



1
2
3
4

Document Number: DSP1065

Date: 2014-02-14

Version: 1.0.0a

5
6

Network Management - Virtual Routing and Forwarding 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: 2014-06-31

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

7
8
9

Document Type: Specification

Document Status: Work in Progress

Document Language: en-US

10 Copyright Notice

11 Copyright © 2014 Distributed Management Task Force, Inc. (DMTF). All rights reserved.

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

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26 withdrawn or modified after publication, and shall be indemnified and held harmless by any party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

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

32

33

34

35

36

37

38

39

40

CONTENTS

42	Foreword	5
43	Introduction.....	6
44	1 Scope	7
45	2 Normative references	7
46	3 Terms and definitions	7
47	4 Symbols and abbreviated terms.....	8
48	5 Synopsis	9
49	6 Description	9
50	6.1 Class diagram	9
51	6.2 CIM_VirtualRoutingAndForwardingTable	10
52	6.3 CIM_VRFConfigurationService.....	10
53	6.4 CIM_System	11
54	6.5 CIM_ServiceAccessPoint	11
55	6.6 CIM_EthernetPort	11
56	6.7 CIM_NextHopIPRoute	11
57	7 Implementation.....	11
58	7.1 Representing the Virtual routing and forwarding table management capabilities	11
59	7.1.1 CIM_System	11
60	7.1.2 CIM_VRFConfigurationService.....	11
61	7.1.3 CIM_VirtualRoutingAndForwardingTable	12
62	7.1.4 CIM_NextHopIPRoute	12
63	7.1.5 CIM_EthernetPort	12
64	7.1.6 CIM_ServiceAccessPoint	12
65	8 Methods.....	12
66	8.1 Extrinsic Methods.....	12
67	8.1.2 CIM_VRFConfigurationService.CreateVRF()	13
68	8.1.3 CIM_VRFConfigurationService.RemoveVRF()	14
69	8.1.4 CIM_VRFConfigurationService.AddRoute()	15
70	8.1.5 CIM_VRFConfigurationService.RemoveRoute()	15
71	8.1.6 CIM_VRFConfigurationService.AddPortMember().....	16
72	8.1.7 CIM_VRFConfigurationService.RemovePortMember().....	17
73	8.1.8 CIM_VRFConfigurationService.AddSAPMember()	17
74	8.1.9 CIM_VRFConfigurationService.RemoveSAPMember()	18
75	8.2 Profile conventions for operations	18
76	8.3 CIM_HostedService	19
77	8.4 CIM_HostedCollection	19
78	8.5 CIM_ServiceAvalableToElement	20
79	8.6 CIM_ServiceAffectsElement	20
80	8.7 CIM_VRFRoute.....	20
81	8.8 CIM_MemberOfCollection	21
82	8.9 CIM_VRFConfigurationService.....	21
83	8.10 CIM_NextHopIPRoute	21
84	8.11 CIM_VirtualRoutingAndForwardingTable	21
85	8.12 CIM_System	21
86	9 Use cases.....	22
87	9.1 Profile Registration.....	22
88	9.2 VRF with Routes and Interfaces	23
89	9.3 VRF in a BGP router	24
90	10 CIM Elements.....	26
91	10.1 CIM_HostedService	26
92	10.2 CIM_VRFRoute.....	27
93	10.3 CIM_RegisteredProfile.....	27

94 10.4 CIM_VirtualForwardingAndRoutingTable 27
 95 10.5 CIM_NextHopRoute 28
 96 10.6 CIM_NextHopIPRoute 28
 97 ANNEX A (informative) Change log 29

98

99 **Figures**

100 Figure 1 – Network Management - Routing and Forwarding Profile: Class diagram 10
 101 Figure 2 – Registered profile with Computer System Profile 22
 102 Figure 3 – Registered profile BGPConfiguration Profile 22
 103

104 **Tables**

105 Table 1 – Referenced profiles 9
 106 Table 2 – CreateVRF() Method: Parameters 14
 107 Table 3 – RemoveVRF() Method: Parameters 14
 108 Table 4 – AddRoute() Method: Parameters 15
 109 Table 5 – RemoveRoute() Method: Parameters 16
 110 Table 6 – AddPortMember() Method: Parameters 16
 111 Table 7 – RemovePortMember() Method: Parameters 17
 112 Table 8 – AddSAPMember() Method: Parameters 17
 113 Table 9 – RemoveSAPMember() Method: Parameters 18
 114 Table 10 – Operations: CIM_HostedService 19
 115 Table 12 – Operations: CIM_ServiceAvailableToElement 20
 116 Table 13 – Operations: CIM_ServiceAffectsElement 20
 117 Table 14 – Operations: CIM_VRFRoute 20
 118 Table 15 – Operations: CIM_MemberOfCollection 21
 119 Table 16 – CIM Elements: Network Management - Routing and Forwarding Profile 26
 120 Table 17 – Class: CIM_HostedService 26
 121 Table 18 – Class: CIM_VRFRoute 27
 122 Table 19 – Class: CIM_RegisteredProfile 27
 123 Table 20 – Class: CIM_VirtualForwardingAndRoutingTable 27
 124 Table 21 – Class: CIM_NextHopRoute 28
 125 Table 21 – Class: CIM_NextHopIPRoute 28
 126

127

Foreword

128 The *Network Management - Virtual Routing and Forwarding Profile* (DSP1065) was prepared by the
129 Network Services Management Working Group of the DMTF.

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

132 Acknowledgments

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

134 Editors:

- 135 • John Parchem – Microsoft
- 136 • John Parchem – DMTF Fellow

137 Contributors:

- 138 • Hemal Shah – Broadcom Corporation
- 139 • John Crandall – Brocade Communications System
- 140 • Alex Zhdankin – Cisco Systems
- 141 • Steve Neely – Cisco Systems
- 142 • Shishir Pardikar – Citrix
- 143 • John Leung – Intel Corporation
- 144 • John Parchem – Microsoft Corporation
- 145 • Lawrence Lamers – VMware
- 146 • Dr. Bhumip Khasnabish - ZTE Corporation

147

148

Introduction

149 The information in this specification should be sufficient for a provider or consumer of this data to identify
150 unambiguously the classes, properties, methods, and values that shall be instantiated and manipulated to
151 represent and manage Network Services and the associated configuration information. The target
152 audience for this specification is implementers who are writing CIM-based providers or consumers of
153 management interfaces that represent the component described in this document.

154 Document conventions

155 Typographical conventions

156 The following typographical conventions are used in this document:

- 157 • Document titles are marked in *italics*.
- 158 • ABNF rules are in `monospaced font`.

159

160 Network Management - Virtual Routing and Forwarding 161 Profile

162 1 Scope

163 The *Network Management - Virtual Routing and Forwarding Profile* is a profile that will specify the CIM
164 schema and use cases associated with the general and common aspects of routing and forwarding table
165 including VRFs found in an Ethernet Switch with routing capabilities. This profile includes a specification
166 of the Layer 3 interface configuration service, Sub-Interface, Tunnel Interface switch virtual interface and
167 loopback interface.

168 2 Normative references

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

173 DMTF DSP0004, *CIM Infrastructure Specification 2.7*,
174 http://www.dmtf.org/standards/published_documents/DSP0004_2.7.pdf

175 DMTF DSP0200, *CIM Operations over HTTP 1.3.1*,
176 http://www.dmtf.org/standards/published_documents/DSP0200_1.3.1.pdf

177 DMTF DSP0223, *Generic Operations 1.0*,
178 http://www.dmtf.org/standards/published_documents/DSP0223_1.0.2.pdf

179 DMTF DSP1001, *Management Profile Specification Usage Guide 1.1*,
180 http://www.dmtf.org/standards/published_documents/DSPF1001_1.1.pdf

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

183 DMTF DSP1097, *Virtual Ethernet Switch Profile 1.1*,
184 http://dmtf.org/sites/default/files/standards/documents/DSP1097_1.1.0.pdf

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

187 3 Terms and definitions

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

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

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

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

201 The terms defined in [DSP0004](#), [DSP0223](#), and [DSP1001](#) apply to this document. The following additional
202 terms are used in this document (update the entire doc using the following revised terminology)

203 **3.1**

204 **If**

205 indicates requirements to be followed strictly to conform to the document when the specified conditions
206 are met

207 **3.2**

208 **Shall**

209 Shall indicates requirements to be followed strictly to conform to the document and from which no
210 deviation is permitted

211 Add ‘Should’ as well (Note: we need ot keep “shall”, “should” and “may”)

212 **3.3**

213 **May**

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

215 **3.4**

216 **pending configuration (update per ISO Directive)**

217 indicates the configuration that will be applied to an IP network connection the next time the IP network
218 connection accepts a configuration

219 **3.5**

220 **referencing profile (update per ISO Directive)**

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 **3.6**

224 **unspecified (update per ISO Directive)**

225 indicates that this profile does not define any constraints for the referenced CIM element or operation
226

227 **4 Symbols and abbreviated terms**

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

230 **4.1**

231 **IP**

232 Internet Protocol

233 **4.2**

234 **VLAN**

235 Virtual Local Area Network

236 **4.3**

237 **VRF**

238 Virtual Routing and Forwarding table
239

240 **4.4**
 241 **BGP**
 242 Border Gateway Protocol

243 **5 Synopsis**

244 **Profile name:** Network Management - Virtual Routing and Forwarding Profile

245 **Version:** 1.0.0

246 **Organization:** DMTF

247 **CIM Schema version:** 2.41e

248 **Central class:** CIM_VRFConfigurationService

249 **Scoping class:** CIM_System

250 The *Network Management - Virtual Routing and Forwarding Profile* is a base (abstract) profile that will
 251 specify the CIM schema and use cases associated with the general and common aspects of Network
 252 Policy Management. This profile includes a specification of the Network Policy Service, Network Policy,
 253 Network Policy Rule and Setting Data, Policy Conditions and Action and describes how the network
 254 Policies can be applied to the Managed Elements.

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

256 **Table 1 – Referenced profiles**

Profile Name	Organization	Version	Requirement	Description
Profile Registration	DMTF	1.0	Mandatory	None
Virtual Ethernet Switch	DMTF	1.1	Mandatory	None

257 **6 Description**

258 The *Network Management - Virtual Routing and Forwarding Profile* is a base (abstract) profile that will
 259 specify the CIM schema and use cases associated with the general and common aspects of creating and
 260 configuring The routes and associated interfaces of a Virtual Routing and Forwarding table typically found
 261 in an Ethernet switch with routing capabilities. This includes the VRF configuration service, to instantiate
 262 and configure VRFs and their associated routes.

263

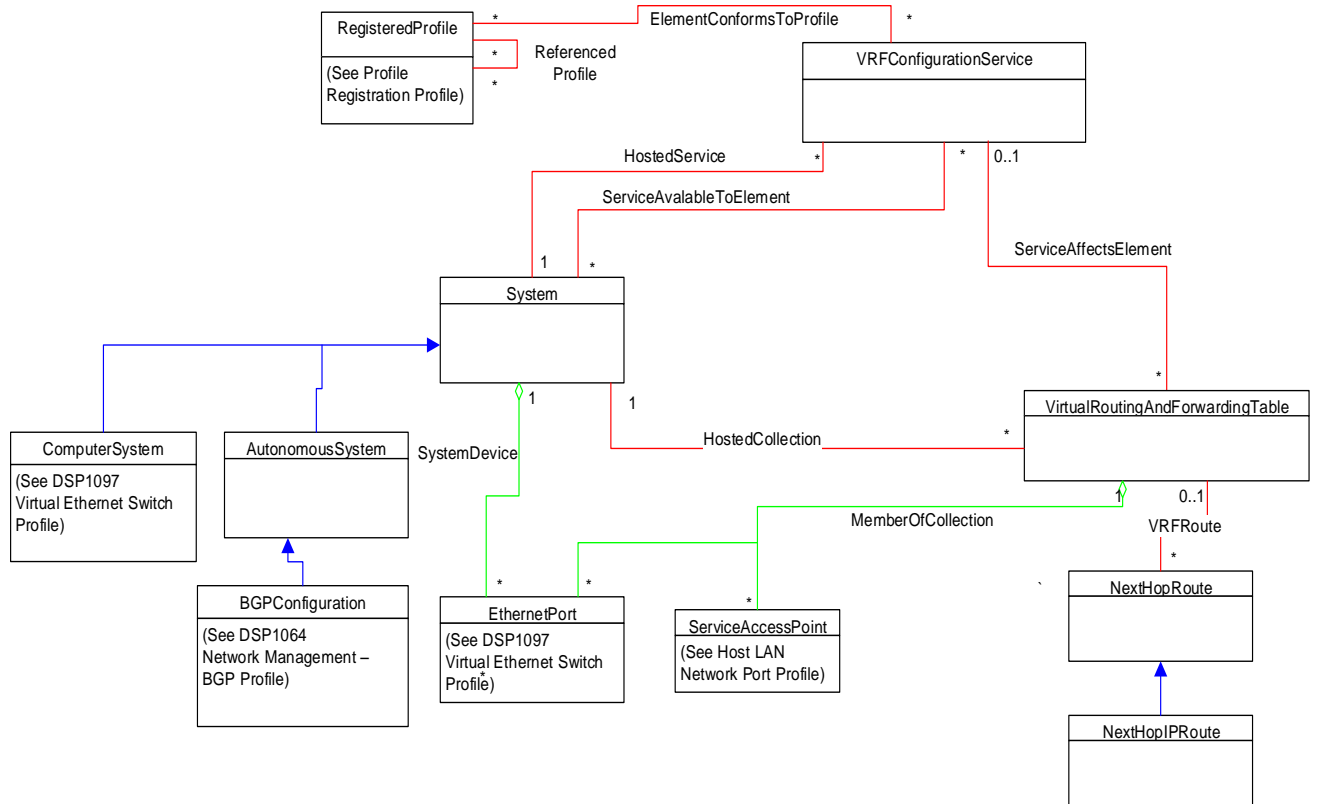
264

265

266

267 **6.1 Class diagram**

268 Figure 1 represents the class schema for the *Network Management - Virtual Routing and Forwarding*
 269 *Profile*. For simplicity, the CIM_ prefix has been removed from the names of the classes.



270
271

272 **Figure 1 – Network Management - Virtual Routing and Forwarding Profile: Class diagram**

273 Figure 1 is a class diagram for the configuration of the virtual routing and forwarding tables (VRF) typically
 274 found in an Ethernet switch that supports routing. The figure shows the CIM_VRFConfigurationService
 275 and the CIM_VirtualRoutingAndForwardingTable classes and the associated ports and interfaces that are
 276 within the VRF. A set of CIM_NextHopIPRoute instances describe the associated routes in the VRF. A
 277 VRF is hosted by a System typically either an instance of CIM_ComputerSystem, representing an
 278 Ethernet Switch or an Instance of a router possibly within a switch represented by
 279 CIM_AutonomousSystem.

280 **6.2 CIM_VirtualRoutingAndForwardingTable**

281 A VRF allows multiple routing tables with independent, the same or overlapping IP addresses within the
 282 same Ethernet switch. An instance of CIM_VirtualRoutingAndForwardingTable represents a single VRF. It
 283 is a subclass of CIM_SystemSpecificCollection where the members are interfaces, represented as
 284 CIM_ServiceAccessPoint instances or ports represented by CIM_EthernetPort. Also associated with the
 285 VRF are the routes configured for the VRF.

286 **6.3 CIM_VRFConfigurationService**

287 The VRF configuration service manages the creation and deletion of VRF, the addition and removal of
 288 ports and interfaces to the VRF and the configuration of the next hop routes associated with the VRF.

289 6.4 CIM_System

290 Subclasses of CIM_System represents either an Ethernet switch, CIM_ComputerSystem, or a router,
291 CIM_AutonomousSystem and CIM_BGPConfiguration. VRFs can be created and associated with any of
292 these example subclasses of CIM_System.

293 6.5 CIM_ServiceAccessPoint

294 This is the base class for interfaces and service access points in an Ethernet switch. An interface within
295 an Ethernet switch or router can be a member of no more than one
296 CIM_VirtualRoutingAndForwardingTable collection. An example would be an instance of
297 CIM_RemoteServiceAccessPoint representing an NTP or RADIUS service. Another example would be a
298 layer 3 interface such as CIM_IPSubinterface which also has CIM_ServiceAccessPoint as a super class.

299 6.6 CIM_EthernetPort

300 Represents the switch ports in an Ethernet port. Once a VRF is configured, a port can be assigned to the
301 VRF which would make the CIM_EthernetPort instance a member of the
302 CIM_VirtualRoutingAndForwardingTable collection.

303 6.7 CIM_NextHopIPRoute

304 Each VRF can have an associated set of next hop routes. Static routes can be configured using the
305 associated CIM_VRFConfigurationService. Each route results is an instance of CIM_NextHopIPRoute
306 that is associated to VRF through a CIM_VRFRoute instance.

307 7 Implementation

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

310 7.1 Representing the Virtual routing and forwarding table management 311 capabilities

312

313 7.1.1 CIM_System

314 An instance of CIM_System shall be the scoping class for this profile. This scoping instance should be an
315 instance of either CIM_ComputerSystem representing an Ethernet switch or an instance of
316 CIM_AutonomousSystem representing a router.

317 The scoping class instance of CIM_System shall be associated to central class instance of
318 CIM_VRFConfigurationService through an instance of CIM_HostedService.

319 If a CIM_System instance can be the TargetRouter of a CIM_VRFConfigurationService.CreateVRF()
320 method call, the CIM_System instance shall be associated to that instance of the
321 CIM_VRFConfigurationService through an instance of CIM_ServiceAvailableToElement.

322 7.1.2 CIM_VRFConfigurationService

323 One or more instances of CIM_VRFConfigurationService shall be instantiated.

324 Each instance of the CIM_VRFConfigurationService shall be associated with the instance of the scoping
325 CIM_System through instance of CIM_HostedService.

326 Instances of CIM_System that may be used as the HostSystem parameter of a
327 CIM_VRFConfigurationService.CreateVRF() method shall be associated to the
328 CIM_VRFConfigurationService instance through an instance of CIM_ServiceAvailableToElement.

329 The instances of the CIM_VRFConfigurationService class shall be associated to each
330 CIM_VirtualRoutingAndForwardingTable instance that may be used as the VRF parameter of its
331 AddRoute(), AddPortMember() or AddSAPMember() method through an instance of
332 CIM_ServiceAffectsElement.

333

334 7.1.3 CIM_VirtualRoutingAndForwardingTable

335 Each CIM_VirtualRoutingAndForwardingTable instance shall be associated to an instance of
336 CIM_System through an instance of CIM_HostedCollection.

337 If the CIM_VirtualRoutingAndForwardingTable instance was created with a
338 CIM_VRFConfigurationService.CreateVRF() method, the CIM_System instance referenced in the
339 TargetRoute method parameter shall be associated through an instance of CIM_HostedCollection.

340 If the CIM_VirtualRoutingAndForwardingTable instance was created with a
341 CIM_VRFConfigurationService.CreateVRF() method with a null TargetRouter method parameter, the
342 CIM_VirtualRoutingAndForwardingTable instance shall be associated to the scoping instance of
343 CIM_System through an instance of CIM_HostedCollection.

344 7.1.4 CIM_NextHopIPRoute

345 Each instance of CIM_NextHopIPRoute that describes a route for an instance of a
346 CIM_VirtualRoutingAndForwardingTable (see 7.1.3) shall be associated to that instance through an
347 instance of CIM_VRFRoute.

348 7.1.5 CIM_EthernetPort

349 Each instance of CIM_EthernetPort that describes a switch port contained in a VRF,
350 CIM_VirtualRoutingAndForwardingTable (see 7.1.3), shall be associated to the VRF instance through an
351 instance of CIM_MemberOfCollection. An instance of CIM_EthernetPort shall be associated to no more
352 than one instance of CIM_VirtualRoutingAndForwardingTable.

353 7.1.6 CIM_ServiceAccessPoint

354 Each instance of CIM_ServiceAccessPort that describes an interface or a remote service access point
355 contained in a VRF, CIM_VirtualRoutingAndForwardingTable (see 7.1.3), shall be associated to the VRF
356 instance through an instance of CIM_MemberOfCollection. An instance of CIM_ServiceAccessPoint shall
357 be associated to no more than one instance of CIM_VirtualRoutingAndForwardingTable.

358 8 Methods

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

361

362 8.1 Extrinsic Methods

363 If synchronous execution of a method succeeds, the implementation shall set a return value of
364 0 (Completed with No Error).

365 If synchronous execution of a method fails, the implementation shall set a return value of 2 (Failed) or a
366 more specific return code as specified with the respective method.

367 If a method is executed as an asynchronous task, the implementation shall perform all of the following ac-
368 tions:

- 369 • Set a return value of 4096 (Job Started).
- 370 • Set the value of the Job output parameter to refer to an instance of the CIM_ConcreteJob class
371 that represents the asynchronous task.
- 372 • Set the values of the JobState and TimeOfLastStateChange properties in that instance to repre-
373 sent the state and last state change time of the asynchronous task.

374 In addition, the implementation may present state change indications as task state changes occur.

375 If the method execution as an asynchronous task succeeds, the implementation shall perform all of the
376 following actions:

- 377 • Set the value of the JobState property to 7 (Completed).
- 378 • Provide an instance of the CIM_AffectedJobEntity association with property values set as fol-
379 lows:
 - 380 – The value of the AffectedElement property shall refer to the object that represents the top-
381 level entity that was created or modified by the asynchronous task. For example, for the
382 CIM_IPConfigurationService.AddIPProtocolEndpoint() method, this is an instance of the
383 CIM_IPProtocolEndpoint class
 - 384 – The value of the AffectingElement property shall refer to the instance of the
385 CIM_ConcreteJob class that represents the completed asynchronous task.
 - 386 – The value of the first element in the ElementEffects[] array property (ElementEffects[0])
387 shall be set to 5 (Create) for the CIM_IPConfigurationService.AddIPProtocolEndpoint()
388 method. Otherwise, this value shall be 0 (Unknown).

389 If the method execution as an asynchronous task fails, the implementation shall set the value of the
390 JobState property to 9 (Killed) or 10 (Exception).

391 **8.1.1.1 Job parameter**

392 The implementation shall set the value of the Job parameter as a result of an asynchronous execution of
393 a method of the CIM_IPConfigurationService as follows:

- 394 • If the method execution is performed synchronously, the implementation shall set the value to
395 NULL.
- 396 • If the method execution is performed asynchronously, the implementation shall set the value to
397 refer to the instance of the CIM_ConcreteJob class that represents the asynchronous task.

398

399 **8.1.2 CIM_VRFConfigurationService.CreateVRF()**

400 The implementation of the CreateVRF() method is optional, the provisions in this sub clause apply in
401 addition to behavior applicable to all extrinsic methods as specified in 8.1.

402 The successful execution of the CreateVRF() method shall create an instance of
403 CIM_VirtualRoutingAndForwardingTable as described in the sub clause 7.1.3.

404 Table 2 contains requirements for parameters of this method.

405

Table 2 – CreateVRF() Method: Parameters

<i>Qualifiers</i>	<i>Name</i>	<i>Type</i>	<i>Description/Values</i>
IN	TargetRouter	CIM_System REF	See 8.1.2.1.1
IN	VRF	String	See 8.1.2.1.2
OUT	ResultingVRF	REF	See 8.1.2.1.3
OUT	Job	CIM_ConcreteJob REF	See 8.1.2.1.4

406 **8.1.2.1.1 TargetRouter**

407 An optional reference to a CIM_System instance. The referenced instance shall comply with the
408 subclause 7.1.1.

409 **8.1.2.1.2 VRF**

410 A required string containing one embedded instances of the class-subclass of
411 CIM_VirtualRoutingAndForwardingTable that describes the configuration of the resultant
412 CIM_VirtualRoutingAndForwardingTable instance. The populated properties of the embedded instance
413 should not contain key properties, and any key property values may be ignored.

414

415 **8.1.2.1.3 ResultingVRF**

416 If the creation of the VRF was successful, a reference to the resultant instance of class
417 CIM_VirtualRoutingAndForwardingTable that represents the newly defined VRF shall be returned. The
418 created CIM_VirtualRoutingAndForwardingTable instance shall comply with sub clause 7.1.3.

419 **8.1.2.1.4 Job**

420 See 8.1.1.1

421 **8.1.3 CIM_VRFConfigurationService.RemoveVRF()**

422 The implementation of the RemoveVRF() method is optional, the provisions in this sub clause apply in
423 addition behavior applicable to all extrinsic methods as specified in 8.1.

424 The successful execution of the RemoveVRF() method shall remove the instance referenced in the
425 methods VRF parameter and should also remove any associated CIM_NextHopRoute instances.

426 Table 3 contains requirements for parameters of this method.

427

Table 3 – RemoveVRF() Method: Parameters

<i>Qualifiers</i>	<i>Name</i>	<i>Type</i>	<i>Description/Values</i>
IN	VRF	CIM_VirtualRoutingAndForwardingTable REF	See 8.1.3.1.1
OUT	Job	CIM_ConcreteJob REF	See 8.1.3.1.2

428 **8.1.3.1.1 VRF**

429 A reference to the instance of the class CIM_VirtualRoutingAndForwardingTable that shall be removed.

430 **8.1.3.1.2 Job**

431 See 8.1.1.1

432

433 **8.1.4 CIM_VRFConfigurationService.AddRoute()**

434 The implementation of the AddRoute() method is required, the provisions in this sub clause shall apply in
435 addition to the behavior applicable to all extrinsic methods as specified in 8.1.

436 The successful execution of the AddRoute() method shall create an instance of CIM_NextHopIPRoute as
437 described in the sub clause 7.1.4. This instance shall be associated with the referenced VRF through an
438 instance of CIM_VRFRoute.

439 Table 4 contains requirements for parameters of this method.

440 **Table 4 – AddRoute() Method: Parameters**

<i>Qualifiers</i>	<i>Name</i>	<i>Type</i>	<i>Description/Values</i>
IN	VRF	CIM_VirtualRoutingAndForwardingTable REF	See 8.1.4.1.1
IN	Route	String	See 8.1.4.1.2
OUT	ResultingRoute	CIM_NextHopIPRoute REF	See 8.1.4.1.3
OUT	Job	CIM_ConcreteJob REF	See 8.1.4.1.4

441 **8.1.4.1.1 VRF**

442 A required reference to a CIM_VirtualRoutingAndForwardingTable instance.

443 **8.1.4.1.2 Route**

444 A required string containing one embedded instance of the class or subclass of CIM_NextHopIPRoute
445 that describes the configuration of the resultant CIM_NextHopIPRoute instance. The populated properties
446 of the embedded instance should not contain key properties, and any key property values may be
447 ignored.

448

449 **8.1.4.1.3 ResultingRoute**

450 If the creation of the next hop route was successful, a reference to the resultant instance of class
451 CIM_NextHopIPRoute that represents the newly defined route for the VRF shall be returned. The created
452 CIM_NextHopIPRoute instance shall comply with sub clause 7.1.4.

453 **8.1.4.1.4 Job**

454 See 8.1.1.1

455 **8.1.5 CIM_VRFConfigurationService.RemoveRoute()**

456 The implementation of the RemoveRoute() method is required, the provisions in this sub clause shall
457 apply in addition to the behavior applicable to all extrinsic methods as specified in 8.1.

458 The successful execution of the RemoveRoute() method shall remove the instance referenced in the
459 method's Route parameter from the VRF referenced in the VRF parameter

460 Table 5 contains requirements for parameters of this method.

461

Table 5 – RemoveRoute() Method: Parameters

<i>Qualifiers</i>	<i>Name</i>	<i>Type</i>	<i>Description/Values</i>
IN	VRF	CIM_VirtualRoutingAndForwardingTable REF	See 8.1.5.1.1
IN	Route	CIM_NextHopIPRoute REF	See 8.1.5.1.2
OUT	Job	CIM_ConcreteJob REF	See 8.1.5.1.3

462

463 8.1.5.1.1 VRF

464 A required reference to a CIM_VirtualRoutingAndForwardingTable instance from which the route shall be
465 removed.

466 8.1.5.1.2 Route

467 A references to instance of the class CIM_NextHopIPRoute that shall be removed.

468 8.1.5.1.3 Job

469 See 8.1.1.1

470 8.1.6 CIM_VRFConfigurationService.AddPortMember()

471 The implementation of the AddPortMember() method is required, the provisions in this sub clause shall
472 apply in addition to the behavior applicable to all extrinsic methods as specified in 8.1.

473 The successful execution of the AddPortMember() shall associate the referenced port in the PortMember
474 method parameter to the referenced VRF in the VRF method parameter through an instance of
475 CIM_MemberOfCollection.

476 Table 6 contains requirements for parameters of this method.

477

Table 6 – AddPortMember() Method: Parameters

<i>Qualifiers</i>	<i>Name</i>	<i>Type</i>	<i>Description/Values</i>
IN	VRF	CIM_VirtualRoutingAndForwardingTable REF	See 8.1.6.1.1
IN	PortMember	CIM_EthernetPort REF	See 8.1.4.1.2
OUT	Job	CIM_ConcreteJob REF	See 8.1.6.1.3

478 8.1.6.1.1 VRF

479 A required reference to a CIM_VirtualRoutingAndForwardingTable instance to which the port shall be
480 added.

481 8.1.6.1.2 PortMember

482 A reference to the CIM_EthernetPort instance that is being added to the VRF.

483 8.1.6.1.3 Job

484 See 8.1.1.1

485 **8.1.7 CIM_VRFConfigurationService.RemovePortMember()**

486 The implementation of the RemovePortMember() method is required, the provisions in this sub clause
 487 shall apply in addition to the behavior applicable to all extrinsic methods as specified in 8.1.

488 The successful execution of the RemovePortMember() method shall remove the referenced
 489 CIM_EthernetPort passed in the PortMember parameter from the VRF passed in the VRF method
 490 parameter, by removing the CIM_MemberOfCollection instance forming the association.

491 Table 2 contains requirements for parameters of this method.

492 **Table 7 – RemovePortMember() Method: Parameters**

Qualifiers	Name	Type	Description/Values
IN	VRF	CIM_VirtualRoutingAndForwardingTable	See 8.1.7.1.1
IN	PortMember	CIM_EthernetPort REF	See 8.1.7.1.2
OUT	Job	CIM_ConcreteJob REF	See 8.1.7.1.3

493 **8.1.7.1.1 VRF**

494 A required reference to a CIM_VirtualRoutingAndForwardingTable instance from which the port shall be
 495 removed.

496 **8.1.7.1.2 PortMember**

497 A required reference to instance of the class CIM_EthreNetPort that shall be removed from the referenced
 498 VRF.

499 **8.1.7.1.3 Job**

500 See 8.1.1.1

501

502 **8.1.8 CIM_VRFConfigurationService.AddSAPMember()**

503 The implementation of the AddSAPMember() method is required, the provisions in this sub clause shall
 504 apply in addition to the behavior applicable to all extrinsic methods as specified in 8.1.

505 The successful execution of the AddSAPMember() shall associate the referenced instance of
 506 CIM_ServiceAccessPoint in the SAPMember method parameter to the referenced VRF in the VRF
 507 method parameter through an instance of CIM_MemberOfCollection.

508 Table 8 contains requirements for parameters of this method.

509 **Table 8 – AddSAPMember() Method: Parameters**

Qualifiers	Name	Type	Description/Values
IN	VRF	CIM_VirtualRoutingAndForwardingTable REF	See 8.1.8.1.1
IN	SAPMember	CIM_ServiceAccessPoint REF	See 8.1.8.1.2
OUT	Job	CIM_ConcreteJob REF	See 8.1.8.1.3

510 **8.1.8.1.1 VRF**

511 A required reference to a CIM_VirtualRoutingAndForwardingTable instance to which the instance of
512 CIM_ServiceAccessPoint shall be added.

513 **8.1.8.1.2 SAPMember**

514 A required reference to the instance of CIM_ServiceAccessPoint that shall be added to the referenced
515 VRF.

516 **8.1.8.1.3 Job**

517 See 8.1.1.1

518 **8.1.9 CIM_VRFConfigurationService.RemoveSAPMember()**

519 The implementation of the RemoveSAPMember() method is required, the provisions in this sub clause
520 shall apply in addition to the behavior applicable to all extrinsic methods as specified in 8.1.

521 The successful execution of the RemoveSAPMember() method shall remove the referenced
522 CIM_EthernetPort passed in the SAPMember parameter from the VRF passed in the VRF method
523 parameter, by removing the CIM_MemberOfCollection instance forming the association.

524 Table 9 contains requirements for parameters of this method.

525 **Table 9 – RemoveSAPMember() Method: Parameters**

<i>Qualifiers</i>	<i>Name</i>	<i>Type</i>	<i>Description/Values</i>
IN	VRF	CIM_VirtualRoutingAndForwardingTable	See 8.1.9.1.1
IN	SAPMember	CIM_ServiceAccessPoint REF	See 8.1.9.1.2
OUT	Job	CIM_ConcreteJob REF	See 8.1.9.1.3

526 **8.1.9.1.1 VRF**

527 A required reference to a CIM_VirtualRoutingAndForwardingTable instance from which the
528 CIM_ServiceAccessPoint shall be removed.

529 **8.1.9.1.2 SAPMember**

530 A required reference to instance of the class CIM_ServiceAccessPoint that shall be removed from the
531 referenced VRF.

532 **8.1.9.1.3 Job**

533 See 8.1.1.1

534

535

536 **8.2 Profile conventions for operations**

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

539 The default list of operations is as follows:

- 540 • GetInstance
- 541 • EnumerateInstances
- 542 • EnumerateInstanceNames
- 543 • Associators
- 544 • AssociatorNames
- 545 • References
- 546 • ReferenceNames

547

548 8.3 CIM_HostedService

549 Table 10 lists implementation requirements for operations. If implemented, these operations shall be
 550 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 10, all operations
 551 in the default list in 0 shall be implemented as defined in [DSP0200](#).

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

553 **Table 10 – Operations: CIM_HostedService**

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

554 8.4 CIM_HostedCollection

555 Table 11 lists implementation requirements for operations. If implemented, these operations shall be
 556 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 10, all operations
 557 in the default list in 0 shall be implemented as defined in [DSP0200](#).

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

559

560 **Table 11 – Operations: CIMHostedCollection**

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

561 **8.5 CIM_ServiceAvailableToElement**

562 Table 12 lists implementation requirements for operations. If implemented, these operations shall be
 563 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 10, all operations
 564 in the default list in 0 shall be implemented as defined in [DSP0200](#).

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

566 **Table 12 – Operations: CIM_ServiceAvailableToElement**

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

567 **8.6 CIM_ServiceAffectsElement**

568 Table 13 lists implementation requirements for operations. If implemented, these operations shall be
 569 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 10, all operations
 570 in the default list in 0 shall be implemented as defined in [DSP0200](#).

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

572 **Table 13 – Operations: CIM_ServiceAffectsElement**

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

573 **8.7 CIM_VRFRoute**

574 Table 14 lists implementation requirements for operations. If implemented, these operations shall be
 575 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 10, all operations
 576 in the default list in 0 shall be implemented as defined in [DSP0200](#).

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

578 **Table 14 – Operations: CIM_VRFRoute**

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

579 **8.8 CIM_MemberOfCollection**

580 Table 15 lists implementation requirements for operations. If implemented, these operations shall be
 581 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 10, all operations
 582 in the default list in 0 shall be implemented as defined in [DSP0200](#).

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

584 **Table 15 – Operations: CIM_MemberOfCollection**

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

585

586 **8.9 CIM_VRFConfigurationService**

587 All operations in the default list in 0 shall be implemented as defined in [DSP0200](#).

588 **8.10 CIM_NextHopIPRoute**

589 All operations in the default list in 0 shall be implemented as defined in [DSP0200](#).

590 **8.11 CIM_VirtualRoutingAndForwardingTable**

591 All operations in the default list in 0 shall be implemented as defined in [DSP0200](#).

592 **8.12 CIM_System**

593 All operations in the default list in 0 shall be implemented as defined in [DSP0200](#).

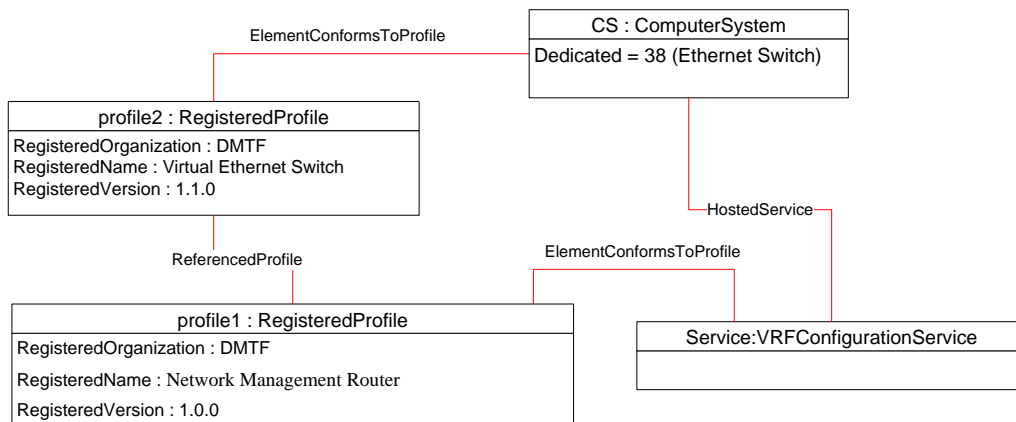
594

595 **9 Use cases**

596 This clause contains object diagrams and use cases for the *Network Management - Virtual Routing and*
 597 *Forwarding Profile*.

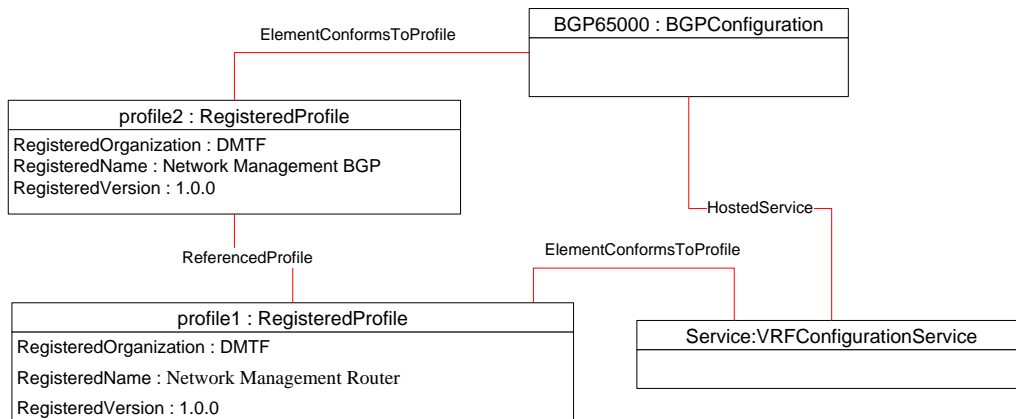
598 **9.1 Profile Registration**

599 The object diagram in Figure 2 and Figure 3 show two possible examples for advertising profile
 600 conformance. Figure 2 is an example where an instance of CIM_ComputerSystem that is a compliant
 601 Virtual Ethernet Switch is the scoping class. The second example shows a CIM_BGPCConfiguration
 602 representing a router that is the scoping CIM_System instance.



603

604 **Figure 2 – Registered profile with Computer System Profile**



605

606 **Figure 3 – Registered profile BGPCConfiguration Profile**

607

608 9.2 VRF with Routes and Interfaces

609 The object diagram in Figure 4 contains the basic element used to model a VRF. The instance diagram
610 shows a CIM_ComputerSystem instance CS0 hosting an instance of CIM_VRFConfigurationService,
611 CS0VRF. In this instance diagram one instance of a VRF, ManagementVRF has been created and is
612 under the CIM_VRFConfigurationService instance CS0VRF. The diagram also shows that the
613 ManagementVRF has two members, one an Ethernet port (E0/1) the other a remote service access point
614 (NPT0).

615 The following method call through the CS0VRF instance of CIM_VRFConfigurationService were
616 performed to create and configure the VRF.

617 The ManagementVRF was created with a CIM_VRFConfigurationService.CreateVRF() method with the
618 following parameters. Note, this is for illustration purposes, other properties can be populated in the
619 embedded class instances as required.

- 620 • TargetRouter - Reference to CS0: CIM_ComputerSystem
- 621 • VRF
 - 622 ○ Embedded instance of CIM_VirtualRoutingAndForwardingTable {
 - 623 ElementName = ManagementVRF }

624 The method call would return:

- 625 • ResultingVRF – Reference to ManagementVRF: CIM_VirtualRoutingAndForwardingTable.

626 The Ethernet port E0/1 was added to the VRF with the CIM_VRFConfigurationService.AddMemberPort()
627 method with the following parameters.

- 628 • VRF - Reference to CIM_VirtualRoutingAndForwardingTable: ManagementVRF
- 629 • MemberPort – Reference to CIM_EthernetPort: E0/1

630 The remote service access point NPT0 was added to the VRF with the
631 CIM_VRFConfigurationService.AddMemberSAP() method with the following parameters.

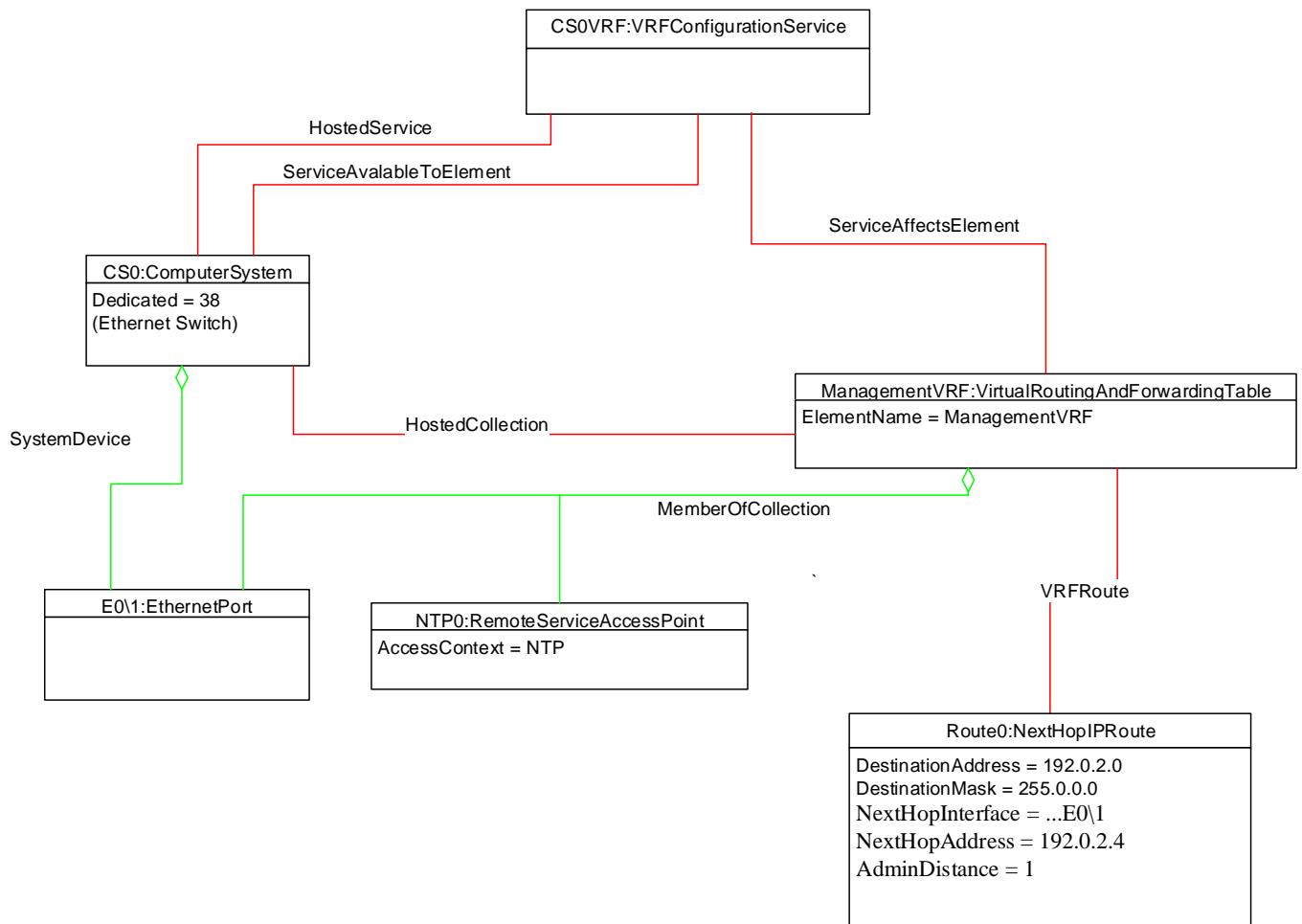
- 632 • VRF - Reference to CIM_VirtualRoutingAndForwardingTable: ManagementVRF
- 633 • MemberSAP – Reference to CIM_RemoteServiceAccessPoint: NPT0

634 A single route was added using the CIM_VirtualRoutingAndForwardingTable.AddRoute method with the
635 following parameters.

- 636 • VRF - Reference to CIM_VirtualRoutingAndForwardingTable: ManagementVRF
- 637 • Route -
 - 638 ○ Embedded instance of CIM_NextHopIPRoute {
 - 639 DestinationAddress = 192.0.2.0
 - 640 DestinationMask = 255.0.0.0
 - 641 NextHopInterface = ...E0\1
 - 642 NextHopAddress = 192.0.2.4
 - 643 AdminDistance = 1 }

644 The method call would return:

- 645 • ResultingRoute – Reference Route0: CIM_NextHopIPRoute.



646
647

648

Figure 4 – VRF Instance Diagram

649 **9.3 VRF in a BGP router**

650 Figure 5 shows that a CIM_VRFConfigurationService instance CS0VRF hosted by the
651 CIM_ComputerSystem instance CS0 representing an Ethernet switch can create a VRF hosted by a
652 router within the Switch. In this example the instance BGP64401BGPCOnfiguration.

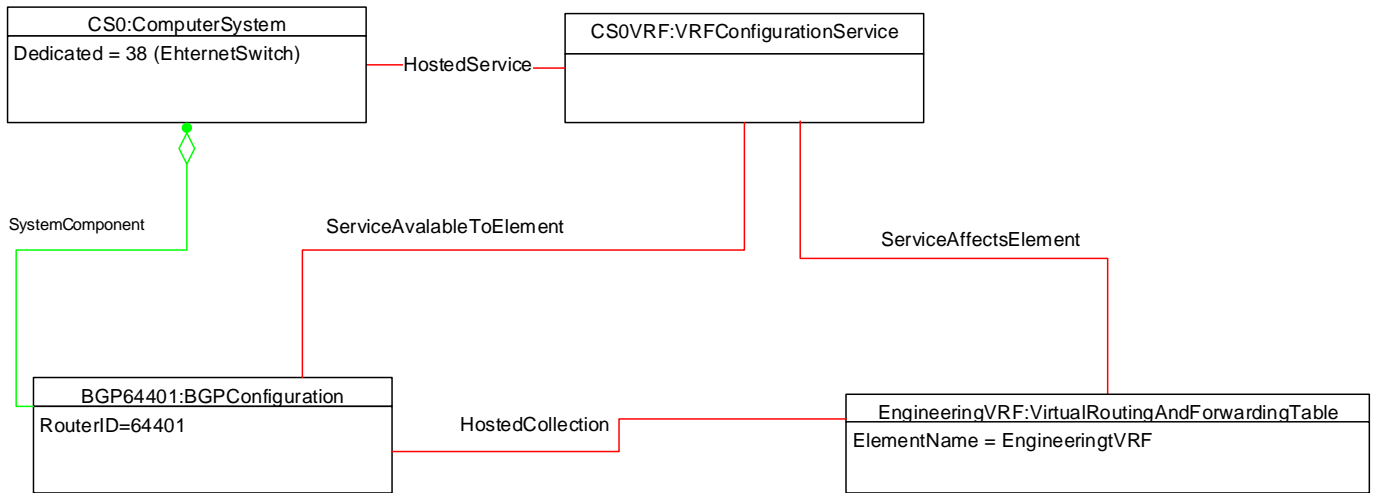
653 The ManagementVRF was created with a CIM_VRFConfigurationService.CreateVRF() method with the
654 following parameters. Note, this is for illustration purposes, and other properties can be populated in
655 the embedded class instances as required. TargetRouter - Reference to
656 BGP64401:CIM_BGPCOnfiguration

- 657 • VRF
 - 658 ○ Embedded instance of CIM_VirtualRoutingAndForwardingTable {
 - 659 ElementName EngineeringVRF }

660 The method call would return:

- 661 • ResultingVRF – Reference to EngineeringVRF:CIM_VirtualRoutingAndForwardingTable.

662



663
664

Figure 5 – BGP with VRF

665
666
667
668

669 **10 CIM Elements**

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

673 **Table 16 – CIM Elements: Network Management - Virtual Routing and Forwarding Profile**

Element Name	Requirement	Description
Classes		
CIM_VRFRoute	Optional	See clause 7.1.3
CIM_AutonomousSystem	Optional	See clause 7.1.1
CIM_EthernetPort	Optional	See clause 7.1.5
CIM_HostedCollection	Mandatory	See clause 7.1.3
CIM_HostedService	Mandatory	See clause 7.1.2
CIM_MemberOfCollection	Optional	See clause 7.1.5 and 7.1.6
CIM_NextHopIPRoute	Optional	See clause 7.1.4
CIM_ServiceAccessPoint	Optional	See clause 7.1.6
CIM_ServiceAffectsElement	Mandatory	See clause 7.1.2
CIM_ServiceAvailableToElement	Mandatory	See clause 7.1.2
CIM_System	Mandatory	See clause 7.1.1
CIM_VirtualRoutingAndForwardingTable	Mandatory	See clause 7.1.3
CIM_VRFConfigurationService	Mandatory	See clause 7.1.2
Indications		
None defined in this profile		

674

675 **10.1 CIM_HostedService**

676 CIM_HostedService relates the CIM_VRFConfigurationService instance to its scoping
 677 CIM_ComputerSystem instance. Table 17 provides information about the properties of
 678 CIM_HostedService.

679 **Table 17 – Class: CIM_HostedService**

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

680 **10.2 CIM_VRFRoute**

681 CIM_VRFRoute is used to associate a set of CIM_NextHopRoute instances with a
 682 CIM_VirtualRoutingAndForwardingTable instance. Table 18 provides information about the properties of
 683 CIM_HostedService.

684 **Table 18 – Class: CIM_VRFRoute**

Elements	Requirement	Description
Antecedent	Mandatory	Key: This shall be a reference to a CIM_VirtualRoutingAndForwardingTable instance. Cardinality 1
Dependent	Mandatory	Key: This shall be a reference to an instance CIM_NextHopRoute. Cardinality *

685 **10.3 CIM_RegisteredProfile**

686 CIM_RegisteredProfile identifies the *Network Management - Virtual Routing and Forwarding Profile* in
 687 order for a client to determine whether an instance of CIM_IPProtocolEndpoint is conformant with this
 688 profile. The CIM_RegisteredProfile class is defined by the [Profile Registration Profile](#). With the exception
 689 of the mandatory values specified for the properties in Table 19, the behavior of the
 690 CIM_RegisteredProfile instance is in accordance with the [Profile Registration Profile](#).

691 **Table 19 – Class: CIM_RegisteredProfile**

Elements	Requirement	Description
RegisteredName	Mandatory	This property shall have a value of "Network Management Routing and Forwarding Profile".
RegisteredVersion	Mandatory	This property shall have a value of "1.0.0.a".
RegisteredOrganization	Mandatory	This property shall have a value of "DMTF".

692

693 **10.4 CIM_VirtualForwardingAndRoutingTable**

694 CIM_VirtualForwardingAndRoutingTable is a collection of interfaces and routes that form the context
 695 used for a virtual routing and forwarding table (VRF). Table 20 provides information about the properties
 696 of CIM_VirtualForwardingAndForwardingTable.

697 **Table 20 – Class: CIM_VirtualForwardingAndRoutingTable**

Elements	Requirement	Description
InstanceID	Mandatory	Key: This property shall contain a unique ID to uniquely identified the specific instance.
ElementName	Optional	This property should contain the friendly VRF context name.
RouteDistinguisher	Optional	If populated this shall be an eight Octet field that uniquely distinguishes a route when there are multiple VRFs in a single router.

698 **10.5 CIM_NextHopRoute**

699 CIM_NextHopRoute represents one of a series a “hops” to reach a network destination.

700 **Table 21 – Class: CIM_NextHopRoute**

Elements	Requirement	Description
InstanceID	Mandatory	Key: This property shall contain a unique ID to uniquely identifies the specific instance.
DestinationAddress	Mandatory	The address of the destination that needs to be reached.

701

702 **10.6 CIM_NextHopIPRoute**

703 CIM_NextHopIPRoute contains the properties required to specialize CIM_NextHopRoute for an IP route.

704 **Table 22 – Class: CIM_NextHopIPRoute**

Elements	Requirement	Description
InstanceID	Mandatory	Key: This property shall contain a unique ID to uniquely identified the specific instance.

705

706
707
708
709

ANNEX A (informative)

Change log

Version	Date	Description
1.0.0a	2014-02-14	DMTF Work in Progress

710
711