that has been statically configured.



974

**Figure 9 – Service Processor and Server Share an NIC**

975

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

981 Note that in the following figures extraneous classes that are not relevant to the point being illustrated are  
 982 not shown. For example, the CIM\_HostedAccessPoint associations are never included.

983 The object diagram in Figure 10 outlines the alternate configurations presented by the instrumentation  
 984 layer for the system. Three alternate configurations are shown: static\_only, dhcp\_only, and dhcp\_static.

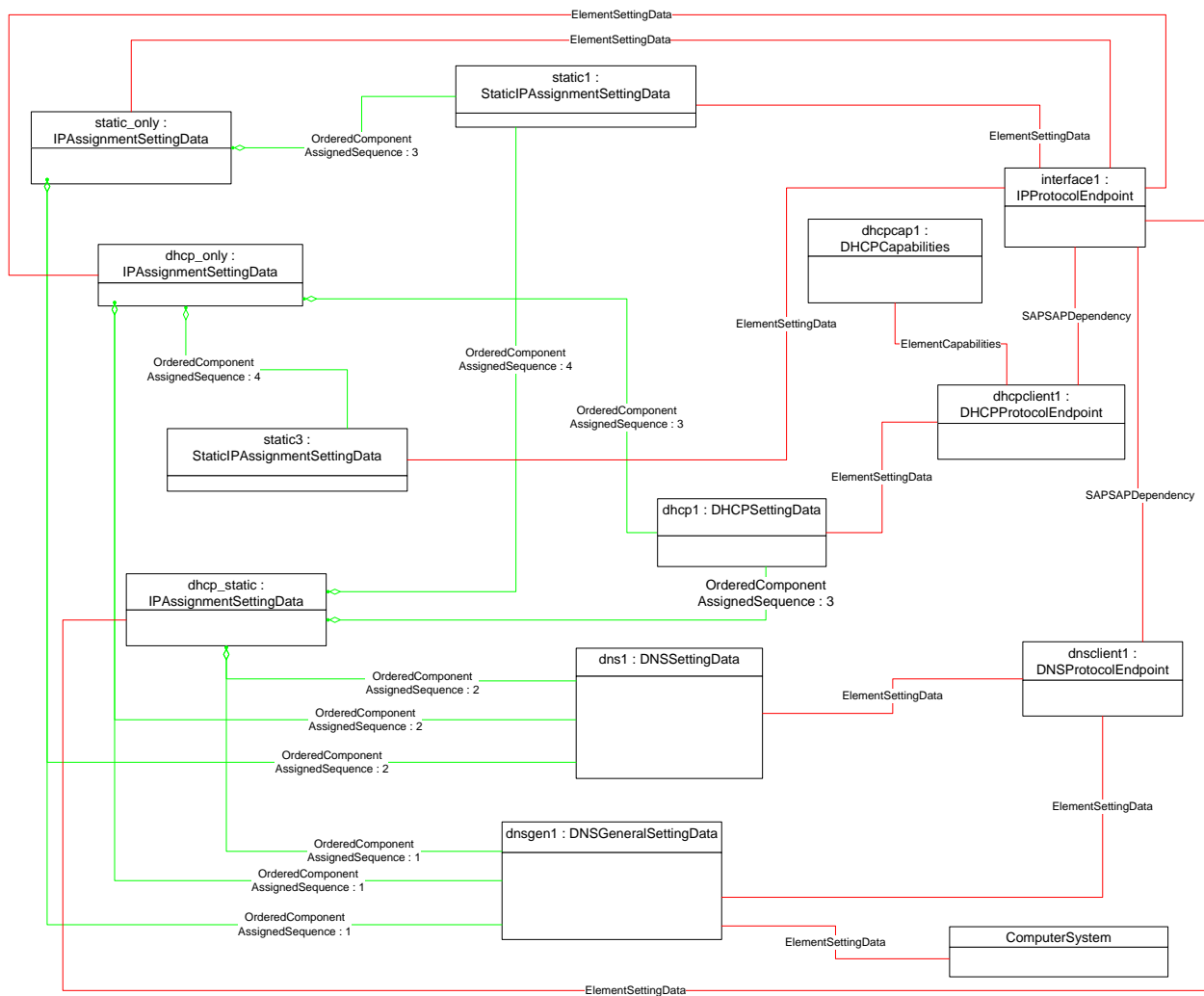
985 The system persists a single underlying static IP configuration, which is represented by static1. When the  
 986 configuration selected is static only or DHCP and then static, the same client static IP configuration is  
 987 used.

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

989 static\_only represents a configuration that uses static assignment of the IP configuration, including  
 990 support for static configuration of the DNS client. This behavior is indicated by the aggregated instances:  
 991 static1, dns1, and dnsgen1.

992 dhcp\_only represents a configuration that uses DHCP to obtain the IP configuration. This behavior is  
 993 indicated by the aggregated instance dhcp1. The DNS configuration can be assigned through DHCP or  
 994 statically assigned. This behavior is indicated by the aggregated instances dns1 and dnsgen1. In the  
 995 event the DHCP client is unable to obtain a configuration, the system is implemented to default to a hard-  
 996 coded, well-known default static IP configuration. The existence of a default configuration is indicated by  
 997 the aggregated instance static3. Note that no advertisement mechanism is specified in the profile to  
 998 indicate that static3 represents hard-coded values that cannot be modified by the client. If the system  
 999 were implemented such that the DHCP client would be continually in use without a timeout to a static  
 1000 configuration, the aggregated instance static3 would not exist.

1001 dhcp\_static represents a configuration that attempts to use DHCP to obtain an IP configuration. In the  
 1002 event the DHCP client fails to obtain a configuration, the system defaults to a client-assigned static IP  
 1003 configuration. This behavior is indicated by the instances dhcp1 and static1 and the relative values of the  
 1004 AssignedSequence property of the instances of CIM\_OrderedComponent, which aggregate them into  
 1005 dhcp\_static.



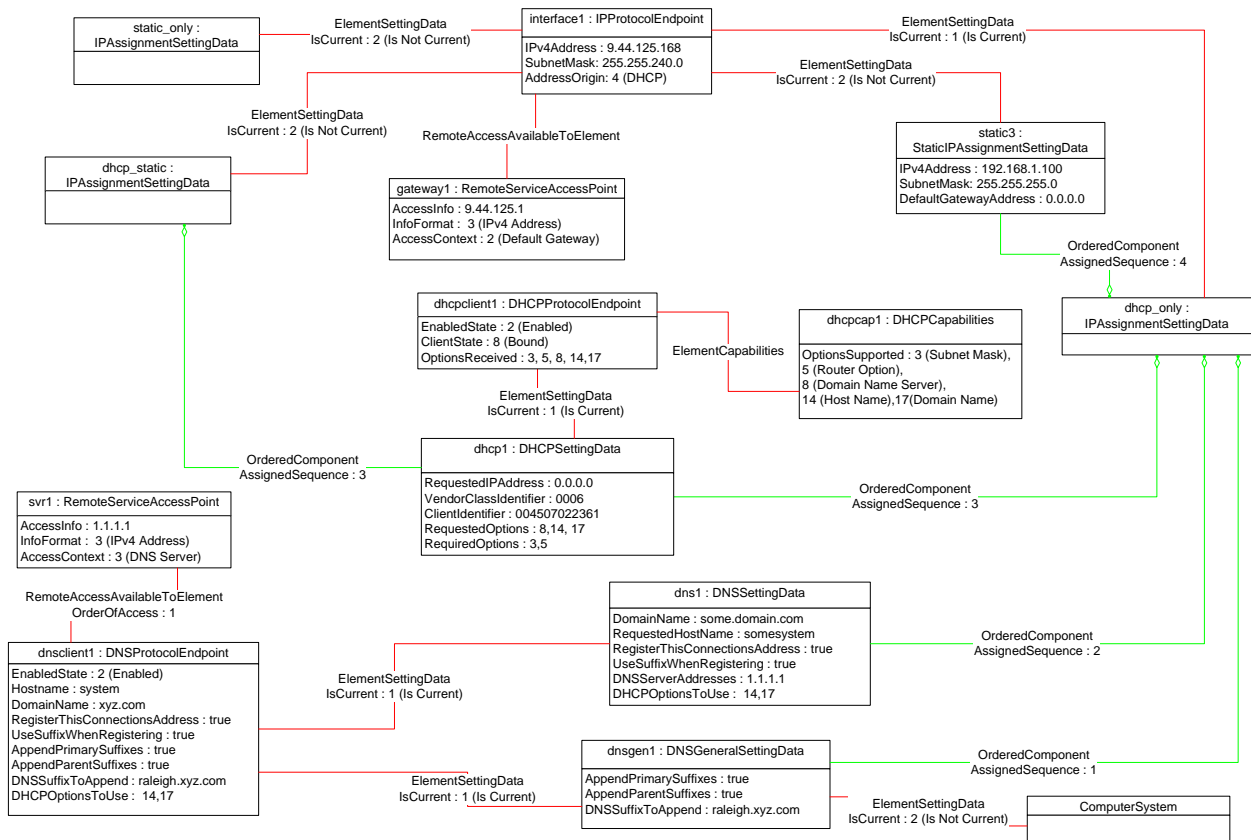
1006

1007

Figure 10 – Configuration Choices

1008 The object diagram in Figure 11 reflects the system when the DHCP configuration method has been used  
 1009 and the DNS configuration has partially been assigned through DHCP and partially statically configured.

1010 The use of the DHCP-only configuration is indicated by the IsCurrent property of the instance of  
 1011 CIM\_ElementSettingData that associates dhcp\_only to interface1 having the value 1 (Is Current). The  
 1012 DHCP configuration includes DHCP options that affect the DNS configuration. The DHCP options 8, 14,  
 1013 and 17 are requested as indicated by the RequestedOptions property of dhcp1. Each of these options  
 1014 was in turn received by the DHCP client, which is indicated by the value of the OptionsReceived property  
 1015 of dhcpclient1. The DNS client has been configured to use the values received for options 14 and 17 as  
 1016 indicated by the presence of these values in the DHCPOptionsToUse property of dnsclient1. The  
 1017 properties on dnsclient1 reflect the current DNS client configuration. Note that the actual current  
 1018 configuration does not directly reflect the configuration indicated by dns1 and dnsngen1. The two  
 1019 properties for which values were supplied by the DHCP options instead reflect the values assigned by the  
 1020 DHCP server.



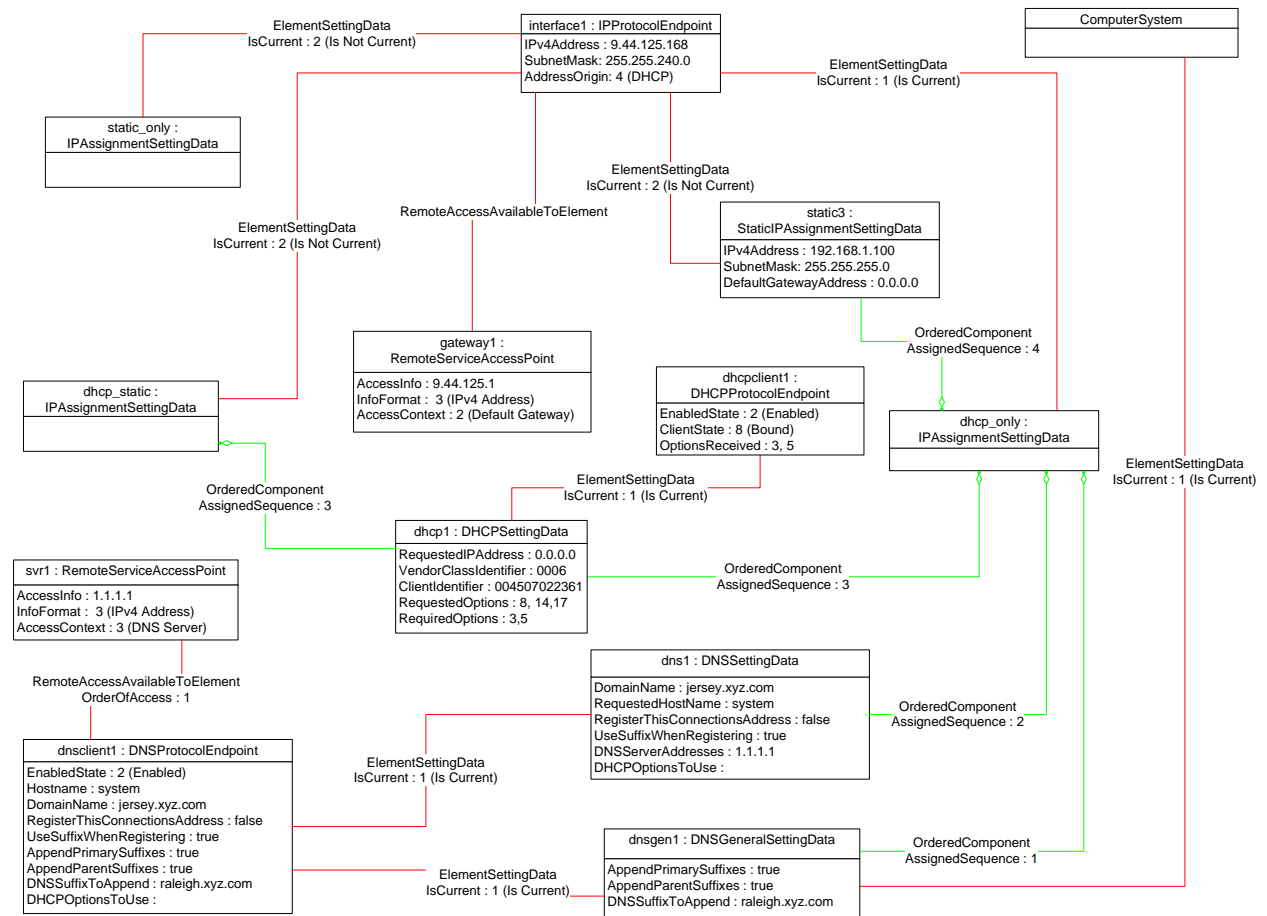
1021

Figure 11 – DHCP Assigned Partial DNS

1022

1023 The object diagram in Figure 12 reflects the system when the DHCP configuration method has been used  
 1024 and the DNS configuration has been statically configured.

1025 The use of the DHCP-only configuration is indicated by the IsCurrent property of the instance of  
 1026 CIM\_ElementSettingData that associates dhcp\_only to interface1 having the value 1 (Is Current).  
 1027 Although the DHCP configuration includes DHCP options that affect the DNS configuration, the values  
 1028 returned are not being used by the DNS client. This behavior is indicated by the absence of any values in  
 1029 the DHCPOptionsToUse property of dnsclient1. The actual current configuration directly reflects the  
 1030 configuration indicated by dns1 and dnsngen1 because no DHCP options are selected for use.



1031

1032

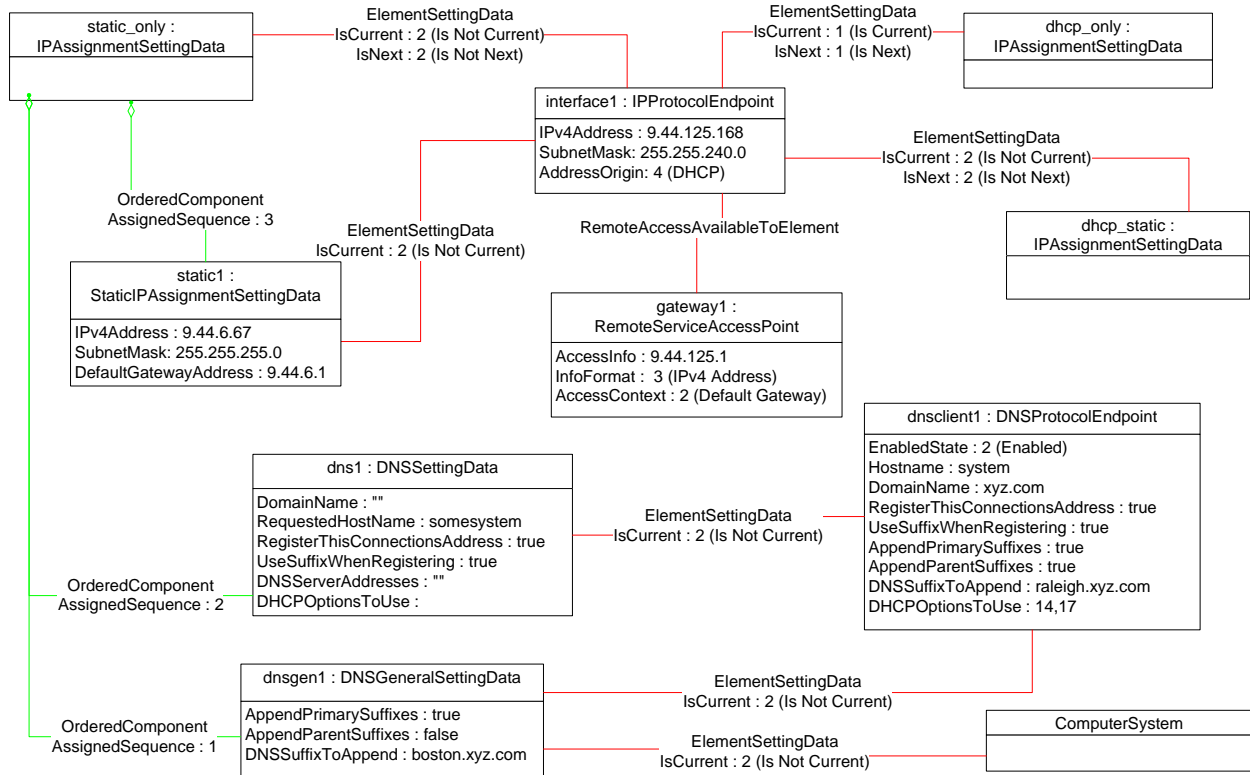
Figure 12 – DHCP with DNS Statically Configured

1033 **9.1.1 Sequence for Disabled DNS Client**

1034 The following three object diagrams illustrate the system when a client is configuring it to use a static IP  
 1035 configuration with the DNS client disabled. The client first modifies the pending static configuration so that  
 1036 the DNS settings will not be applied. Then it disables the DNS client directly. Finally, it applies the static  
 1037 configuration.

1038 The object diagram in Figure 13 illustrates the state of the system before the client begins modifying it to  
 1039 use a static IP configuration with DNS disabled. The last configuration applied was the DHCP-only  
 1040 configuration, which is indicated by the value of the IsCurrent property of the CIM\_ElementSettingData  
 1041 instance that references dhcp\_only and interface1. The static\_only configuration has not yet been  
 1042 modified by the client. As shown, the alternate DNS configuration represented by dns1 and dnsngen1  
 1043 would be applied if static\_only were applied to interface1.





1044

1045

**Figure 13 – Static without DNS Configuration – One**

1046

1047

1048

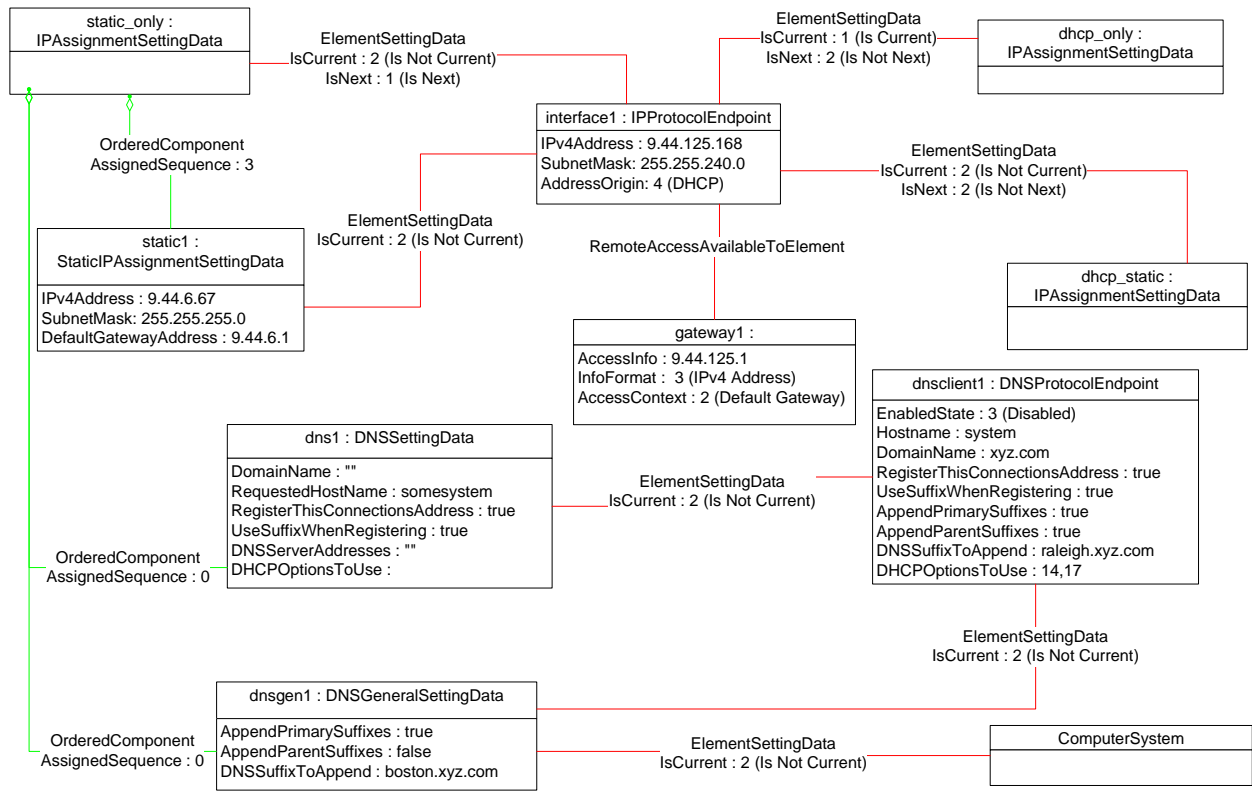
1049

1050

1051

1052

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`.



1053

1054

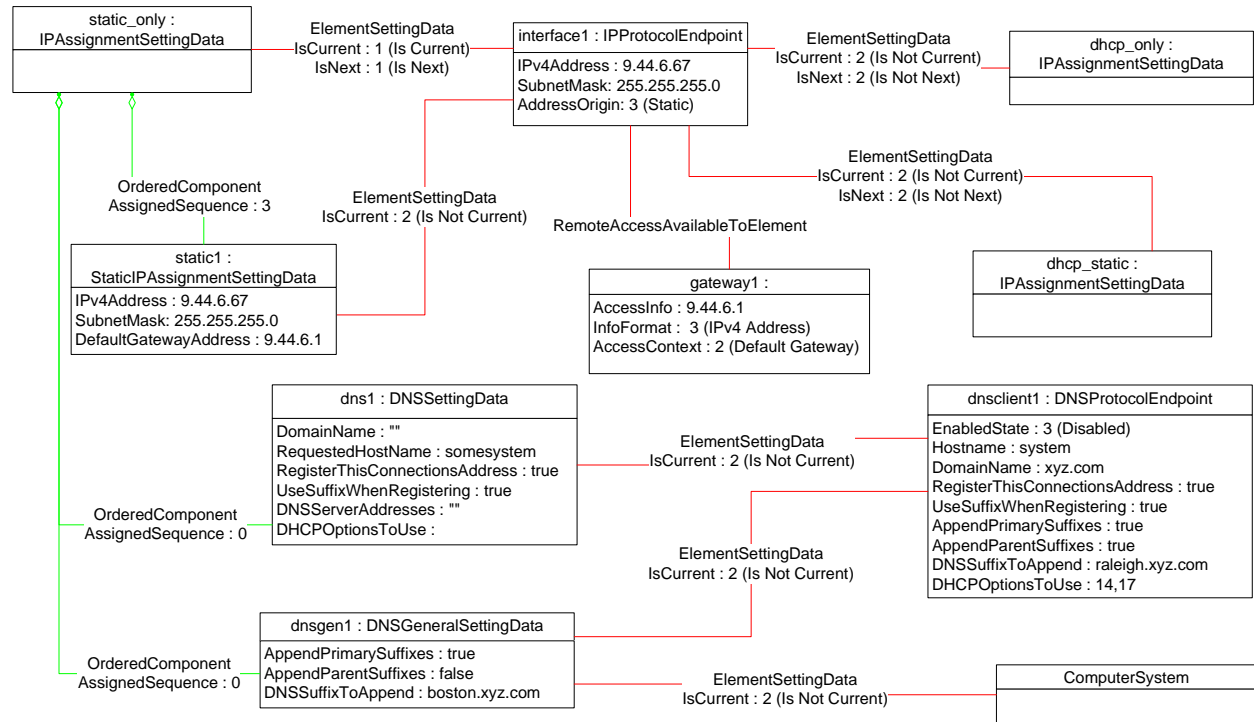
**Figure 14 – Static without DNS Configuration – Two**

1055

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.

1056

1057



1058

1059

Figure 15 – Static without DNS Configuration – Three

1060 **9.2 Determine Supported Configuration Methods**

1061 A client can determine which configuration methods are supported for a given interface as follows:

- 1062 1) Find all instances of CIM\_IPAssignmentSettingData that are associated with the
- 1063 CIM\_IPProtocolEndpoint instance.
- 1064 2) For each instance of CIM\_IPAssignmentSettingData:
- 1065 • Find all instances of subclasses of CIM\_IPAssignmentSettingData that are associated with
  - 1066 the CIM\_IPAssignmentSettingData instance through an instance of
  - 1067 CIM\_OrderedComponent.
  - 1068 • Query the value of the AddressOrigin property to determine the supported identified
  - 1069 configuration method.

1070 **9.3 Determine Gateway Address**

1071 A client can find the default gateway in use for an IP interface as follows:

- 1072 1) Find all instances of CIM\_RemoteServiceAccessPoint that are associated with the
- 1073 CIM\_IPProtocolEndpoint instance through an instance of
- 1074 CIM\_RemoteAccessAvailableToElement.
- 1075 2) For each instance of CIM\_RemoteServiceAccessPoint, determine if the value of the
- 1076 AccessContext property is "Default Gateway". If so, query the value of the AccessInfo property.

1077 **9.4 Determine Method Used for Current Configuration**

1078 A client can determine the method by which the IP configuration was assigned by querying the

1079 AddressOrigin property of the CIM\_IPProtocolEndpoint instance.

## 1080 9.5 Determine Whether DHCP Then Static Is Supported

1081 An implementation may support attempting to acquire its IP configuration through a DHCP client and  
1082 defaulting to a static configuration if the client fails to acquire a configuration from a DHCP server. A client  
1083 can determine whether this functionality is supported as follows:

- 1084 1) Find all instances of CIM\_IPAssignmentSettingData (the parent class and not subclasses) that  
1085 are associated with the CIM\_IPProtocolEndpoint instance.
- 1086 2) For each instance of CIM\_IPAssignmentSettingData:
  - 1087 a) Find all instances of CIM\_DHCPSettingData that are associated through an instance of  
1088 CIM\_OrderedComponent.
  - 1089 b) Find all instances of CIM\_StaticIPAssignmentSettingData that are associated through an  
1090 instance of CIM\_OrderedComponent.
- 1091 3) Determine if there is an instance of CIM\_DHCPSettingData such that the value of the  
1092 AssignedSequence property of the CIM\_OrderedComponent that associates the instance of  
1093 CIM\_DHCPSettingData with the instance of CIM\_IPAssignmentSettingData is less than the  
1094 value of the AssignedSequence property of an instance of CIM\_OrderedComponent that  
1095 associates the CIM\_StaticIPAssignmentSettingData with the instance of  
1096 CIM\_IPAssignmentSettingData. If so, DHCP then static is supported.

## 1097 9.6 View Default Configuration

1098 A client can view the default configuration for an IP interface as follows:

- 1099 1) Find all instances of CIM\_ElementSettingData that associate an instance of  
1100 CIM\_IPAssignmentSettingData (the parent class and not subclasses) with the  
1101 CIM\_IPProtocolEndpoint instance.
- 1102 2) For each instance of CIM\_ElementSettingData, see if the value of the IsDefault property is 1 (Is  
1103 Default).

## 1104 9.7 Configure the Interface to Use DHCP

1105 An implementation may support attempting to acquire its IP configuration through a DHCP client. A client  
1106 can determine whether this functionality is supported and configure the interface to use it as follows:

- 1107 1) Find all instances of CIM\_IPAssignmentSettingData (the parent class and not subclasses) that  
1108 are associated with the CIM\_IPProtocolEndpoint instance.
- 1109 2) For each instance of CIM\_IPAssignmentSettingData:
  - 1110 a) Find an instance of CIM\_DHCPSettingData that is associated through an instance of  
1111 CIM\_OrderedComponent.
  - 1112 b) Verify that no instances of CIM\_StaticIPAssignmentSettingData are associated with the  
1113 instance of CIM\_IPAssignmentSettingData.

1114 This instance of CIM\_IPAssignmentSettingData represents a DHCP configuration.

- 1115 3) Find an instance of CIM\_IPConfigurationService that is associated with the  
1116 CIM\_IPProtocolEndpoint instance through an instance of CIM\_ServiceAffectsElement.
- 1117 4) Invoke the ApplySettingToIPProtocolEndpoint() method of the CIM\_IPConfigurationService  
1118 instance, specifying the instances of CIM\_IPProtocolEndpoint and  
1119 CIM\_IPAssignmentSettingData.

## 1120 9.8 Establish a Static IP Configuration for an Interface

1121 A client can manually assign an IP configuration to an interface 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 an instance of CIM\_StaticIPAssignmentSettingData that is associated through an  
1126 instance of CIM\_OrderedComponent.
  - 1127 b) Verify that no other instances of CIM\_StaticIPAssignmentSettingData or instances of  
1128 CIM\_DHCPSettingData are associated with the instance of CIM\_IPAssignmentSettingData  
1129 through an instance of CIM\_OrderedComponent.
  - 1130 c) For the instance of CIM\_ElementSettingData that associates the  
1131 CIM\_IPAssignmentSettingData instance with the instance of CIM\_IPProtocolEndpoint,  
1132 verify that the value of the IsDefault property is 2 (Is Not Default).  
 1133 This instance of CIM\_IPAssignmentSettingData represents a modifiable, static configuration for  
1134 the IP interface.
- 1135 3) Modify the properties of the CIM\_StaticIPAssignmentSettingData instance to contain the  
1136 appropriate configuration for the IP interface.
- 1137 4) Apply the pending configuration using the steps in section 9.9 or 9.10.

## 1138 9.9 Apply a Pending Configuration – Synchronously

1139 Some implementations may support modifying the configuration of an IP interface without requiring a  
1140 restart of the underlying network interface. If this behavior is supported by the implementation, then given  
1141 an instance of CIM\_IPProtocolEndpoint for which the configuration should be modified and an instance of  
1142 CIM\_IPAssignmentSettingData that represents the new configuration, a client can:

- 1143 1) Find an instance of CIM\_IPConfigurationService that is associated with the  
1144 CIM\_IPProtocolEndpoint instance through an instance of CIM\_ServiceAffectsElement.
- 1145 2) Invoke the ApplySettingToIPProtocolEndpoint() method of the CIM\_IPConfigurationService,  
1146 specifying the instances of CIM\_IPProtocolEndpoint and CIM\_IPAssignmentSettingData.

## 1147 9.10 Apply a Pending Configuration – Upon Restart

1148 Some implementations may require that the IP interface be restarted in order for a new configuration that  
1149 is bound to the interface to take effect. If an implementation requires that the IP interface be restarted,  
1150 then given an instance of CIM\_IPProtocolEndpoint for which the configuration should be modified and an  
1151 instance of CIM\_IPAssignmentSettingData that represents the new configuration, a client can:

- 1152 1) Find an instance of CIM\_ElementSettingData that associates the CIM\_IPAssignmentSettingData  
1153 instance with the CIM\_IPProtocolEndpoint instance.
- 1154 2) Set the IsNext property of the CIM\_ElementSettingData instance to a value of 1 (Is Next).
- 1155 3) Invoke the RequestStateChange() method of the CIM\_IPProtocolEndpoint instance, with a  
1156 RequestedState of 11 (Reset).

### 1157 **9.11 Determine Whether DNS Configuration Was DHCP Assigned**

1158 Starting at the CIM\_DNSProtocolEndpoint instance, a client can determine if any elements of the DNS  
1159 configuration were assigned through DHCP as follows:

- 1160 1) Find the instance of CIM\_IPProtocolEndpoint that is associated through an instance of  
1161 CIM\_SAPSAPDependency.
- 1162 2) Find the instance of CIM\_DHCPProtocolEndpoint that is associated with the  
1163 CIM\_IPProtocolEndpoint instance through an instance of CIM\_SAPSAPDependency.
- 1164 3) Query the EnabledState property of the CIM\_DHCPProtocolEndpoint instance for the value 2  
1165 (Enabled) to ensure that the DHCP client was used.
- 1166 4) Query the OptionsReceived property of the CIM\_DHCPProtocolEndpoint instance to determine  
1167 if one of the DNS-related options (8, 14, or 17) was received.

### 1168 **9.12 Determine Whether ElementName Can Be Modified**

1169 A client can determine whether it can modify the ElementName property of an instance of  
1170 CIM\_IPProtocolEndpoint as follows:

- 1171 1) Find the CIM\_EnabledLogicalElementCapabilities instance that is associated with the  
1172 CIM\_IPProtocolEndpoint instance.
- 1173 2) Query the value of the ElementNameEditSupported property of the  
1174 CIM\_EnabledLogicalElementCapabilities instance. If the value is TRUE, the client can modify  
1175 the ElementName property of the target instance.

### 1176 **9.13 Determine Whether State Management Is Supported**

1177 A client can determine whether state management is supported for an instance of  
1178 CIM\_IPProtocolEndpoint as follows:

- 1179 1) Find the CIM\_EnabledLogicalElementCapabilities instance that is associated with the  
1180 CIM\_IPProtocolEndpoint instance.
- 1181 2) Query the value of the RequestedStatesSupported property. If at least one value is specified,  
1182 state management is supported.

## 1183 **10 CIM Elements**

1184 Table 16 shows the instances of CIM Elements for this profile. Instances of the CIM Elements shall be  
1185 implemented as described in Table 16. Sections 7 (“Implementation”) and 8 (“Methods”) may impose  
1186 additional requirements on these elements.

1187 **Table 16 – CIM Elements: IP Interface Profile**

| Element Name                          | Requirement | Description                       |
|---------------------------------------|-------------|-----------------------------------|
| <b>Classes</b>                        |             |                                   |
| CIM_BindsToLANEndpoint                | Optional    | See sections 7.6 and 10.1.        |
| CIM_ElementCapabilities               | Conditional | See sections 7.1.2 and 10.2.      |
| CIM_EnabledLogicalElementCapabilities | Optional    | See sections 7.1.2 and 10.5.      |
| CIM_ElementSettingData                | Conditional | See sections 7.4, 10.3, and 10.4. |
| CIM_HostedAccessPoint                 | Mandatory   | See sections 10.6 and 10.7.       |
| CIM_HostedService                     | Conditional | See sections 7.4.1 and 10.8.      |

| Element Name                       | Requirement | Description                 |
|------------------------------------|-------------|-----------------------------|
| CIM_IPAssignmentSettingData        | Conditional | See sections 7.4 and 10.9.  |
| CIM_IPConfigurationService         | Optional    | See sections 7.4 and 10.10. |
| CIM_IPProtocolEndpoint             | Mandatory   | See section 10.11.          |
| CIM_OrderedComponent               | Conditional | See section 10.12.          |
| CIM_RegisteredProfile              | Mandatory   | See section 10.13.          |
| CIM_RemoteAccessAvailableToElement | Conditional | See section 10.14.          |
| CIM_RemoteServiceAccessPoint       | Optional    | See section 10.15.          |
| CIM_ServiceAffectsElement          | Conditional | See sections 7.4 and 10.16. |
| CIM_StaticIPAssignmentSettingData  | Conditional | See section 10.17.          |
| <b>Indications</b>                 |             |                             |
| None defined in this profile       |             |                             |

1188 **10.1 CIM\_BindsToLANEndpoint**

1189 CIM\_BindsToLANEndpoint relates the CIM\_IPProtocolEndpoint instance with the CIM\_LANEndpoint  
 1190 instance on which it depends. Table 17 provides information about the properties of  
 1191 CIM\_BindsToLANEndpoint.

1192 **Table 17 – Class: CIM\_BindsToLANEndpoint**

| Elements   | Requirement | Notes   |
|------------|-------------|---|
| Antecedent | Mandatory   | <b>Key</b> This shall be a reference to an instance of CIM_LANEndpoint.<br>Cardinality 0..1 |
| Dependent  | Mandatory   | <b>Key</b> This shall be a reference to the Central Instance.<br>Cardinality 1              |

1193 **10.2 CIM\_ElementCapabilities**

1194 CIM\_ElementCapabilities associates an instance of CIM\_EnabledLogicalElementCapabilities with the  
 1195 CIM\_IPProtocolEndpoint instance. Table 18 provides information about the properties of  
 1196 CIM\_ElementCapabilities.

1197 **Table 18 – Class: CIM\_ElementCapabilities**

| Elements       | Requirement | Notes  |
|----------------|-------------|--|
| ManagedElement | Mandatory   | <b>Key</b> This shall be a reference to the Central Instance.<br>Cardinality 1..*                                  |
| Capabilities   | Mandatory   | <b>Key</b> This shall be a reference to the instance of CIM_EnabledLogicalElementCapabilities.<br>Cardinality 0..1 |

### 1198 10.3 CIM\_ElementSettingData—CIM\_IPAssignmentSettingData Reference

1199 CIM\_ElementSettingData associates instances of CIM\_IPAssignmentSettingData with the  
1200 CIM\_IPProtocolEndpoint instance. Table 19 provides information about the properties of  
1201 CIM\_ElementSettingData.

1202 **Table 19 – Class: CIM\_ElementSettingData – CIM\_IPAssignmentSettingData**

| Elements       | Requirement | Notes  |
|----------------|-------------|--|
| ManagedElement | Mandatory   | <b>Key</b> This shall be a reference to the Central Instance.<br>Cardinality 1..*                    |
| SettingData    | Mandatory   | <b>Key</b> This shall be a reference to an instance of CIM_IPAssignmentSettingData.<br>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)                                  |

### 1203 10.4 CIM\_ElementSettingData—CIM\_StaticIPAssignmentSettingData Reference

1204 CIM\_ElementSettingData associates instances of CIM\_StaticIPAssignmentSettingData with the  
1205 CIM\_IPProtocolEndpoint instance. Table 20 provides information about the properties of  
1206 CIM\_ElementSettingData.

1207 **Table 20 – Class: CIM\_ElementSettingData – CIM\_StaticIPAssignmentSettingData**

| Elements       | Requirement | Notes  |
|----------------|-------------|--|
| ManagedElement | Mandatory   | <b>Key</b> This shall be a reference to the Central Instance.<br>Cardinality 1..*                          |
| SettingData    | Mandatory   | <b>Key</b> This shall be a reference to an instance of CIM_StaticIPAssignmentSettingData.<br>Cardinality * |
| IsCurrent      | Mandatory   | Matches 1 (Is Current) or 2 (Is Not Current)   |

### 1208 10.5 CIM\_EnabledLogicalElementCapabilities

1209 CIM\_EnabledLogicalElementCapabilities indicates support for managing the IP interface. Table 21  
1210 provides information about the properties of CIM\_EnabledLogicalElementCapabilities.

1211 **Table 21 – Class: CIM\_EnabledLogicalElementCapabilities**

| Elements                 | Requirement | Notes                                 |
|--------------------------|-------------|---------------------------------------|
| InstanceID               | Mandatory   | <b>Key</b>                            |
| RequestedStatesSupported | Mandatory   | See sections 7.1.2.1.1 and 7.1.3.1.1. |
| ElementNameEditSupported | Mandatory   | See sections 7.1.4.1.1 and 7.1.5.1.1. |
| MaxElementNameLen        | Conditional | See sections 7.1.4.1.2 and 7.1.5.1.2. |



1212 **10.6 CIM\_HostedAccessPoint—CIM\_RemoteServiceAccessPoint Reference**

1213 An instance of CIM\_HostedAccessPoint Association between an instance of CIM\_ProtocolEndpoint and  
 1214 CIM\_RemoteServiceAccessPoint shall only be instantiated if CIM\_RemoteServiceAccessPoint is  
 1215 supported.

1216 CIM\_HostedAccessPoint relates the CIM\_RemoteServiceAccessPoint instance that represents the  
 1217 default gateway with its scoping CIM\_ComputerSystem instance. Table 22 provides information about the  
 1218 properties of CIM\_HostedAccessPoint.

1219 **Table 22 – Class: CIM\_HostedAccessPoint – CIM\_RemoteServiceAccessPoint**

| Elements   | Requirement | Notes   |
|------------|-------------|---|
| Antecedent | Mandatory   | <b>Key</b> This shall be a reference to the Scoping Instance.<br>Cardinality 1                        |
| Dependent  | Mandatory   | <b>Key</b> This shall be a reference to an instance of CIM_RemoteServiceAccessPoint.<br>Cardinality * |

1220 **10.7 CIM\_HostedAccessPoint—CIM\_IPProtocolEndpoint Reference**

1221 CIM\_HostedAccessPoint relates the Central Instance with its Scoping Instance. Table 23 provides  
 1222 information about the properties of CIM\_HostedAccessPoint.

1223 **Table 23 – Class: CIM\_HostedAccessPoint – CIM\_IPProtocolEndpoint**

| Elements   | Requirement | Notes  |
|------------|-------------|--|
| Antecedent | Mandatory   | <b>Key</b> This shall be a reference to the Central Instance.<br>Cardinality 1                           |
| Dependent  | Mandatory   | <b>Key</b> This shall be a reference to an instance of CIM_RemoteServiceAccessPoint.<br>Cardinality 1..* |

1224 **10.8 CIM\_HostedService**

1225 CIM\_HostedService relates the CIM\_IPConfigurationService instance to its scoping  
 1226 CIM\_ComputerSystem instance. Table 24 provides information about the properties of  
 1227 CIM\_HostedService.

1228 **Table 24 – Class: CIM\_HostedService**

| Elements   | Requirement | Notes   |
|------------|-------------|---|
| Antecedent | Mandatory   | <b>Key</b> This shall be a reference to the Central Instance.<br>Cardinality 1                      |
| Dependent  | Mandatory   | <b>Key</b> This shall be a reference to an instance of CIM_IPConfigurationService.<br>Cardinality * |

1229 **10.9 CIM\_IPAssignmentSettingData**

1230 CIM\_IPAssignmentSettingData is the aggregation point for the SettingData instances that define a  
 1231 configuration that can be applied to an IP interface. Table 25 provides information about the properties of  
 1232 CIM\_IPAssignmentSettingData.

1233 **Table 25 – Class: CIM\_IPAssignmentSettingData**

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

1234 **10.10 CIM\_IPConfigurationService**

1235 CIM\_IPConfigurationService represents the ability to configure an IP interface. Table 26 provides  
 1236 information about the properties of CIM\_IPConfigurationService.

1237 **Table 26 – Class: CIM\_IPConfigurationService**

| Elements                           | Requirement | Notes                |
|------------------------------------|-------------|----------------------|
| SystemCreationClassName            | Mandatory   | Key                  |
| CreationClassName                  | Mandatory   | Key                  |
| SystemName                         | Mandatory   | Key                  |
| Name                               | Mandatory   | Key                  |
| ElementName                        | Mandatory   | Pattern ".*"         |
| ApplySettingToIPProtocolEndpoint() | Optional    | See section 8.1.1.1. |

1238 **10.11 CIM\_IPProtocolEndpoint**

1239 CIM\_IPProtocolEndpoint represents an IP interface that is associated with an Ethernet interface. Table 27  
 1240 provides information about the properties of CIM\_IPProtocolEndpoint.

1241 **Table 27 – Class: CIM\_IPProtocolEndpoint**

| Elements                | Requirement | Notes                             |
|-------------------------|-------------|-----------------------------------|
| SystemCreationClassName | Mandatory   | Key                               |
| CreationClassName       | Mandatory   | Key                               |
| SystemName              | Mandatory   | Key                               |
| Name                    | Mandatory   | Key                               |
| NameFormat              | Mandatory   | Pattern ".*"                      |
| ProtocolIFType          | Mandatory   | See section 7.1.1.2.              |
| RequestedState          | Mandatory   | See sections 7.1.2.2 and 7.1.3.2. |
| EnabledState            | Mandatory   | See sections 7.1.2.3 and 7.1.3.3. |
| ElementName             | Mandatory   | Pattern ".*"                      |
| RequestStateChange()    | Conditional | See section 8.1.                  |

| Elements               | Requirement | Notes   |
|------------------------|-------------|---|
| IPv4Address            | Conditional | See section 7.1.1.2.                            |
| SubnetMask             | Conditional | See sections 7.1.1.2 and 7.1.1.4.               |
| AddressOrigin          | Mandatory   | See section 7.1.1.1.                            |
| IPv6Address            | Conditional | See sections 7.1.1.2 and 7.1.1.5 – EXPERIMENTAL |
| IPv6AddressType        | Conditional | See section 7.1.1.2 – EXPERIMENTAL              |
| IPv6SubnetPrefixLength | Conditional | See section 7.1.1.2 – EXPERIMENTAL              |

1242 **10.12 CIM\_OrderedComponent**

1243 CIM\_OrderedComponent associates an instance of CIM\_IPAssignmentSettingData to the instances of  
 1244 CIM\_StaticIPAssignmentSettingData, CIM\_DHCPSettingData, CIM\_DNSSettingData, and  
 1245 CIM\_DNSGeneralSettingData that compose a configuration. Table 28 provides information about the  
 1246 properties of CIM\_OrderedComponent.

1247 **Table 28 – Class: CIM\_OrderedComponent**

| Elements         | Requirement | Notes                           |
|------------------|-------------|---------------------------------|
| GroupComponent   | Mandatory   | <b>Key</b> See section 7.4.3.1. |
| PartComponent    | Mandatory   | <b>Key</b> See section 7.4.3.2. |
| AssignedSequence | Mandatory   | See section 7.4.3.3.            |

1248 **10.13 CIM\_RegisteredProfile**

1249 CIM\_RegisteredProfile identifies the *IP Interface Profile* in order for a client to determine whether an  
 1250 instance of CIM\_IPProtocolEndpoint is conformant with this profile. The CIM\_RegisteredProfile class is  
 1251 defined by the *Profile Registration Profile*. With the exception of the mandatory values specified for the  
 1252 properties in Table 29, the behavior of the CIM\_RegisteredProfile instance is in accordance with the  
 1253 *Profile Registration Profile*.

1254 **Table 29 – Class: CIM\_RegisteredProfile**

| Elements               | Requirement | Notes   |
|------------------------|-------------|---|
| RegisteredName         | Mandatory   | This property shall have a value of "IP Interface". |
| RegisteredVersion      | Mandatory   | This property shall have a value of "1.0.0".        |
| RegisteredOrganization | Mandatory   | This property shall have a value of "DMTF".         |

1255 NOTE: Previous versions of this document included the suffix "Profile" for the RegisteredName value. If  
 1256 implementations querying for the RegisteredName value find the suffix "Profile", they should ignore the suffix, with  
 1257 any surrounding white spaces, before any comparison is done with the value as specified in this document.

## 1258 10.14 CIM\_RemoteAccessAvailableToElement

1259 CIM\_RemoteAccessAvailableToElement associates the CIM\_IPProtocolEndpoint instance with the  
1260 CIM\_RemoteServiceAccessPoint instance that represents the network gateway. Table 30 provides  
1261 information about the properties of CIM\_RemoteAccessAvailableToElement.

1262 **Table 30 – Class: CIM\_RemoteAccessAvailableToElement**

| Elements      | Requirement | Notes                           |
|---------------|-------------|---------------------------------|
| Antecedent    | Mandatory   | <b>Key</b> See section 7.1.6.2. |
| Dependent     | Mandatory   | <b>Key</b> See section 7.1.6.3. |
| OrderOfAccess | Mandatory   | See section 7.1.6.4.            |

## 1263 10.15 CIM\_RemoteServiceAccessPoint

1264 CIM\_RemoteServiceAccessPoint represents the managed system's view of the default gateway. Table  
1265 31 provides information about the properties of CIM\_RemoteServiceAccessPoint.

1266 **Table 31 – Class: CIM\_RemoteServiceAccessPoint**

| Elements                | Requirement | Notes                       |
|-------------------------|-------------|-----------------------------|
| SystemCreationClassName | Mandatory   | <b>Key</b>                  |
| CreationClassName       | Mandatory   | <b>Key</b>                  |
| SystemName              | Mandatory   | <b>Key</b>                  |
| Name                    | Mandatory   | <b>Key</b>                  |
| AccessContext           | Mandatory   | Matches 2 (Default Gateway) |
| AccessInfo              | Mandatory   | See section 7.1.6.1.        |
| InfoFormat              | Mandatory   | Matches 3 (IPv4 Address)    |
| ElementName             | Mandatory   | Pattern ".*"                |

## 1267 10.16 CIM\_ServiceAffectsElement

1268 CIM\_ServiceAffectsElement associates an instance of CIM\_IPConfigurationService with an instance of  
1269 CIM\_IPProtocolEndpoint that the service is able to configure. Table 32 provides information about the  
1270 properties of CIM\_ServiceAffectsElement.

1271 **Table 32 – Class: CIM\_ServiceAffectsElement**

| Elements         | Requirement | Notes   |
|------------------|-------------|---|
| AffectingElement | Mandatory   | <b>Key</b> This shall be a reference to the instance of<br>CIM_IPConfigurationService.<br>Cardinality * |
| AffectedElement  | Mandatory   | <b>Key</b> This shall be a reference to the Central Instance.<br>Cardinality 1..*                       |
| ElementAffects   | Mandatory   | Matches 5 (Manages)   |

1272 **10.17 CIM\_StaticIPAssignmentSettingData**

1273 CIM\_StaticIPAssignmentSettingData represents a static configuration that can be applied to an instance  
 1274 of CIM\_IPProtocolEndpoint. Table 33 provides information about the properties of  
 1275 CIM\_StaticIPAssignmentSettingData.

1276 **Table 33 – Class: CIM\_StaticIPAssignmentSettingData**

| Elements               | Requirement | Notes                |
|------------------------|-------------|----------------------|
| InstanceID             | Mandatory   | <b>Key</b>           |
| AddressOrigin          | Mandatory   | Matches 3 (Static)   |
| ElementName            | Mandatory   | Pattern ".*"         |
| IPv4Address            | Mandatory   |                      |
| SubnetMask             | Mandatory   |                      |
| GatewayIPv4Address     | Conditional | See section 7.5.3.1. |
| IPv6Address            | Optional    | EXPERIMENTAL         |
| IPv6AddressType        | Optional    | EXPERIMENTAL         |
| IPv6SubnetPrefixLength | Optional    | EXPERIMENTAL         |
| GatewayIPv6Address     | Optional    | EXPERIMENTAL         |

1277

**ANNEX A  
(informative)****Change Log**1278  
1279  
1280  
1281

| Version | Date       | Author       | Description   |
|---------|------------|--------------|---|
| 1.0.0a  | 2006/07/11 | Aaron Merkin | Preliminary Standard                                      |
| 1.0.0   | 2008/07/27 | Jeff Hilland | Final Standard & addition of IPv6 support as Experimental |
|         |            |              |   |
|         |            |              |   |

1282

## ANNEX B (informative)

1283  
1284  
1285  
1286

### Acknowledgments

1287 The authors wish to acknowledge the following people.

1288 **Editors:**

- 1289 • Aaron Merkin – IBM
- 1290 • Jeff Hilland – HP

1291 **Contributors:**

- 1292 • RadhaKrishna Dasari – Dell
- 1293 • Jon Hass – Dell
- 1294 • Khachatur Papanyan – Dell
- 1295 • Enoch Suen – Dell
- 1296 • Jeff Hilland – HP
- 1297 • Christina Shaw – HP
- 1298 • Aaron Merkin – IBM
- 1299 • Perry Vincent – Intel
- 1300 • John Leung – Intel
- 1301