



Document Number: DSP1092

Date: 2013-08-22

Version: 1.0.0g

WBEM Server Profile

Information for Work-in-Progress version:

IMPORTANT: This document is not a standard. It does not necessarily reflect the views of the DMTF or all of its members. Because this document is a Work in Progress, it may still change, perhaps profoundly. This document is available for public review and comment until the stated expiration date.

It expires on: <2014-01-31>

Provide any comments through the DMTF Feedback Portal:
<http://www.dmtf.org/standards/feedback>

Document Type: Specification

Document Status: Work in Progress

Document Language: en-US

10 Copyright notice

11 Copyright © 2013 Distributed Management Task Force, Inc. (DMTF). All rights reserved.

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

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26 withdrawn or modified after publication, and shall be indemnified and held harmless by any party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

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

32

CONTENTS

34	Foreword	6
35	Introduction.....	7
36	1 Scope	8
37	2 Normative references	8
38	3 Terms and definitions	9
39	4 Symbols and abbreviated terms.....	9
40	5 Synopsis	10
41	6 Description (Informative)	10
42	7 Implementation.....	12
43	7.1 WBEM server.....	12
44	7.2 WBEM server namespaces	12
45	7.3 WBEM protocols	13
46	7.4 WBEM protocol management.....	13
47	7.5 Indications Profile.....	15
48	8 Methods.....	15
49	8.1 Profile conventions for operations	15
50	8.2 CIM_ComputerSystem.....	15
51	8.3 CIM_WBEMServer.....	15
52	8.4 CIM_WBEMServerNamespace	17
53	8.5 CIM_ProtocolService	18
54	8.6 CIM_HostedDependency.....	20
55	8.7 CIM_ServiceServiceDependency	20
56	8.8 CIM_HostedService (WBEMServer).....	21
57	8.9 CIM_HostedService (ProtocolService)	21
58	8.10 CIM_WBEMServerCapabilities	21
59	8.11 CIM_ElementCapabilities (WBEMServerCapabilities)	21
60	8.12 CIM_WBEMProtocolServiceCapabilities	22
61	8.13 CIM_ElementCapabilities (WBEMProtocolServiceCapabilities).....	22
62	8.14 CIM_TCPProtocolEndpoint.....	22
63	8.15 CIM_ServiceAccessBySAP	22
64	8.16 CIM_HostedAccessPoint (TCPProtocolEndpoint).....	23
65	8.17 CIM_BindsTo	23
66	8.18 CIM_CIMXMLCapabilities.....	23
67	8.19 CIM_WSMangementCapabilities.....	23
68	9 Use cases (Informative)	24
69	9.1 Determine the namespaces of a WBEM server	25
70	9.2 Determine the contents of a namespace	25
71	9.3 Modify WBEMServerNamespace to update what is represented	26
72	9.4 Determine the WBEM protocols supported and state	26
73	9.5 Determine the port used for a WBEM protocol	26
74	9.6 Determine the IP address for a WBEM protocol.....	26
75	9.7 Determine the capabilities of a WBEM protocol	27
76	9.8 Modify the port for a WBEM protocol.....	27
77	9.9 Disable/Enable a WBEM protocol.....	27
78	9.10 Reset the WBEM server	28
79	9.11 Shut down the WBEM server.....	28
80	10 CIM elements	29
81	10.1 CIM_BindsTo	30
82	10.2 CIM_CIMXMLCapabilities.....	30
83	10.3 CIM_ComputerSystem.....	30

84 10.4 CIM_ConcreteJob 31

85 10.5 CIM_ElementCapabilities (WBEMServerCapabilities) 31

86 10.6 CIM_ElementCapabilities (WBEMProtocolServiceCapabilities) 31

87 10.7 CIM_ElementConformsToProfile 31

88 10.8 CIM_GenericOperationCapabilitiesStructure 32

89 10.9 CIM_HostedAccessPoint 32

90 10.10 CIM_HostedAccessPoint (TCPProtocolEndpoint) 32

91 10.11 CIM_HostedDependency 32

92 10.12 CIM_HostedService (WBEMServer) 33

93 10.13 CIM_HostedService (ProtocolService) 33

94 10.14 CIM_IPProtocolEndpoint 33

95 10.15 CIM_ProtocolService 33

96 10.16 CIM_RegisteredProfile 34

97 10.17 CIM_SchemaInformationStructure 34

98 10.18 CIM_ServiceAccessBySAP 34

99 10.19 CIM_ServiceServiceDependency 35

100 10.20 CIM_TCPProtocolEndpoint 35

101 10.21 CIM_WBEMProtocolServiceCapabilities 35

102 10.22 CIM_WBEMServer 36

103 10.23 CIM_WBEMServerCapabilities 36

104 10.24 CIM_WBEMServerNamespace 37

105 10.25 CIM_WSManagementCapabilities 37

106 ANNEX A (informative) Change log 38

107 **Figures**

109 Figure 1 – WBEM Server Profile: Class diagram 11

110 Figure 2 – WBEM Server Profile: Instance diagram 24

111 Figure 3 – WBEM Server Profile: Instance diagram with multiple WBEM protocols 25

112

113 **Tables**

114 Table 1 – Referenced profiles 10

115 Table 2 – RequestStateChange() method: Return code values 16

116 Table 3 – RequestStateChange() method: Parameters 16

117 Table 4 – CreateWBEMServerNamespace() method: Return code values 17

118 Table 5 – CreateWBEMServerNamespace() method: Parameters 17

119 Table 6 – Operations: CIM_HostedDependency 17

120 Table 7 – RequestStateChange() method: Return code values 18

121 Table 8 – RequestStateChange() method: Parameters 19

122 Table 9 – ListenOnPortIF() method: Return code values 19

123 Table 10 – ListenOnPortIF() method: Parameters 20

124 Table 11 – Operations: CIM_HostedDependency 20

125 Table 12 – Operations: CIM_ServiceServiceDependency 20

126 Table 13 – Operations: CIM_HostedService (WBEMServer) 21

127 Table 14 – Operations: CIM_HostedService (ProtocolService) 21

128 Table 15 – Operations: CIM_ElementCapabilities (WBEMServerCapabilities) 21

129 Table 16 – Operations: CIM_ElementCapabilities (WBEMProtocolServiceCapabilities) 22

130 Table 17 – Operations: CIM_ServiceAccessBySAP 22

131 Table 18 – Operations: CIM_HostedAccessPoint (TCPProtocolEndpoint) 23

132 Table 19 – Operations: CIM_BindsTo..... 23

133 Table 20 – CIM Elements: WBEM Server Profile 29

134 Table 21 – Class: CIM_BindsTo 30

135 Table 22 – Class: CIM_CIMXMLCapabilities..... 30

136 Table 23 – Class: CIM_ComputerSystem..... 30

137 Table 24 – Class: CIM_ElementCapabilities (WBEMServerCapabilities) 31

138 Table 25 – Class: CIM_ElementCapabilities (WBEMProtocolServiceCapabilities) 31

139 Table 26 – Class: CIM_ElementConformsToProfile 31

140 Table 27 – Class: CIM_GenericOperationCapabilitiesStructure..... 32

141 Table 28 – Class: CIM_HostedAccessPoint (TCPProtocolEndpoint) 32

142 Table 29 – Class: CIM_HostedDependency..... 32

143 Table 30 – Class: CIM_HostedService (WBEMServer)..... 33

144 Table 31 – Class: CIM_HostedService (ProtocolService) 33

145 Table 32 – Class: CIM_ProtocolService 33

146 Table 33 – Class: CIM_RegisteredProfile 34

147 Table 34 – Class: CIM_SchemaInformationStructure 34

148 Table 35 – Class: CIM_ServiceAccessBySAP 34

149 Table 36 – Class: CIM_ServiceServiceDependency 35

150 Table 37 – Class: CIM_TCPProtocolEndpoint..... 35

151 Table 38 – Class: CIM_WBEMProtocolServiceCapabilities 35

152 Table 39 – Class: CIM_WBEMServer..... 36

153 Table 40 – Class: CIM_WBEMServerCapabilities 36

154 Table 41 – Class: CIM_WBEMServerNamespace 37

155 Table 42 – Class: CIM_WSManagementCapabilities 37

156

157

Foreword

158 The *WBEM Server Profile* (DSP1092) was prepared by the DMTF WBEM Infrastructure Modeling
159 Working Group.

160 DMTF is a not-for-profit association of industry members dedicated to promoting enterprise and systems
161 management and interoperability. For information about the DMTF, see <http://www.dmtf.org>.

162 Acknowledgments

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

- 164 • Jim Davis – WS (Editor)
- 165 • George Ericson – EMC
- 166 • Paul Ferdinand – WS
- 167 • Peter Lamanna – EMC
- 168 • Larry Lamers – VMWare
- 169 • Paul Lapomardo – EMC
- 170 • Andreas Maier – IBM
- 171 • Jim Marshall – WS
- 172 • Karl Schopmeyer – Inova Development
- 173 • Mike Walker – Individual Contributor

174 The DMTF also acknowledges the contributions of the Storage Network Industry Association (SNIA).

175

176

Introduction

177 The information in this specification should be sufficient for a provider or consumer of this data to
178 unambiguously identify the classes, properties, methods, and values that shall be instantiated to manage
179 and monitor WBEM server and WBEM protocols using the DMTF Common Information Model (CIM)
180 Schema.

181 The target audience for this specification is implementers who are writing CIM-based providers or
182 consumers of management interfaces that represent the components described in this document.

183 Document conventions

184 Typographical conventions

185 The following typographical conventions are used in this document:

- 186 • Document titles are marked in *italics*.
- 187 • Important terms that are used for the first time are marked in *italics*.
- 188 • ABNF rules are in `monospaced font`.

189 ABNF usage conventions

190 Format definitions in this document are specified using ABNF (see [RFC5234](#)), with the following
191 deviations:

- 192 • Literal strings are to be interpreted as case-sensitive Unicode characters, as opposed to the
193 definition in [RFC5234](#) that interprets literal strings as case-insensitive US-ASCII characters.

194 Experimental material

195 Experimental material has yet to receive sufficient review to satisfy the adoption requirements set forth by
196 the DMTF. Experimental material is included in this document as an aid to implementers who are
197 interested in likely future developments. Experimental material may change as implementation
198 experience is gained. It is likely that experimental material will be included in an upcoming revision of the
199 specification. Until that time, experimental material is purely informational.

200 The following typographical convention indicates experimental material:

201 **EXPERIMENTAL**

202 Experimental material appears here.

203 **EXPERIMENTAL**

204 In places where this typographical convention cannot be used (for example, tables or figures), the
205 "EXPERIMENTAL" label is used alone.

206

WBEM Server Profile

207 1 Scope

208 The *WBEM Server Profile* defines the CIM elements that are used to report and manage information
209 regarding the WBEM server.

210 2 Normative references

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

215 DMTF DSP0004, *CIM Infrastructure Specification 2.7*,
216 http://dmtof.org/standards/published_documents/DSP0004_2.7.pdf

217 DMTF DSP0200, *CIM Operations over HTTP 1.3*,
218 http://dmtof.org/standards/published_documents/DSP0200_1.3.pdf

219 DMTF DSP1052, *Computer System Profile 1.0*,
220 http://dmtof.org/standards/published_documents/DSP1052_1.0.pdf

221 DMTF DSP0207, *WBEM URI Mapping 1.0*,
222 http://dmtof.org/standards/published_documents/DSP0207_1.0.pdf

223 DMTF DSP0223, *Generic Operations Specification 1.0*,
224 http://dmtof.org/standards/published_documents/DSP0223_1.0.pdf

225 DMTF DSP1036, *IP Interface Profile 1.1*,
226 http://dmtof.org/standards/published_documents/DSP1036_1.1.pdf

227 DMTF DSP1054, *Indications Profile 1.2*,
228 http://dmtof.org/standards/published_documents/DSP1054_1.2.pdf

229 DMTF DSP1103, *Job Control Profile 1.0*,
230 http://dmtof.org/standards/published_documents/DSP1103_1.0.pdf

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

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

235 DMTF DSP1034, *Simple Identity Management Profile 1.1*,
236 http://dmtof.org/standards/published_documents/DSP1034_1.1.pdf

237 DMTF DSP1039, *Role Based Authorization Profile 1.1*,
238 http://dmtof.org/standards/published_documents/DSP1039_1.1.pdf

239 IETF RFC3986, *Uniform Resource Identifier (URI): Generic Syntax, Jan. 2005*,
240 <http://www.ietf.org/rfc/rfc3986.txt>

241 IETF RFC5234, *Augmented BNF for Syntax Specifications: ABNF, Jan. 2008*,
242 <http://www.ietf.org/rfc/rfc5234.txt>

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

245 **3 Terms and definitions**

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

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

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

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

259 The terms defined in [DSP0004](#), [DSP0200](#), and [DSP1001](#) apply to this document. The following additional
260 terms are used in this document.

261 **3.1**

262 **WBEM server**

263 a CIM server (see [DSP0004](#)) that supports at least one WBEM protocol.

264 **3.2**

265 **WBEM protocol**

266 A communications protocol that defines WBEM Operations (see [DSP0223](#))

267

268 **4 Symbols and abbreviated terms**

269 The abbreviations defined in [DSP0004](#), [DSP0200](#), and [DSP1001](#) apply to this document. The following
270 additional abbreviations are used in this document.

271 **4.1**

272 **CQL**

273 CIM Query Language

274 **4.2**

275 **FQL**

276 Filter Query Language

277 **4.3**

278 **URI**

279 Uniform Resource Identifier

280 **4.4**
 281 **WBEM**
 282 Web Based Enterprise Management

283 **5 Synopsis**

284 **Profile name:** WBEM Server

285 **Version:** 1.0.0

286 **Organization:** DMTF

287 **CIM schema version:** 2.38

288 **Central Class:** CIM_ComputerSystem

289 **Scoping Class:** CIM_ComputerSystem

290 The *WBEM Server Profile* is an autonomous profile that specializes the *Computer System Profile*
 291 ([DSP1052](#)) to provide the capability to discover, monitor and manage the WBEM server infrastructure.

292 The central instance of this profile shall be an instance of CIM_ComputerSystem. The scoping instance
 293 shall be the instance of CIM_ComputerSystem.

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

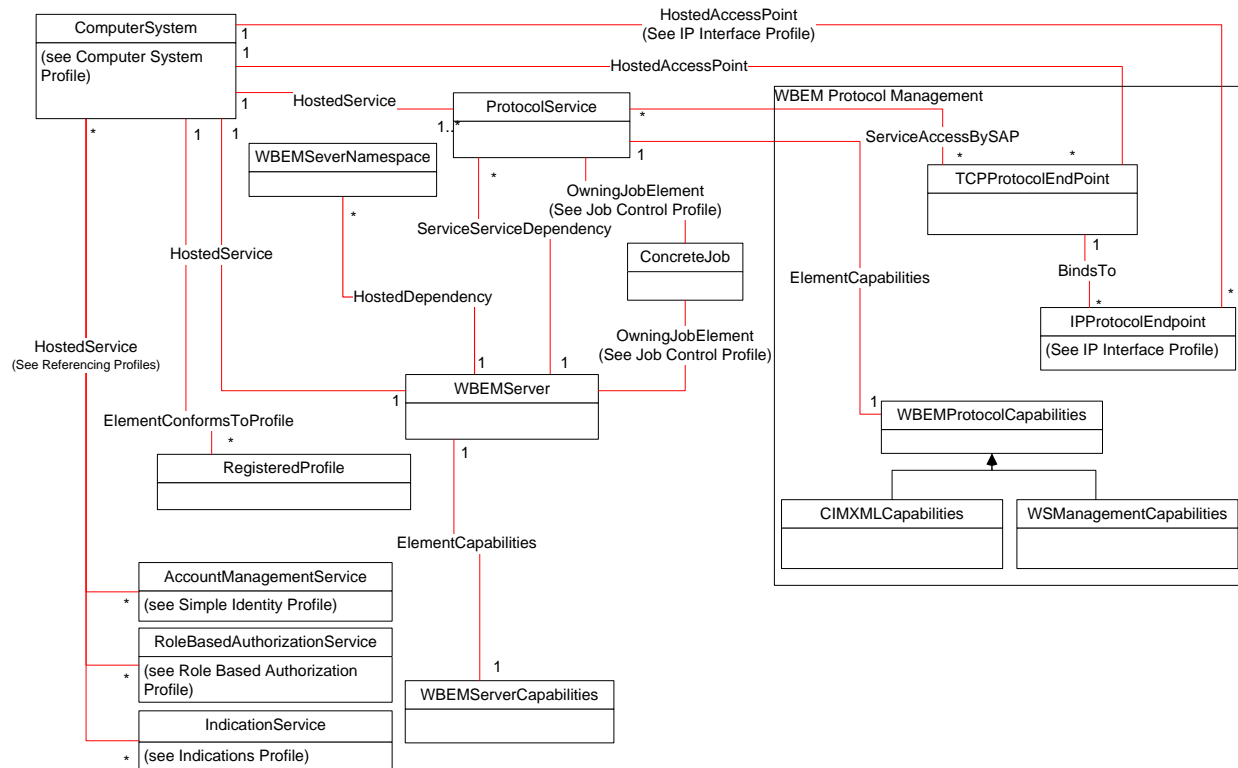
295 **Table 1 – Referenced profiles**

Profile Name	Organization	Version	Relationship	Behavior
Computer System (DSP1052)	DMTF	1.0	Specializes	
Indications Profile (DSP1054)	DMTF	1.2	Optional	
IP Interface Profile (DSP1036)	DMTF	1.1	Conditional	Mandatory when WBEM protocol management is supported.
Job Control Profile (DSP1103)	DMTF	1.0	Optional	
Profile Registration Profile (DSP1033)	DMTF	1.0	Mandatory	
Role Based Authorization (DSP1039)	DMTF	1.0	Mandatory	
Simple Identity Management (DSP1034)	DMTF	1.1	Mandatory	

296 **6 Description (Informative)**

297 The *WBEM Server Profile* describes the WBEM server and WBEM protocol management.

298 Figure 1 represents the UML class diagram for the *WBEM Server Profile*. For simplicity, the *CIM_* prefix
 299 has been removed from the names of the classes in Figure 1.



300

301

Figure 1 – WBEM Server Profile: Class diagram

302 The *WBEM Server Profile* represents the capabilities of the WBEM server and supported WBEM
 303 protocols. Functionality within the scope of this profile includes: namespace discovery, WBEM server
 304 settings and WBEM protocol discovery and management.

305 The WBEM Server profile mandates the support of the Simple Identity Profile and Role Based
 306 Authorization profiles to provide an interoperable means for authentication and authorization.

307 The WBEM server is modeled as an instance of CIM_WBEMServer, a subclass of CIM_Service. Aspects
 308 of the WBEM server’s configuration are modeled through the capabilities and settings associated to the
 309 instance of CIM_WBEMServer.

310 Namespaces are modeled using the class CIM_WBEMServerNamespace. Namespace instances include
 311 the information regarding the schemas contained in the namespace.

312 A WBEM protocol is modeled using a single instance of CIM_ProtocolService. A WBEM server may have
 313 support for one or more WBEM protocols.

314 A WBEM server may support the managing of WBEM protocols. If supported, the administrator can
 315 enable, disable or reset a protocol, determine the capabilities supported, change the port(s) the protocol
 316 is listening on and bind the protocol to a specific IP Address.

317 WBEM protocol session management is outside the scope of this profile.

318 7 Implementation

319 This clause details the requirements related to the arrangement of instances and their properties for
320 implementations of this profile. Methods are listed in clause 8 ("Methods"), and properties are listed in
321 clause 10 ("CIM Elements").

322 The WBEM Server profile shall be implemented in the interop namespace.

323 7.1 WBEM server

324 Exactly one instance of this class shall exist that represents the WBEM server.

325 7.1.1 WBEM server capabilities

326 The WBEM server capabilities are represented through an instance of CIM_WBEMServerCapabilities,
327 Exactly one instance of this class shall exist and be associated to the CIM_WBEMServer instance
328 through an instance of CIM_WBEMServerCapabilities.OperationsSupported

329 7.1.1.1 CIM_WBEMServerCapabilities.OperationsSupported

330 Support for methods for CIM_WBEMServer is optional. For each method supported, the
331 OperationsSupported property shall contain the value representing the method.

332 7.1.1.2 CIM_WBEMServerCapabilities.RequestedStatesSupported

333 If the CIM_WBEMServerCapabilities.OperationsSupported property value includes 2
334 (RequestStateChange) this property value shall include at least one of the following values: 4 (Shut
335 Down), 9 (Reboot) or 10 (Reset).

336 If the CIM_WBEMServerCapabilities.OperationsSupported property does not contain the value 2
337 (RequestedStateChange), then the RequestedStatesSupported property shall be Null.

338 7.1.2 CIM_WBEMServer.RequestedState

339 When the CIM_WBEMServer.RequestStateChange() method is successfully invoked, the value of the
340 RequestedState property shall be the value of the RequestedState parameter.

341 The CIM_WBEMServer.RequestedState property shall have one of the values specified in the
342 CIM_WBEMServerCapabilities.RequestedStatesSupported property or a value of 5 (No Change).

343 7.1.3 CIM_WBEMServer.EnabledState

344 The EnabledState property shall have the value 2 (Enabled) or 4 (Shutting Down).

345 When the RequestedState parameter has a value of 4 (Shut Down) and the
346 CIM_WBEMServer.RequestStateChange() method completes successfully, the EnabledState property
347 shall change to 4 (Shutting Down) until the WBEM server is no longer available.

348 If the method does not complete successfully, the value of the EnabledState shall be 2 (Enabled) and the
349 WBEM server shall not try to shut down.

350 7.2 WBEM server namespaces

351 A WBEM server namespace is modeled using the class CIM_WBEMServerNamespace. A WBEM server
352 may contain one or more namespaces. The following clauses describe implementation requirements for
353 CIM_WBEMServerNamespace.

354 7.2.1 CIM_WBEMServerNamespace.Name

355 The Name property is the actual name of the namespace. The namespace name shall be unique, in
356 other words you shall not have two namespace with the same name in a WBEM server.

357 7.3 WBEM protocols

358 A WBEM protocol is modeled using the CIM_ProtocolService class. For each WBEM protocol supported
359 an instance of CIM_ProtocolService shall exist. At least one instance of CIM_ProtocolService shall exist.

360 The following subclauses describe implementation requirements for CIM_ProtocolService.

361 7.3.1 CIM_ProtocolService.RequestedStateChange

362 When the CIM_ProtocolService.RequestStateChange() method is successfully invoked, the value of the
363 RequestedState property shall be the value of the RequestedState parameter.

364 The CIM_ProtocolService.RequestedState property shall have one of the values specified in the
365 CIM_WBEMProtocolServiceCapabilities.RequestedStatesSupported property or a value of 5 (No
366 Change).

367 7.3.2 CIM_ProtocolService.EnabledState

368 When the RequestedState parameter has a value of 2 (Enabled) or 3 (Disabled) and the
369 CIM_ProtocolService.RequestStateChange() method completes successfully, the value of the
370 EnabledState property shall equal the value of the CIM_ProtocolService.RequestedState property.

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

372 7.4 WBEM protocol management

373 WBEM protocol management for a WBEM protocol is conditional. A WBEM server may support one or
374 more WBEM protocols. If WBEM protocol management is supported for a specified protocol, an instance
375 of CIM_WBEMProtocolServiceCapabilities shall be associated to the instance of CIM_ProtocolService via
376 CIM_ElementCapabilities and the
377 CIM_WBEMProtocolServiceCapabilities.ListeningPortManagementSupported value shall be True.

378 7.4.1 WBEM protocol capabilities

379 WBEM protocol capabilities are modeled using CIM_WBEMProtocolServiceCapabilities. The support for
380 the CIM_WBEMProtocolServiceCapabilities is optional.

381 7.4.1.1 CIM_WBEMProtocolServiceCapabilities.ListeningPortManagementSupported

382 If the CIM_WBEMProtocolServiceCapabilities is implemented, then the
383 ListeningPortManagementSupported property shall be supported. When the
384 ListeningPortManagementSupported property has a value of True, the instance(s) of
385 CIM_ProtocolService associated via CIM_ElementCapabilities shall support the ListenOnPortIF() method.
386 When the ListeningPortManagementSupported property has a value of False, the ListenOnPortIF()
387 method shall not be supported.

388 7.4.1.2 CIM_WBEMProtocolServiceCapabilities.RequestedStatesSupported

389 If WBEM protocol management is supported (See 7.4) then the RequestedStatesSupported property
390 value shall contain the following values: 2 (Enabled), 3 (Disabled), or 11 (Reset). If WBEM protocol
391 management (See 7.4) is not supported, then the RequestedStatesSupported property shall be Null.

392 7.4.1.3 CIM_WBEMProtocolServiceCapabilities.GenericOperationsSupported

393 The GenericOperationsSupported property is mandatory for all protocols. The property value is an array
394 of CIM_GenericOperationCapabilitiesStructure. The following subclauses describe implementation
395 requirements for CIM_GenericOperationCapabilitiesStructure.

396 7.4.1.3.1 CIM_GenericOperationCapabilitiesStructure.ContinueOnErrorSupported

397 The CIM_GenericOperationCapabilitiesStructure.ContinueOnErrorSupported property is mandatory when
398 the CIM_GenericOperationCapabilitiesStructure.OperationsSupported property includes any of the
399 following values 12(OpenClassInstancesWithPath), 13(OpenClassInstancePaths),
400 14(OpenAssociatedInstancesWithPath), 15(OpenAssociatedInstancePaths),
401 16(OpenReferencingInstancesWithPath), 17(OpenReferencingInstancePaths), 18(OpenQueryInstances),
402 19(PullInstancesWithPath), 20 (PullInstancePaths), 21 (PullInstances).

403 7.4.1.3.2 CIM_GenericOperationCapabilitiesStructure.MinimumOperationTimeout

404 The CIM_GenericOperationCapabilitiesStructure.MinimumOperationTimeout property is mandatory when
405 the CIM_GenericOperationCapabilitiesStructure.OperationsSupported property includes any of the
406 following values 12(OpenClassInstancesWithPath), 13(OpenClassInstancePaths),
407 14(OpenAssociatedInstancesWithPath), 15(OpenAssociatedInstancePaths),
408 16(OpenReferencingInstancesWithPath), 17(OpenReferencingInstancePaths), 18(OpenQueryInstances),
409 19(PullInstancesWithPath), 20 (PullInstancePaths), 21 (PullInstances).

410 7.4.1.3.3 CIM_GenericOperationCapabilitiesStructure.MaximumOperationTimeout

411 The CIM_GenericOperationCapabilitiesStructure.MaximumOperationTimeout property is mandatory when
412 the CIM_GenericOperationCapabilitiesStructure.OperationsSupported property includes any of the
413 following values 12(OpenClassInstancesWithPath), 13(OpenClassInstancePaths),
414 14(OpenAssociatedInstancesWithPath), 15(OpenAssociatedInstancePaths),
415 16(OpenReferencingInstancesWithPath), 17(OpenReferencingInstancePaths), 18(OpenQueryInstances),
416 19(PullInstancesWithPath), 20 (PullInstancePaths), 21 (PullInstances).

417 7.4.1.4 Protocol-specific capabilities

418 A WBEM protocol may have a subclass of CIM_WBEMProtocolServiceCapabilities to define protocol
419 specific capabilities. The following clauses describe when a subclass may be required instead.

420 7.4.1.4.1 CIM-XML capabilities

421 If the CIM_WBEMProtocolServiceCapabilities is implemented and the associated
422 CIM_ProtocolService.Protocol has the value 5 (CIM-XML), CIM_CIMXMLCapabilities (subclass of
423 CIM_WBEMProtocolServiceCapabilities) shall be supported and associated via CIM_ElementCapabilities.

424 7.4.1.4.2 WS-Management capabilities

425 If the CIM_WBEMProtocolServiceCapabilities is implemented and the associated
426 CIM_ProtocolService.Protocol has the value 6 (WS-Management), CIM_WSMManagementCapabilities
427 (subclass of CIM_WBEMProtocolServiceCapabilities) shall be supported and associated via
428 CIM_ElementCapabilities.

429 7.4.2 CIM_TCPProtocolEndpoint

430 Support for CIM_TCPProtocolEndpoint is conditional. If WBEM protocol management is supported (see
431 7.4), then CIM_TCPProtocolEndpoint and its two associations CIM_HostedAccessPoint and
432 CIM_ServiceAccessBySAP are Mandatory. If WBEM protocol management is not supported (see 7.4),

433 support for CIM_TCPProtocolEndpoint, CIM_HostedAccessPoint and CIM_ServiceAccessBySAP is
434 optional.

435 7.5 Indications Profile

436 *Indications Profile* ([DSP1054](#)) support is optional. If the Indications Profile is implemented, the
437 CIM_HostedService association shall be implemented with the Antecedent property referencing the
438 central instance of this profile and the Dependent referencing the central instance of the Indications
439 Profile.

440 The WBEM server related requirements defined in the *Indications Profile* ([DSP1054](#)) shall be
441 implemented.

442 8 Methods

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

445 8.1 Profile conventions for operations

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

448 The default list of operations is as follows:

- 449 • GetInstance
- 450 • Associators
- 451 • AssociatorNames
- 452 • References
- 453 • ReferenceNames
- 454 • EnumerateInstances
- 455 • EnumerateInstanceNames

456 8.2 CIM_ComputerSystem

457 All operations in the default list in 8.1 shall be implemented as defined in [DSP0200](#).

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

459 8.3 CIM_WBEMServer

460 All operations in the default list in 8.1 shall be implemented as defined in [DSP0200](#).

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

462 The following additional methods shall be implemented.

463 8.3.1 RequestStateChange()

464 A WBEM server may be requested to reboot, reset or shut down. A WBEM server reboot will shut down
465 and re-enable the WBEM server. Resetting a WBEM server means that the WBEM server goes into the
466 disabled state and then back to enabled state. Shutdown means that the WBEM server stops executing,
467 usually meaning that any cleanup needed is completed as part of the shutdown process. The WBEM

468 server shall respond to the method (for example, if shut down is requested, the server will respond to the
 469 method prior to shutting down, this will require the WBEM server to ensure that is can shut down before it
 470 responds).

471 A WBEM server shall not return 4096 (i.e. shall not utilize jobs, See Table 2) when shut down is
 472 requested. A WBEM server may utilize a job for reset or reboot. If the method utilizes a job, it shall
 473 support the *Job Control Profile* ([DSP1103](#)).

474 A WBEM server shall be unavailable after a shutdown (until started again). A WBEM server may be
 475 unavailable during a reboot or reset. The amount of time the WBEM server is unavailable is up to the
 476 implementation.

477 Support for the RequestStateChange() method is conditional; if the
 478 CIM_WBEMServerCapabilities.OperationsSupported property value includes the value 2
 479 (RequestStateChange) then the RequestStateChange() method shall be supported. The
 480 RequestStateChange() method may support the RequestedState parameter with a value of 4 (Shut
 481 Down), 9 (Reboot) or 10 (Reset). The method shall support the values in
 482 CIM_WBEMServerCapabilities.RequestedStatesSupported property value for the RequestedState
 483 parameter.

484 The permissible return code values and parameters for the RequestStateChange() method are specified
 485 in Table 2 and Table 3, respectively.

486 **Table 2 – RequestStateChange() method: Return code values**

Value	Description
0 (Completed with No Error)	The method executed successfully.
3 (Cannot complete within Timeout Period)	The requested amount of time is less than the time the requested state transition takes.
4 (Failed)	The method failed.
5 (Invalid Parameters)	One or more parameters are invalid.
6 (In Use)	Another client has requested a state change that has not completed.
4096 (Method Parameters Checked – Job Started)	The method parameters were validated and a job has been started.
4097 (Invalid State Transition)	The state change requested is invalid for the current state.
4098 (Use of Timeout Parameter Not Supported)	This implementation does not support the TimeoutPeriod parameter. A client may pass Null for the TimeoutPeriod and try again.
4099 (Busy)	A state change is underway in the job; as such, the state cannot be changed. An implementation may use this return code to indicate the job cannot be suspended, killed, or terminated at all or in the current phase of execution.

487 **Table 3 – RequestStateChange() method: Parameters**

Qualifiers	Name	Type	Description
IN	RequestedState	uint16	For the purposes of this profile, the valid states that can be requested are 9 (Reboot), 10 (Reset), and 4 (Shut Down). Each value is optional to be supported.

Qualifiers	Name	Type	Description
IN	TimeoutPeriod	datetime	A timeout period that specifies the maximum amount of time that the client expects the transition to the new state to take. The interval format must be used to specify the TimeoutPeriod. A value of 0 or a Null parameter indicates that the client has no time requirements for the transition. If this property does not contain 0 or Null and the implementation does not support this parameter, a return code of 4098 (Use of Timeout Parameter Not Supported) shall be returned.

488 **8.3.2 CreateWBEMServerNamespace()**

489 Support for CreateWBEMServerNamespace() is conditional, if the CIM_WBEMServerCapabilities
 490 .OperationsSupported property value includes the value 3 (CreateNamespace) then this method shall be
 491 supported.

492 CreateWBEMServerNamespace() creates a namespace. Upon successful completion of the method
 493 (return value 0 (Completed with No Error)), there shall be a newly created instance of
 494 CIM_WBEMServerNamespace associated through the CIM_HostedDependency association with this
 495 instance of CIM_WBEMServer and a new namespace shall be created. The method shall return 2
 496 (Failed) if a known error occurred (e.g. namespace already exists).

497 **Table 4 – CreateWBEMServerNamespace() method: Return code values**

Value	Description
0 (Completed with No Error)	The method executed successfully.
1 (Failed)	The method failed.

498 **Table 5 – CreateWBEMServerNamespace() method: Parameters**

Qualifiers	Name	Type	Description
IN	NamespaceTemplate	String	NamespaceTemplate is an instance of CIM_WBEMServerNamespace with the desired property values for the namespace to be created.
OUT	Namespace	REF	Reference to the instance of CIM_WBEMServerNamespace created when the method returns a value of 0.
OUT	Errors[]	String	An array of CIM_Error instances. If the error fails, at least one CIM_Error instance shall exist

500 **8.4 CIM_WBEMServerNamespace**

501 Table 6 lists implementation requirements for operations. If implemented, these operations shall be
 502 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 6, all operations in
 503 the default list in 8.1 shall be implemented as defined in [DSP0200](#).

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

505 **Table 6 – Operations: CIM_HostedDependency**

Operation	Requirement	Messages
ModifyInstance	Conditional	None

Operation	Requirement	Messages
DeleteInstance	Conditional	None

506 8.4.1 ModifyInstance

507 Support for the ModifyInstance operation is conditional, if the
 508 CIM_WBEMServerCapabilities.OperationsSupported property value includes the value 5
 509 (ModifyNamespace) then the ModifyInstance operation shall be supported.

510 The Name and InstanceID properties shall not be modified, only the SchemaInformation property value
 511 shall support modification.

512 8.4.2 DeleteInstance

513 Support for the DeleteInstance operation is conditional, if the
 514 CIM_WBEMServerCapabilities.OperationsSupported property value includes the value 4
 515 (DeleteNamespace) then the DeleteInstance operation shall be supported.

516 The namespace being deleted may include qualifier types, classes and instances. All of the information
 517 shall be deleted when this operation is executed. The managed environment that is represented in the
 518 namespace shall not be affected.

519 8.5 CIM_ProtocolService

520 All operations in the default list in 8.1 shall be implemented as defined in [DSP0200](#).

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

522 The following additional methods shall be implemented.

523 8.5.1 RequestStateChange()

524 A WBEM protocol may be “enabled”, “disabled” or “reset”.

525 The RequestStateChange() method is conditional. If WBEMServerCapabilities.OperationsSupported has
 526 a value of 2 (RequestStateChange), then the RequestStateChange() method is mandatory. The
 527 RequestStateChange() method shall support the RequestedState parameter with a value of 2 (Enabled),
 528 3 (Disabled) or 11 (Reset). 4 (Shut Down) and 10 (Reboot) are optional.

529 The permissible return code values and parameters for the RequestStateChange() method are specified
 530 in Table 7 and Table 8, respectively.

531 **Table 7 – RequestStateChange() method: Return code values**

Value	Description
0 (Completed with No Error)	The method executed successfully.
3 (Cannot complete within Timeout Period)	The requested amount of time is less than the time the requested state transition takes.
4 (Failed)	The method failed.
5 (Invalid Parameters)	One or more parameters are invalid.
6 (In Use)	Another client has requested a state change that has not completed.

Value	Description
4096 (Method Parameters Checked – Job Started)	The method parameters were validated and a job has been started.
4097 (Invalid State Transition)	The state change requested is invalid for the current state.
4098 (Use of Timeout Parameter Not Supported)	This implementation does not support the TimeoutPeriod parameter. A client may pass Null for the TimeoutPeriod and try again. There is no mechanism to determine what state changes are supported by a particular implementation
4099 (Busy)	A state change is underway in the job; as such, the state cannot be changed. An implementation may use this return code to indicate the job cannot be suspended, killed, or terminated at all or in the current phase of execution.

532

Table 8 – RequestStateChange() method: Parameters

Qualifiers	Name	Type	Description
IN	RequestedState	uint16	For the purposes of this profile, the valid states can be requested as defined in this subclause.
IN	TimeoutPeriod	datetime	A timeout period that specifies the maximum amount of time that the client expects the transition to the new state to take. The interval format must be used to specify the TimeoutPeriod. A value of 0 or a Null parameter indicates that the client has no time requirements for the transition. If this property does not contain 0 or Null and the implementation does not support this parameter, a return code of 4098 (Use of Timeout Parameter Not Supported) shall be returned.

533

534 **8.5.2 ListenOnPortIF()**

535 The ListenOnPortIF() method shall be supported when the
 536 CIM_WBEMProtocolServiceCapabilities.ListeningPortManagementSupported property has a value of
 537 True.

538 The ListenOnPortIF() method is used to configure ports on which the protocol represented by the
 539 CIM_ProtocolService instance will listen. Detailed requirements of the ListenOnPortIF() method are
 540 specified in Table 9 and Table 10.

541

Table 9 – ListenOnPortIF() method: Return code values

Return Code Values	Description
0	Request was successfully executed.
1	Method is unsupported in the implementation.
2	Error occurred
0x1000	Job started: REF returned to started CIM_ConcreteJob

542

543

Table 10 – ListenOnPortIF() method: Parameters

Qualifiers	Name	Type	Description/Values
IN	IPEndpoint	CIM_IPProtocolEndpoint REF	Optional reference to the specific CIM_IPProtocolEndpoint instance to which the created CIM_TCPProtocolEndpoint instance will be bound
OUT	TCPEndpoint	CIM_TCPProtocolEndpoint REF	CIM_TCPProtocolEndpoint instance that is created if the method is successful
IN, REQ	PortNumber	uint16	Desired port number for the service to listen on
IN, REQ	ProtocolIFType	uint32	Desired IF Type(e.g. HTTPS, HTTP ...) to listen for

544

545 **8.6 CIM_HostedDependency**

546 Table 11 lists implementation requirements for operations. If implemented, these operations shall be
 547 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 11, all operations
 548 in the default list in 8.1 shall be implemented as defined in [DSP0200](#).

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

550

Table 11 – Operations: CIM_HostedDependency

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

551 **8.7 CIM_ServiceServiceDependency**

552 Table 12 lists implementation requirements for operations. If implemented, these operations shall be
 553 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 12, all operations
 554 in the default list in 8.1 shall be implemented as defined in [DSP0200](#).

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

556

Table 12 – Operations: CIM_ServiceServiceDependency

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

557 **8.8 CIM_HostedService (WBEMServer)**

558 Table 13 lists implementation requirements for operations. If implemented, these operations shall be
 559 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 13, all operations
 560 in the default list in 8.1 shall be implemented as defined in [DSP0200](#)

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

562 **Table 13 – Operations: CIM_HostedService (WBEMServer)**

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

563 **8.9 CIM_HostedService (ProtocolService)**

564 Table 14 lists implementation requirements for operations. If implemented, these operations shall be
 565 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 14, all operations
 566 in the default list in 8.1 shall be implemented as defined in [DSP0200](#)

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

568 **Table 14 – Operations: CIM_HostedService (ProtocolService)**

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

569 **8.10 CIM_WBEMServerCapabilities**

570 All operations in the default list in 8.1 shall be implemented as defined in [DSP0200](#).

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

572 **8.11 CIM_ElementCapabilities (WBEMServerCapabilities)**

573 Table 16 lists implementation requirements for operations. If implemented, these operations shall be
 574 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 15, all operations
 575 in the default list in 8.1 shall be implemented as defined in [DSP0200](#)

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

577 **Table 15 – Operations: CIM_ElementCapabilities (WBEMServerCapabilities)**

Operation	Requirement	Messages
Associators	Unspecified	None
AssociatorNames	Unspecified	None

Operation	Requirement	Messages
References	Unspecified	None
ReferenceNames	Unspecified	None

578 **8.12 CIM_WBEMProtocolServiceCapabilities**

579 All operations in the default list in 8.1 shall be implemented as defined in [DSP0200](#).

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

581 **8.13 CIM_ElementCapabilities (WBEMProtocolServiceCapabilities)**

582 Table 16 lists implementation requirements for operations. If implemented, these operations shall be
 583 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 16, all operations
 584 in the default list in 8.1 shall be implemented as defined in [DSP0200](#)

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

586 **Table 16 – Operations: CIM_ElementCapabilities (WBEMProtocolServiceCapabilities)**

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

587 **8.14 CIM_TCPProtocolEndpoint**

588 All operations in the default list in 8.1 shall be implemented as defined in [DSP0200](#).

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

590 **8.15 CIM_ServiceAccessBySAP**

591 Table 17 lists implementation requirements for operations. If implemented, these operations shall be
 592 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 17, all operations
 593 in the default list in 8.1 shall be implemented as defined in [DSP0200](#)

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

595 **Table 17 – Operations: CIM_ServiceAccessBySAP**

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

596 **8.16 CIM_HostedAccessPoint (TCPProtocolEndpoint)**

597 Table 18 lists implementation requirements for operations. If implemented, these operations shall be
 598 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 18, all operations
 599 in the default list in 8.1 shall be implemented as defined in [DSP0200](#)

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

601 **Table 18 – Operations: CIM_HostedAccessPoint (TCPProtocolEndpoint)**

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

602 **8.17 CIM_BindsTo**

603 Table 19 lists implementation requirements for operations. If implemented, these operations shall be
 604 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 19, all operations
 605 in the default list in 8.1 shall be implemented as defined in [DSP0200](#)

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

607 **Table 19 – Operations: CIM_BindsTo**

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

608 **8.18 CIM_CIMXMLCapabilities**

609 All operations in the default list in 8.1 shall be implemented as defined in [DSP0200](#).

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

611 **8.19 CIM_WSMangementCapabilities**

612 All operations in the default list in 8.1 shall be implemented as defined in [DSP0200](#).

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

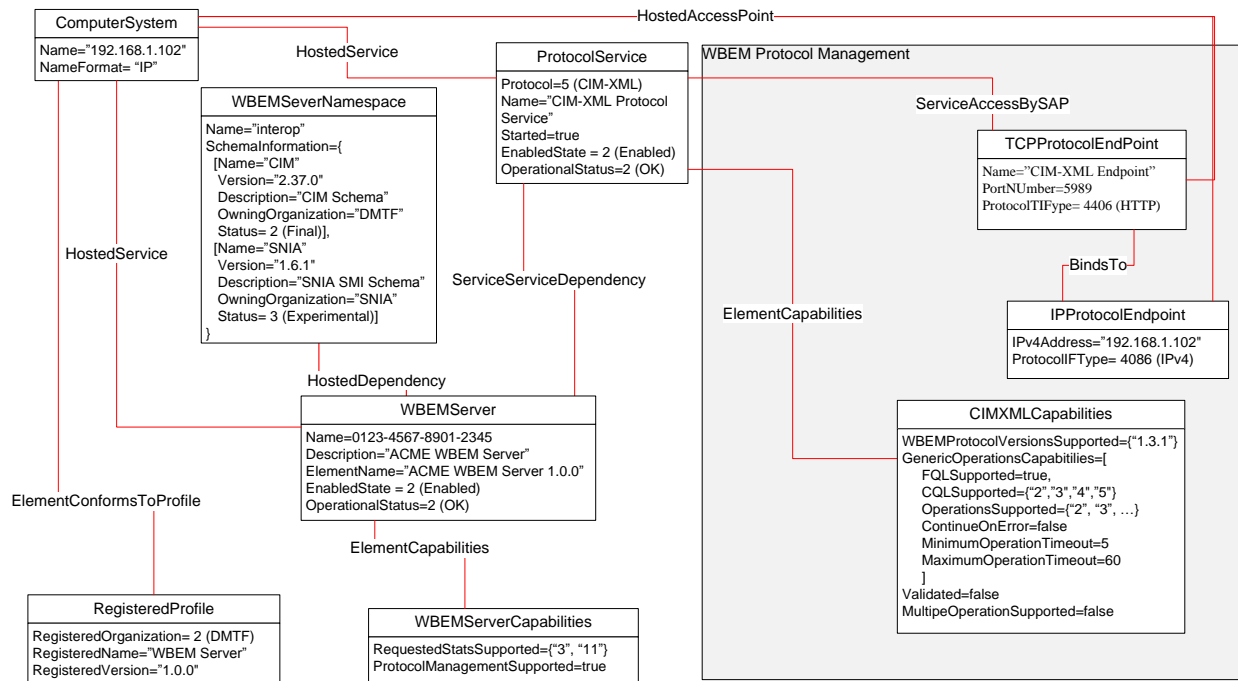
614

615 **9 Use cases (Informative)**

616 This clause provides informative use cases and object diagrams.

617 The following diagrams will be used to illustrate some of the use cases.

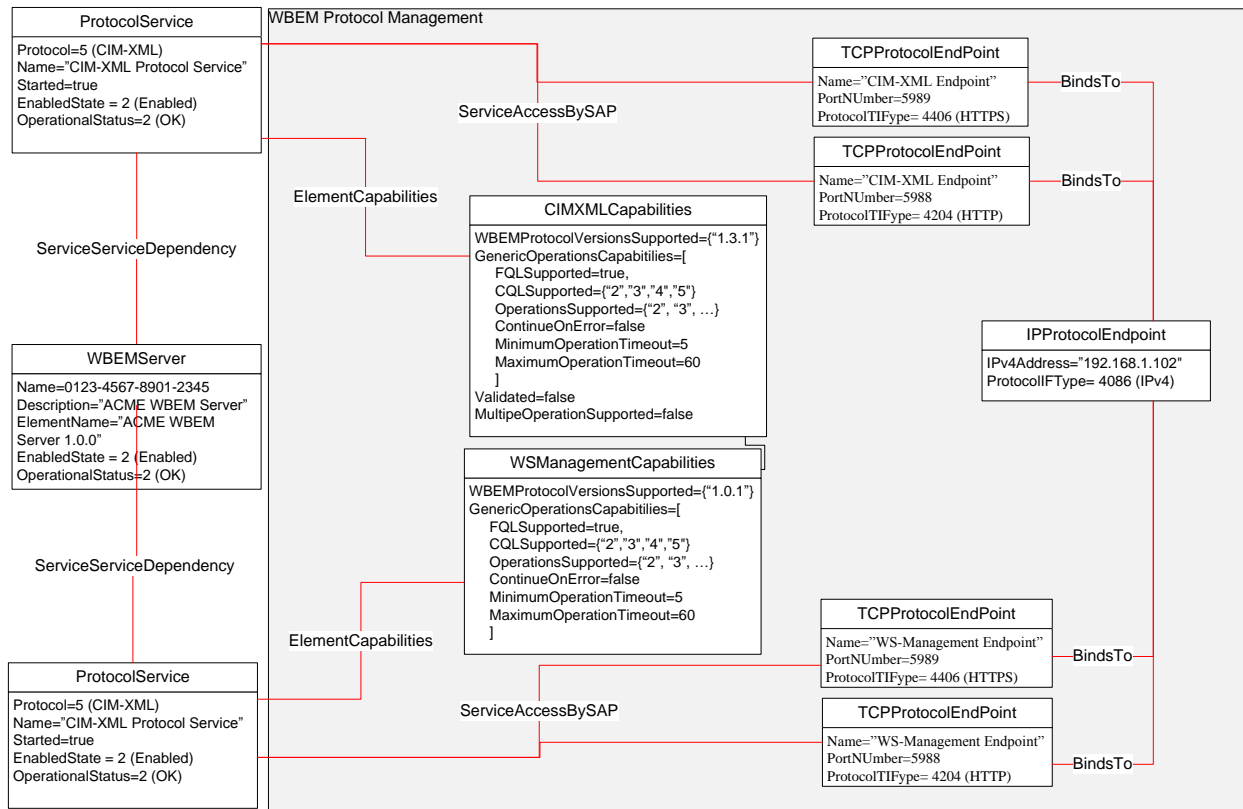
618 Figure 2 represents a WBEM server that supports a single namespace (interop) and a single WBEM
 619 protocol (CIM-XML). Only the relevant instances for the use case are shown, many of the mandatory
 620 instances are not in the diagram for readability.



621

622 **Figure 2 – WBEM Server Profile: Instance diagram**

623 Figure 3 represents an instance diagram of a WBEM server that supports two WBEM protocols (CIM-XML
 624 and WS-Management), each listening on two ports; one for HTTP and one for HTTPS. All the protocols
 625 are bound to the same IP Address. Only the relevant instances for the use case are shown, many of the
 626 mandatory instances are not in the diagram for readability.



627

628 **Figure 3 – WBEM Server Profile: Instance diagram with multiple WBEM protocols**

629

630 **9.1 Determine the namespaces of a WBEM server**

631 A WBEM server may contain one or more namespaces. The interop namespace is mandatory. A client
 632 may retrieve all namespaces supported and optionally information about each namespace.

- 633 1) Starting from the central instance of the profile, traverse the HostedService association (with a
 634 result class of CIM_WBEMServer) to get the instance that represents the WBEM server
- 635 2) From the CIM_WBEMServer instance, traverse the HostedDependency association to get the
 636 instance(s) of CIM_WBEMServerNamespace. Each instance represents a namespace that
 637 exists. The value of the name property is the name of the namespace.

638 **9.2 Determine the contents of a namespace**

639 A namespace may contain multiple schemas. The following steps will describe how a client determines
 640 the contents of a namespace.

- 641 1) Use 9.1 to retrieve the namespace instance(s)
- 642 2) The CIM_WBEMServerNamespace.SchemaInformation property contains an array of structures
 643 that represent the information in the namespace. A namespace may contain multiple schemas
 644 (e.g. CIM, SNIA, ACME). An example of what an entry may look like for the DMTF CIM Schema
 645 2.37.0 is below.

646

Property Name	Value
---------------	-------

Property Name	Value
Name	CIM
Version	2.37.0
Description	CIM 2.37.0 Schema
URI	http://schemas.dmtf.org/wbem/cim-schemas
OwningOrganization	DMTF
Status	2 (Final)

647

648 9.3 Modify WBEMServerNamespace to update what is represented

649 The schema contained in a namespace may change over time. For example, the schema may be
 650 upgraded to a newer version or a new extension schema is added. An installation program that installs
 651 new extension schema in a namespace should update the WBEMServerNamespace instance that
 652 represents the namespace to include the information for this new extension schema. The following steps
 653 describe one way to accomplish this task.

- 654 1) Use 9.1 to retrieve the namespace instance(s)
- 655 2) Find the instance that has the Name property value that matches the namespace
- 656 3) Using the instance from step 2, and a CIM_SchemaInformationStructure structure instance that
 657 represents the information for the extension schema added and use the ModifyInstance operation
 658 to apply the changes.

659 9.4 Determine the WBEM protocols supported and state

- 660 1) Starting from the central instance of the profile, traverse the CIM_HostedService association, with
 661 a result class of CIM_WBEMServer, to retrieve the instance that represents the WBEM server
- 662 2) Traverse the CIM_ServiceServiceDependency association starting from the CIM_WBEMServer
 663 instance with a result class of CIM_ProtocolService.
 - 664 a. For each instance returned, view the Protocol property value for the id that represents the
 665 name of the protocol supported.
 - 666 b. To determine the state of the protocol, view the EnabledState property value. For
 667 example, if the value is 2 (Enabled), the protocol is available to accept requests and
 668 issue responses.

669 9.5 Determine the port used for a WBEM protocol

670 Support for WBEM protocol management is optional.

- 671 1) Starting from the central instance of the profile, traverse the CIM_HostedService association (with
 672 a result class of CIM_WBEMServer) to retrieve the instance that represents the WBEM server
- 673 2) Traverse the CIM_ServiceServiceDependency association starting from the CIM_WBEMServer
 674 instance with a result class of CIM_ProtocolService to retrieve the instance(s) that represent the
 675 WBEM protocol(s) supported.
- 676 3) Traverse the CIM_ServiceAccessBySAP association with a result class of
 677 CIM_TCPProtocolEndpoint to retrieve the instance(s) that represent the endpoint and port.

678 9.6 Determine the IP address for a WBEM protocol

679 Support for WBEM protocol management is optional.

- 680 1) Starting from the central instance of the profile, traverse the CIM_HostedService association (with
681 a result class of CIM_WBEMServer) to retrieve the instance that represents the WBEM server
- 682 2) Traverse the CIM_ServiceServiceDependency association starting from the CIM_WBEMServer
683 instance with a result class of CIM_ProtocolService to retrieve the instance(s) that represent the
684 WBEM protocol(s) supported.
- 685 3) Traverse the CIM_ServiceAccessBySAP association with a result class of
686 CIM_TCIPProtocolEndpoint to retrieve the instance(s) that represent the endpoint and port. View
687 the CIM_TCIPProtocolEndpoint.PortNumber property to see the port that this endpoint is listening
688 on.
- 689 4) Traverse the CIM_BindTo association with a result class of CIM_IPProtocolEndpoint. The
690 instance(s) of CIM_IPProtocolEndpoint returned will have a value for either the IPv4Address
691 and/or IPv6Address properties.

692 9.7 Determine the capabilities of a WBEM protocol

- 693 1) Starting from the central instance of the profile, traverse the CIM_HostedService association (with
694 a result class of CIM_WBEMServer) to retrieve the instance that represents the WBEM server
- 695 2) Traverse the CIM_ServiceServiceDependency association starting from the CIM_WBEMServer
696 instance with a result class of CIM_ProtocolService to retrieve the instance(s) that represent the
697 WBEM protocol(s) supported. If a specific protocol is desired, filter on the
698 CIM_ProtocolService.Protocol property value.
- 699 3) Using the instance(s) from the previous step, traverse the CIM_ElementCapabilities association
700 with a result class of CIM_WBEMProtocolServiceCapabilities. The capabilities include the
701 version(s) of the protocol supported as well as the list of operations supported. Some protocols
702 may have subclasses of CIM_WBEMProtocolServiceCapabilities to provide profile specific
703 capabilities, for example CIM_CIMXMLCapabilities and CIM_WSMManagementCapabilities.

704 9.8 Modify the port for a WBEM protocol

- 705 1) Starting from the central instance of the profile, traverse the CIM_HostedService association (with
706 a result class of CIM_WBEMServer) to retrieve the instance that represents the WBEM server
- 707 2) Traverse the CIM_ServiceServiceDependency association starting from the CIM_WBEMServer
708 instance with a result class of CIM_ProtocolService to retrieve the instance(s) that represent the
709 WBEM protocol(s) supported. . If a specific protocol is desired, filter on the
710 CIM_ProtocolService.Protocol property value.
- 711 3) Using the instance(s) from the previous step, traverse the CIM_ElementCapabilities association
712 with a result class of CIM_WBEMProtocolServiceCapabilities. Support for WBEM protocol
713 management is optional. If the value of the
714 CIM_WBEMProtocolServiceCapabilities.ListeningPortManagementSupported is True, port
715 modification is supported.
- 716 4) Using the CIM_ProtocolService instance(s) from step 2, invoke the ListenOnPortIF() method with
717 the desired port. An implementation may allow listening on multiple ports or only a single port per
718 protocol interface (e.g. HTTP, HTTPS, ...). In other words, if you invoke this method, it may
719 remove the old port and replace it with the new port or may add the new port as an additional
720 port. To determine this, a client can follow the steps in 9.5.

721 9.9 Disable/Enable a WBEM protocol

722 A WBEM server may support one or more protocols. An administrator may enable or disable a WBEM
723 protocol. The following describe the steps to enable a WBEM protocol. To disable a protocol, follow the
724 same steps changing the RequestedState parameter to 3 (Disable).

- 725 1) Starting from the central instance of the profile, traverse the CIM_HostedService association (with
726 a result class of CIM_WBEMServer) to retrieve the instance that represents the WBEM server

- 727 2) Traverse the CIM_ServiceServiceDependency association starting from the CIM_WBEMServer
728 instance with a result class of CIM_ProtocolService to retrieve the instance(s) that represent the
729 WBEM protocol(s) supported. . If a specific protocol is desired, filter on the
730 CIM_ProtocolService.Protocol property value.
- 731 3) Support for the RequestStateChange() method is conditional. A client can determine if the
732 RequestStateChange() method is supported by using the instance(s) from the previous step,
733 traverse the CIM_ElementCapabilities association with a result class of
734 CIM_WBEMProtocolServiceCapabilities and determine if the value of the
735 CIM_WBEMProtocolServiceCapabilities.RequestedStatesSupported property has the value of the
736 desired RequestedState.
- 737 4) Using the instance(s) from Step 2, invoke the CIM_ProtocolService.RequestStateChange()
738 method with the RequestedState parameter value set to 2 (Enable).
- 739 5) If the return value of the method is 0 (Completed with No Error), then the protocol was enabled. If
740 the return value of the method is 4096, a job was started – see the *Job Control Profile* ([DSP1103](#))
741 for information on jobs.

742 9.10 Reset the WBEM server

743 The WBEM server may be reset using the following steps.

- 744 1) Starting from the central instance of the profile, traverse the CIM_HostedService association (with
745 a result class of CIM_WBEMServer) to get the instance that represents the WBEM server
- 746 2) Using the instance of CIM_WBEMServer traverse the CIM_ElementCapabilities association with
747 a result class of CIM_WBEMServerCapabilities
- 748 3) This behavior is optional. If the CIM_WBEMServerCapabilities.RequestedStatesSupported
749 includes the value 11 (Reset), the server supports being reset.
- 750 4) Invoke the CIM_WBEMServer.RequestStateChange() method with the RequestedState
751 parameter value set to 11 (Reset).
- 752 5) If the return value of the method is 0 (Completed with No Error), then the WBEM server will shut
753 down (it may take some amount of time for the WBEM server to shut down all services) and
754 restart. If the return value of the method is 4096, a job was started – see the *Job Control Profile*
755 ([DSP1103](#)) for information on jobs.

756 9.11 Shut down the WBEM server

757 The WBEM server may be shut down using the following steps.

- 758 1) Starting from the central instance of the profile, traverse the CIM_HostedService association (with
759 a result class of CIM_WBEMServer) to get the instance that represents the WBEM server
- 760 2) Using the instance of CIM_WBEMServer traverse the CIM_ElementCapabilities association with
761 a result class of CIM_WBEMServerCapabilities
- 762 3) This behavior is optional. If the CIM_WBEMServerCapabilities.RequestedStatesSupported
763 includes the value 4 (Shut Down), the server supports being shut down.
- 764 4) Invoke the RequestStateChange() method with the RequestedState parameter value set to 4
765 (Shut Down).
- 766 5) If the return value of the method is 0 (Completed with No Error), then the WBEM server will shut
767 down (it may take some amount of time for the WBEM server to shut down all services). If the
768 return value of the method is 4096, a job was started – see the *Job Control Profile* ([DSP1103](#)) for
769 information on jobs.

770 **10 CIM elements**

771 Table 20 shows the instances of CIM elements for this profile. Instances of the CIM elements shall be
 772 implemented as described in Table 20. Clauses 7 (“Implementation”) and 8 (“Methods”) may impose
 773 additional requirements on these elements.

774 **Table 20 – CIM Elements: WBEM Server Profile**

Element name	Requirement	Description
Classes		
CIM_BindsTo	Optional	See 10.1.
CIM_CIMXMLCapabilities	Conditional	See 10.2 and 7.4.1.4.1.
CIM_ComputerSystem	Mandatory	See 10.3.
CIM_ConcreteJob	Optional	See 10.4.
CIM_ElementCapabilities (WBEMServerCapabilities)	Mandatory	See 10.5.
CIM_ElementCapabilities (WBEMProtocolServiceCapabilities)	Conditional	See 10.6 and 7.4.1.
CIM_ElementConformsToProfile	Mandatory	See 10.7.
CIM_GenericOperationCapabilitiesStructure	See embedding element.	See 10.8.
CIM_HostedAccessPoint	Conditional	See 10.9 and 7.4.2.
CIM_HostedAccessPoint (TCPProtocolEndpoint)	Conditional	See 10.10.
CIM_HostedDependency	Mandatory	See 10.11.
CIM_HostedService (WBEMServer)	Mandatory	See 10.12.
CIM_HostedService (ProtocolService)	Mandatory	See 10.13.
CIM_IPProtocolEndPoint	Optional	See 10.14.
CIM_ProtocolService	Mandatory	See 10.15.
CIM_RegisteredProfile	Mandatory	See 10.16.
CIM_SchemaInformationStructure	See embedding element.	See 10.17.
CIM_ServiceAccessBySAP	Conditional	See 10.18 and 7.4.2.
CIM_ServiceServiceDependency	Mandatory	See 10.19.
CIM_TCPProtocolEndpoint	Conditional	See 10.20 and 7.4.2.
CIM_WBEMProtocolServiceCapabilities	Conditional	See 10.21 and 7.4.1.
CIM_WBEMServer	Mandatory	See 10.22.
CIM_WBEMServerCapabilities	Mandatory	See 10.23.
CIM_WBEMServerNamespace	Mandatory	See 10.24.
CIM_WSMManagementCapabilities	Conditional	See 10.25 and 7.4.1.4.2.
Indications		
None		

775 **10.1 CIM_BindsTo**

776 CIM_BindsTo represents an association between a ProtocolEndPoint and a ServiceAccessPoint. Table
777 21 contains the requirements for elements of this class.

778 **Table 21 – Class: CIM_BindsTo**

Elements	Requirement	Notes
Antecedent	Mandatory	Key: Shall reference the instance of CIM_ProtocolEndPoint.
Dependent	Mandatory	Key: Shall reference the instance of CIM_ServiceAccessPoint.

779 **10.2 CIM_CIMXMLCapabilities**

780 CIM_BindsTo represents an association between a ProtocolEndPoint and a ServiceAccessPoint. Table
781 21 contains the requirements for elements of this class.

782 **Table 22 – Class: CIM_CIMXMLCapabilities**

Elements	Requirement	Notes
InstanceID	Mandatory	Key
ElementName	Mandatory	None
RequestedStatesSupported	Mandatory	See 7.4.1.2
MaxConnections	Mandatory	A value of 0 (zero) shall indicate unknown
ProtocolVersionSupported	Mandatory	None
GenericOperationCapabilities	Mandatory	See 10.8
AuthenticationMechansimsSupported	Mandatory	None
MultipleOperationsSupported	Mandatory	None
Validated	Mandatory	None

783

784 **10.3 CIM_ComputerSystem**

785 The requirements denoted in Table 23 are in addition to those mandated by the *Computer System Profile*
786 ([DSP1052](#)).

787 **Table 23 – Class: CIM_ComputerSystem**

Elements	Requirement	Notes
CreationClassName	Mandatory	Key
Name	Mandatory	Key
NameFormat	Mandatory	None

788 **10.4 CIM_ConcreteJob**

789 There are no additional requirements than to those mandated by the *Job Control Profile* ([DSP1103](#)).

790 **10.5 CIM_ElementCapabilities (WBEMServerCapabilities)**

791 CIM_ElementCapabilities represents an association between a CIM_WBEMServer and its capabilities.
 792 Table 24 contains the requirements for elements of this class.

793 **Table 24 – Class: CIM_ElementCapabilities (WBEMServerCapabilities)**

Elements	Requirement	Notes
ManagedElement	Mandatory	Key: Shall reference the instance of CIM_WBEMServer.
Capabilities	Mandatory	Key: Shall reference the instance of CIM_WBEMServerCapabilities.
Characteristics	Mandatory	Matches 3

794 **10.6 CIM_ElementCapabilities (WBEMProtocolServiceCapabilities)**

795 CIM_ElementCapabilities represents an association between a CIM_ProtocolService and its capabilities.
 796 Table 25 contains the requirements for elements of this class.

797 **Table 25 – Class: CIM_ElementCapabilities (WBEMProtocolServiceCapabilities)**

Elements	Requirement	Notes
ManagedElement	Mandatory	Key: Shall reference the instance of CIM_ProtocolService.
Capabilities	Mandatory	Key: Shall reference the instance of CIM_WBEMProtocolServiceCapabilities.
Characteristics	Mandatory	Matches 3

798 **10.7 CIM_ElementConformsToProfile**

799 The requirements denoted in Table 26 are in addition to those mandated by the *Profile Registration*
 800 *Profile* ([DSP1033](#)).

801 **Table 26 – Class: CIM_ElementConformsToProfile**

Elements	Requirement	Notes
ManagedElement	Mandatory	Key: Shall reference the instance of CIM_ComputerSystem.
ConformantStandard	Mandatory	Key: Shall reference the instance of CIM_RegisteredProfile.

802

803 10.8 CIM_GenericOperationCapabilitiesStructure

804 The CIM_GenericOperationCapabilitiesStructure is a structure that describes the capabilities for Generic
805 Operations based on the *Generic Operations* ([DSP0223](#)) specification.

806 **Table 27 – Class: CIM_GenericOperationCapabilitiesStructure**

Elements	Requirement	Notes
FQLSupported	Mandatory	None
CQLSupport	Mandatory	None
OperationsSupported	Mandatory	None
OperationsSupportedDescriptions	Conditional	Mandatory when OperationsSupported has 1 (Other)
ContinueOnErrorSupported	Conditional	See 7.4.1.3.1
MinimumOperationTimeout	Conditional	See 7.4.1.3.2
MaximumOperationTimeout	Conditional	See 7.4.1.3.3

807 10.9 CIM_HostedAccessPoint

808 There are no additional requirements than to those mandated by the *IP Interface Profile* ([DSP1036](#)).

809 10.10 CIM_HostedAccessPoint (TCPProtocolEndpoint)

810 CIM_HostedAccessPoint represents an association between the computer system and the TP protocol
811 endpoints. Table 28 contains the requirements for elements of this class.

812 **Table 28 – Class: CIM_HostedAccessPoint (TCPProtocolEndpoint)**

Elements	Requirement	Notes
Antecedent	Mandatory	Key: Shall reference the instance of CIM_ComputerSystem (the central instance of this profile).
Dependent	Mandatory	Key: Shall reference the instance of CIM_TCPProtocolEndpoint.

813

814 10.11 CIM_HostedDependency

815 CIM_HostedDependency associates the CIM_WBEMServerNamespace instances with the
816 CIM_WBEMServer. Table 29 contains the requirements for elements of this class.

817 **Table 29 – Class: CIM_HostedDependency**

Elements	Requirement	Notes
Antecedent	Mandatory	Key: Shall reference the instance of CIM_WBEMServer
Dependent	Mandatory	Key: Shall reference the instance of CIM_WBEMServerNamespace.

818 **10.12 CIM_HostedService (WBEMServer)**

819 CIM_HostedService (WBEMServer) associates the CIM_ComputerSystem and CIM_WBEMServer.
 820 Table 30 contains the requirements for elements of this class.

821 **Table 30 – Class: CIM_HostedService (WBEMServer)**

Elements	Requirement	Notes
Antecedent	Mandatory	Key: Shall reference the instance of CIM_ComputerSystem (the central instance of this profile).
Dependent	Mandatory	Key: Shall reference the instance of CIM_WBEMServer.

822 **10.13 CIM_HostedService (ProtocolService)**

823 CIM_HostedService (ProtocolService) associates the CIM_ComputerSystem and CIM_ProtocolService.
 824 Table 31 contains the requirements for elements of this class.

825 **Table 31 – Class: CIM_HostedService (ProtocolService)**

Elements	Requirement	Notes
Antecedent	Mandatory	Key: Shall reference the instance of CIM_ComputerSystem (the central instance of this profile).
Dependent	Mandatory	Key: Shall reference the instance of CIM_ProtocolService.

826 **10.14 CIM_IPProtocolEndpoint**

827 There are no additional requirements than to those mandated by the *IP Interface Profile* ([DSP1036](#)).

828 **10.15 CIM_ProtocolService**

829 CIM_ProtocolService represents a WBEM protocol. Table 32 contains the requirements for elements of
 830 this class.

831 **Table 32 – Class: CIM_ProtocolService**

Elements	Requirement	Notes
SystemCreationClassName	Mandatory	Key
SystemName	Mandatory	Key
CreationClassName	Mandatory	Key
Name	Mandatory	Key
Protocol	Mandatory	None
MaxConnections	Mandatory	A value of 0 (zero) shall indicate unknown
RequestedState	Mandatory	See 7.3.1
EnabledState	Mandatory	See 7.3.2

Elements	Requirement	Notes
HealthState	Mandatory	None
OperationalStatus	Mandatory	None
ElementName	Mandatory	None
RequestStateChange()	Conditional	See 8.5.1
ListenOnPortIF()	Conditional	See 8.5.2

832 **10.16 CIM_RegisteredProfile**

833 The requirements denoted in Table 33 are in addition to those mandated by the *Profile Registration*
 834 *Profile* ([DSP1033](#)).

835 **Table 33 – Class: CIM_RegisteredProfile**

Properties	Requirement	Notes
RegisteredName	Mandatory	This property shall have a value of "WBEM Server".
RegisteredVersion	Mandatory	This property shall have a value of "1.0.0h".
RegisteredOrganization	Mandatory	This property shall have a value of 2 (DMTF).

836 **10.17 CIM_SchemaInformationStructure**

837 The CIM_SchemaInformationStructure is a structure that describes the schema information that may be
 838 contained in a namespace.

839 **Table 34 – Class: CIM_SchemaInformationStructure**

Elements	Requirement	Notes
Name	Mandatory	None
Version	Mandatory	None
Description	Optional	None
URI	Optional	None
OwningOrganization	Mandatory	None
Status	Optional	None

840 **10.18 CIM_ServiceAccessBySAP**

841 CIM_ServiceAccessBySAP represents an association between a CIM_ProtocolService and the
 842 CIM_ServiceAccessPoint(s). Table 35 contains the requirements for elements of this class.

843 **Table 35 – Class: CIM_ServiceAccessBySAP**

Elements	Requirement	Notes
Antecedent	Mandatory	Key: Shall reference the instance of CIM_ProtocolService that has this CIM_ServiceAccessPoint.

Elements	Requirement	Notes
Dependent	Mandatory	Key: Shall reference the instance of CIM_ServiceAccessPoint.

844 **10.19 CIM_ServiceServiceDependency**

845 CIM_ServiceServiceDependency associates the CIM_WBEMServer instance with the
 846 CIM_ProtocolService instances. Table 36 contains the requirements for the elements of this class.

847 **Table 36 – Class: CIM_ServiceServiceDependency**

Elements	Requirement	Notes
Antecedent	Mandatory	Key: Shall reference the instance of CIM_WBEMServer.
Dependent	Mandatory	Key: Shall reference an instance of CIM_ProtocolService.
TypeOfDependency	Mandatory	Value shall be 3
RestartService	Mandatory	Value shall be False

848 **10.20 CIM_TCPProtocolEndpoint**

849 CIM_TCPProtocolEndpoint represents a protocol endpoint used for communications. Table 37 contains
 850 the requirements for elements of this class.

851 **Table 37 – Class: CIM_TCPProtocolEndpoint**

Elements	Requirement	Notes
Name	Mandatory	Key
CreationClassName	Mandatory	Key
SystemName	Mandatory	Key
SystemCreationClassName	Mandatory	Key
PortNumber	Mandatory	None
ProtocolIFTType	Mandatory	None
NameFormat	Mandatory	Pattern “. *
ElementName	Mandatory	Pattern “. *

852 **10.21 CIM_WBEMProtocolServiceCapabilities**

853 CIM_WBEMProtocolServiceCapabilities represents the capabilities for a CIM_ProtocolService. Table 38
 854 contains the requirements for elements of this class.

855 **Table 38 – Class: CIM_WBEMProtocolServiceCapabilities**

Elements	Requirement	Notes
InstanceID	Mandatory	Key
ElementName	Mandatory	None

Elements	Requirement	Notes
ListeningPortManagementSupported	Mandatory	See 7.4.1.1
RequestedStatesSupported	Mandatory	See 7.4.1.2
MaxConnections	Mandatory	A value of 0 (zero) shall indicate unknown
ProtocolVersionSupported	Mandatory	None
GenericOperationCapabilities	Mandatory	See 10.7
AuthenticationMechanismsSupported	Mandatory	None
AuthenticationMechanismsDescriptions	Conditional	Mandatory when AuthenticationMechanismsSupports has the value 1

856 **10.22 CIM_WBEMServer**

857 CIM_WBEMServer represents the WBEM server as a service running on a system. Table 39 contains the
858 requirements for elements of this class.

859 **Table 39 – Class: CIM_WBEMServer**

Elements	Requirement	Notes
SystemCreationClassName	Mandatory	Key
SystemName	Mandatory	Key
CreationClassName	Mandatory	Key
Name	Mandatory	Key
RequestedState	Mandatory	See 7.1.2
EnabledState	Mandatory	See 7.1.3
HealthState	Mandatory	None
OperationalStatus	Mandatory	None
ElementName	Mandatory	None
RequestStateChange()	Conditional	See 8.3.1
CreateWBEMServerNameSpace()	Conditional	See 8.3.2

860 **10.23 CIM_WBEMServerCapabilities**

861 CIM_WBEMServerCapabilities represents the capabilities for a WBEM server. Table 40 contains the
862 requirements for elements of this class.

863 **Table 40 – Class: CIM_WBEMServerCapabilities**

Elements	Requirement	Notes
InstanceID	Mandatory	Key
OperationsSupported	Optional	See 7.1.1.1
RequestedStatesSupported	Optional	See 7.1.1.2

864 **10.24 CIM_WBEMServerNamespace**

865 CIM_WBEMServerNamespace represents the namespaces of the WBEM server. Table 41 contains the
 866 requirements for elements of this class.

867 **Table 41 – Class: CIM_WBEMServerNamespace**

Elements	Requirement	Notes
InstanceID	Mandatory	Key
Name	Mandatory	See 7.2.1
SchemaInformation	Mandatory	See 10.17
ElementName	Mandatory	None

868 **10.25 CIM_WSMangementCapabilities**

869 CIM_BindsTo represents an association between a ProtocolEndPoint and a ServiceAccessPoint. Table
 870 21 contains the requirements for elements of this class.

871 **Table 42 – Class: CIM_WSMangementCapabilities**

Elements	Requirement	Notes
InstanceID	Mandatory	Key
ElementName	Mandatory	None
RequestedStatesSupported	Mandatory	See 7.4.1.2
MaxConnections	Mandatory	A value of 0 (zero) shall indicate unknown
ProtocolVersionSupported	Mandatory	None
GenericOperationCapabilities	Mandatory	See 10.7
AuthenticationMechansimsSupported	Mandatory	None
XPathFeatures	Mandatory	None

872

873
874
875
876
877

ANNEX A (informative)

Change log

Version	Date	Author	Description
1.0.0a	2013-01-29	Jim Davis	Released as Work in Progress
1.0.0g	2013-08-22	Jim Davis	Released as Work in Progress, with the following changes: <ul style="list-style-type: none">• Update SchemaInformationStructure table per WG discussion.

878