Document Identifier: DSP1065

Date: 2018-09-04

Version: 1.0.0

Network Management - Virtual Routing and Forwarding Profile

Supersedes: None

Document Class: Normative

Document Status: Published

Document Language: en-US

CONTENTS

Figures

Tables

Foreword

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

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

Acknowledgments

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

Editors:

- John Parchem – DMTF Fellow

Contributors:

- John Crandall – Brocade Communications System
- Dr. Bhumip Khasnabish - ZTE Corporation
- Lawrence Lamers – VMware John Leung – Intel Corporation
- Steve Neely – Cisco Systems
- John Parchem – Microsoft Corporation
- Shishir Pardikar – Citrix
- Hemal Shah – Broadcom Corporation
- Alex Zhdankin – Cisco Systems

Introduction

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

Document conventions

Typographical conventions

The following typographical conventions are used in this document:

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

Network Management - Virtual Routing and Forwarding Profile

1 Scope

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

2 Normative references

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

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

DMTF DSP0200, *CIM Operations over HTTP 1.3.1,*
http://www.dmtf.org/standards/published_documents/DSP0200_1.3.pdf

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

DMTF DSP1001, *Management Profile Specification Usage Guide 1.1*,
http://www.dmtf.org/standards/published_documents/DSP1001_1.1.pdf

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

DMTF DSP1097, *Virtual Ethernet Switch Profile 1.1*,
http://dmtf.org/sites/default/files/standards/documents/DSP1097_1.1.pdf

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

3 Terms and definitions

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

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

The terms "clause", "subclause", "paragraph", and "annex" in this document are to be interpreted as described in ISO/IEC Directives, Part 2, Clause 5.

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

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

3.1
If

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

3.2
Shall

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

3.3
Should

indicates that among several possibilities, one is recommended as particularly suitable, without mentioning or excluding others, or that a certain course of action is preferred but not necessarily required.

3.4
May

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

3.5
pending configuration

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

3.6
referencing profile

indicates a profile that owns the definition of this class and can include a reference to this profile in its “Referenced Profiles” table

3.7
unspecified

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

4 Symbols and abbreviated terms

The abbreviations defined in DSP0004, DSP0223, and DSP1001 apply to this document. The following additional abbreviations are used in this document.

4.1
IP

Internet Protocol

4.2
VLAN

Virtual Local Area Network

4.3
VRF
Virtual Routing and Forwarding table

4.4
BGP
Border Gateway Protocol

5 Synopsis

Profile name: Network Management - Virtual Routing and Forwarding Profile

Version: 1.0.0

Organization: DMTF

CIM Schema version: 2.52

Central class: CIM_VRFConfigurationService

Scoping class: CIM_System

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

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

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

6 Description

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

6.1 Class diagram

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

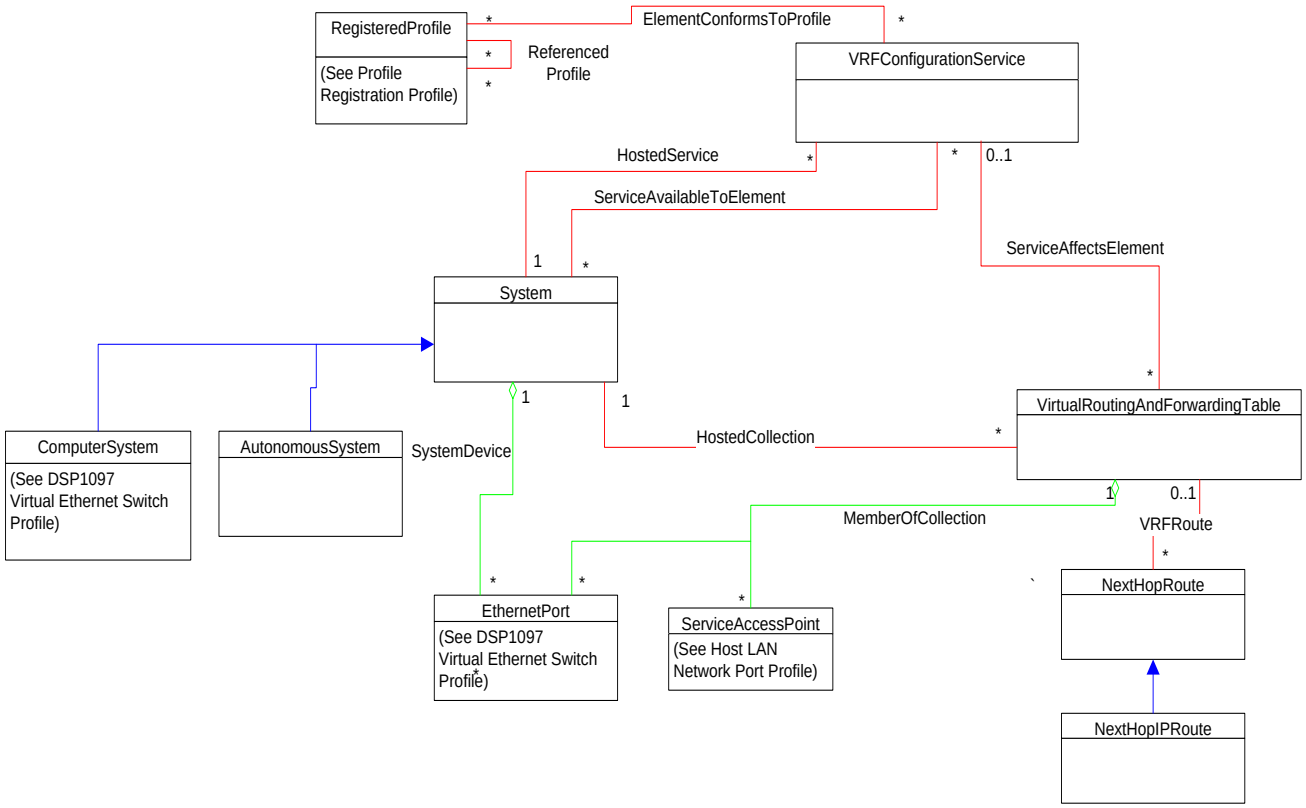

Figure 1 – Network Management - Virtual Routing and Forwarding Profile: Class diagram

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

6.2 CIM_VirtualRoutingAndForwardingTable

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

6.3 CIM_VRFConfigurationService

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

6.4 CIM_System

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

6.5 CIM_ServiceAccessPoint

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

6.6 CIM_EthernetPort

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

6.7 CIM_NextHopIPRoute

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

7 Implementation

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

7.1 Representing the Virtual routing and forwarding table management capabilities

7.1.1 CIM_System

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

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

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

7.1.2 CIM_VRFConfigurationService

One or more instances of CIM_VRFConfigurationService shall be instantiated.

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

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

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

7.1.3 CIM_VirtualRoutingAndForwardingTable

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

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

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

7.1.4 CIM_NextHopIPRoute

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

7.1.5 CIM_EthernetPort

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

7.1.6 CIM_ServiceAccessPoint

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

8 Methods

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

8.1 Extrinsic Methods

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

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

If a method is executed as an asynchronous task, the implementation shall perform all of the following actions:

- Set a return value of 4096 (Job Started).
- Set the value of the Job output parameter to refer to an instance of the CIM_ConcreteJob class that represents the asynchronous task.
- Set the values of the JobState and TimeOfLastStateChange properties in that instance to represent the state and last state change time of the asynchronous task.

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

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

- Set the value of the JobState property to 7 (Completed).
- Provide an instance of the CIM_AffectedJobEntity association with property values set as follows:
  - The value of the AffectedElement property shall refer to the object that represents the top-level entity that was created or modified by the asynchronous task. For example, for the CIM_IPConfigurationService. AddIPProtocolEndpoint() method, this is an instance of the CIM_IPProtocolEndpoint class
  - The value of the AffectingElement property shall refer to the instance of the CIM_ConcreteJob class that represents the completed asynchronous task.
  - The value of the first element in the ElementEffects[ ] array property (ElementEffects[0]) shall be set to 5 (Create) for the CIM_IPConfigurationService. AddIPProtocolEndpoint() method. Otherwise, this value shall be 0 (Unknown).

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

8.1.1 Job parameter

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

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

8.1.2 CIM_VRFConfigurationService.CreateVRF()

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

The successful execution of the CreateVRF( ) method shall create an instance of CIM_VirtualRoutingAndForwardingTable as described in the subclause 7.1.3.

Table 2 contains requirements for parameters of this method.

390 **Table 2 – CreateVRF( ) Method: Parameters**

Qualifiers	Name	Type	Description/Values
IN	TargetRouter	CIM_System REF	See 8.1.2.1
IN	VRF	String	See 8.1.2.2
OUT	ResultingVRF	REF	See 8.1.2.3
OUT	Job	CIM_ConcreteJob REF	See 8.1.2.4

391 #### 8.1.2.1 TargetRouter

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

394 #### 8.1.2.2 VRF

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

399 #### 8.1.2.3 ResultingVRF

400 If the creation of the VRF was successful, a reference to the resultant instance of class
401 CIM_VirtualRoutingAndForwardingTable that represents the newly defined VRF shall be returned. The
402 created CIM_VirtualRoutingAndForwardingTable instance shall comply with subclause 7.1.3.

403 #### 8.1.2.4 Job

404 See 8.1.1

405 ### 8.1.3 CIM_VRFConfigurationService.RemoveVRF()

406 The implementation of the RemoveVRF( ) method is optional, the provisions in this subclause apply in
407 addition behavior applicable to all extrinsic methods as specified in 8.1.

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

410 Table 3 contains requirements for parameters of this method.

411 **Table 3 – RemoveVRF( ) Method: Parameters**

Qualifiers	Name	Type	Description/Values
IN	VRF	CIM_VirtualRoutingAndForwardingTable REF	See 8.1.3.1
OUT	Job	CIM_ConcreteJob REF	See 8.1.3.2

412 #### 8.1.3.1 VRF

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

414 #### 8.1.3.2 Job

415 See 8.1.1

416 ### 8.1.4 CIM_VRFConfigurationService.AddRoute()

417 The implementation of the AddRoute( ) method is required, the provisions in this subclause shall apply in
418 addition to the behavior applicable to all extrinsic methods as specified in 8.1.

419 The successful execution of the AddRoute( ) method shall create an instance of CIM_NextHopIPRoute as
420 described in the subclause 7.1.4. This instance shall be associated with the referenced VRF through an
421 instance of CIM_VRFRoute.

422 Table 4 contains requirements for parameters of this method.

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

Qualifiers	Name	Type	Description/Values
IN	VRF	CIM_VirtualRoutingAndForwardingTable REF	See 8.1.4.1
IN	Route	String	See 8.1.4.2
OUT	ResultingRoute	CIM_NextHopIPRoute REF	See 8.1.4.3
OUT	Job	CIM_ConcreteJob REF	See 8.1.4.4

424 #### 8.1.4.1 VRF

425 A required reference to a CIM_VirtualRoutingAndForwardingTable instance.

426 #### 8.1.4.2 Route

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

431 #### 8.1.4.3 ResultingRoute

432 If the creation of the next hop route was successful, a reference to the resultant instance of class
433 CIM_NextHopIPRoute that represents the newly defined route for the VRF shall be returned. The created
434 CIM_NextHopIPRoute instance shall comply with subclause 7.1.4.

8.1.4.4 Job

See 8.1.1

8.1.5 CIM_VRFConfigurationService.RemoveRoute()

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

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

Table 5 contains requirements for parameters of this method.

Table 5 – RemoveRoute( ) Method: Parameters

Qualifiers	Name	Type	Description/Values
IN	VRF	CIM_VirtualRoutingAndForwardingTable REF	See 8.1.5.1
IN	Route	CIM_NextHopIPRoute REF	See 8.1.5.2
OUT	Job	CIM_ConcreteJob REF	See 8.1.5.3

8.1.5.1 VRF

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

8.1.5.2 Route

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

8.1.5.3 Job

See 8.1.1

8.1.6 CIM_VRFConfigurationService.AddPortMember()

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

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

Table 6 contains requirements for parameters of this method.

Table 6 – AddPortMember( ) Method: Parameters

Qualifiers	Name	Type	Description/Values
IN	VRF	CIM_VirtualRoutingAndForwardingTable REF	See 8.1.6.1
IN	PortMember	CIM_EthernetPort REF	See 8.1.4.2
OUT	Job	CIM_ConcreteJob REF	See 8.1.6.3

8.1.6.1 VRF

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

8.1.6.2 PortMember

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

8.1.6.3 Job

See 8.1.1

8.1.7 CIM_VRFConfigurationService.RemovePortMember()

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

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

Table 7 contains requirements for parameters of this method.

Table 7 – RemovePortMember( ) Method: Parameters

Qualifiers	Name	Type	Description/Values
IN	VRF	CIM_VirtualRoutingAndForwardingTable	See 8.1.7.1
IN	PortMember	CIM_EthernetPort REF	See 8.1.7.2
OUT	Job	CIM_ConcreteJob REF	See 8.1.7.3

8.1.7.1 VRF

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

8.1.7.2 PortMember

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

8.1.7.3 Job

See 8.1.1

8.1.8 CIM_VRFConfigurationService.AddSAPMember()

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

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

Table 8 contains requirements for parameters of this method.

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

Qualifiers	Name	Type	Description/Values
IN	VRF	CIM_VirtualRoutingAndForwardingTable REF	See 8.1.8.1
IN	SAPMember	CIM_ServiceAccessPoint REF	See 8.1.8.2
OUT	Job	CIM_ConcreteJob REF	See 8.1.8.3

491 #### 8.1.8.1 VRF

492 A required reference to a CIM_VirtualRoutingAndForwardingTable instance to which the instance of
493 CIM_ServiceAccessPoint shall be added.

494 #### 8.1.8.2 SAPMember

495 A required reference to the instance of CIM_ServiceAccessPoint that shall be added to the referenced
496 VRF.

497 #### 8.1.8.3 Job

498 See 8.1.1

499 ### 8.1.9 CIM_VRFConfigurationService.RemoveSAPMember()

500 The implementation of the RemoveSAPMember( ) method is required, the provisions in this subclause
501 shall apply in addition to the behavior applicable to all extrinsic methods as specified in 8.1.

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

505 Table 9 contains requirements for parameters of this method.

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

Qualifiers	Name	Type	Description/Values
IN	VRF	CIM_VirtualRoutingAndForwardingTable	See 8.1.9.1
IN	SAPMember	CIM_ServiceAccessPoint REF	See 8.1.9.2
OUT	Job	CIM_ConcreteJob REF	See 8.1.9.3

507 #### 8.1.9.1 VRF

508 A required reference to a CIM_VirtualRoutingAndForwardingTable instance from which the
509 CIM_ServiceAccessPoint shall be removed.

510 #### 8.1.9.2 SAPMember

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

513 #### 8.1.9.3 Job

514 See 8.1.1

8.2 Profile conventions for operations

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

The default list of operations is as follows:

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

8.3 CIM_HostedService

Table 10 lists implementation requirements for operations. If implemented, these operations shall be implemented as defined in DSP0200. In addition, and unless otherwise stated in Table 10, all operations in the default list in 8 shall be implemented as defined in DSP0200.

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

Table 10 – Operations: CIM_HostedService

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

8.4 CIM_HostedCollection

Table 11 lists implementation requirements for operations. If implemented, these operations shall be implemented as defined in DSP0200. In addition, and unless otherwise stated in Table 10, all operations in the default list in 8 shall be implemented as defined in DSP0200.

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

Table 11 – Operations: CIM_HostedCollection

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

8.5 CIM_ServiceAvailableToElement

Table 12 lists implementation requirements for operations. If implemented, these operations shall be implemented as defined in DSP0200. In addition, and unless otherwise stated in Table 10, all operations in the default list in 8 shall be implemented as defined in DSP0200.

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

Table 12 – Operations: CIM_ServiceAvailableToElement

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

8.6 CIM_ServiceAffectsElement

Table 13 lists implementation requirements for operations. If implemented, these operations shall be implemented as defined in DSP0200. In addition, and unless otherwise stated in Table 10, all operations in the default list in 8 shall be implemented as defined in DSP0200.

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

Table 13 – Operations: CIM_ServiceAffectsElement

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

8.7 CIM_VRFRoute

Table 14 lists implementation requirements for operations. If implemented, these operations shall be implemented as defined in DSP0200. In addition, and unless otherwise stated in Table 10, all operations in the default list in 8 shall be implemented as defined in DSP0200.

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

Table 14 – Operations: CIM_VRFRoute

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

8.8 CIM_MemberOfCollection

Table 15 lists implementation requirements for operations. If implemented, these operations shall be implemented as defined in DSP0200. In addition, and unless otherwise stated in Table 10, all operations in the default list in 8 shall be implemented as defined in DSP0200.

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

Table 15 – Operations: CIM_MemberOfCollection

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

8.9 CIM_VRFConfigurationService

All operations in the default list in 0 shall be implemented as defined in DSP0200.

8.10 CIM_NextHopIPRoute

All operations in the default list in 0 shall be implemented as defined in DSP0200.

8.11 CIM_VirtualRoutingAndForwardingTable

All operations in the default list in 0 shall be implemented as defined in DSP0200.

8.12 CIM_System

All operations in the default list in 0 shall be implemented as defined in DSP0200.

9 Use cases

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

9.1 Profile Registration

The object diagram in Figure 2 shows an example for advertising profile conformance. Figure 2 is an example where an instance of CIM_ComputerSystem that is a compliant Virtual Ethernet Switch is the scoping class.

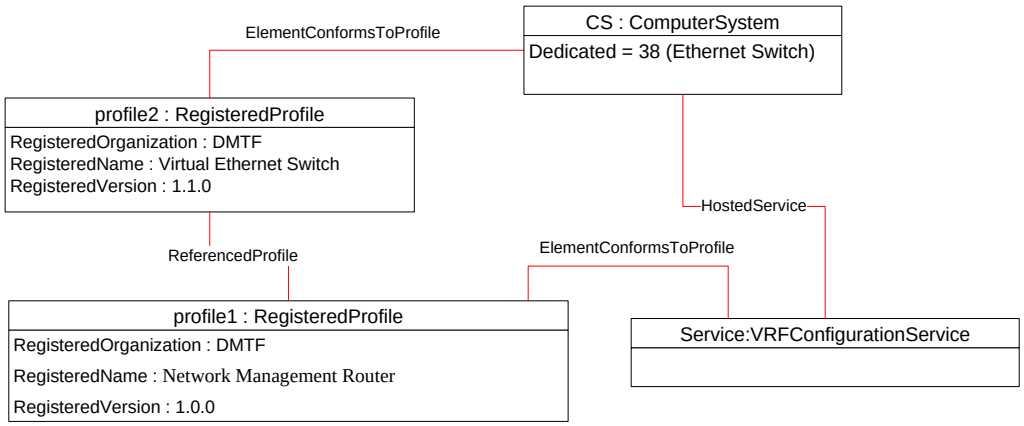

Figure 2 – Registered profile with Computer System Profile

9.2 VRF with Routes and Interfaces

The object diagram is Figure 3 contains the basic element used to model a VRF. The instance diagram shows a CIM_ComputerSystem instance CS0 hosting an instance of CIM_VRFConfigurationService, CS0VRF. In this instance diagram one instance of a VRF, ManagementVRF has been created and is under the CIM_VRFConfigurationService instance CSOVRF. The diagram also shows that the ManagementVRF has two members, one an Ethernet port (E0/1) the other a remote service access point (NPT0).

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

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

- TargetRouter - Reference to CS0:CIM_ComputerSystem
- VRF
  - Embedded instance of CIM_VirtualRoutingAndForwardingTable { ElementName = ManagementVRF }

The method call would return:

- ResultingVRF – Reference to ManagmentVRF:CIM_VirtualRoutingAndForwardingTable.

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

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

The remote service access point NTP0 was added to the VRF with the CIM_VRFConfigurationService.AddMemberSAP() method with the following parameters.

- VRF - Reference to CIM_VirtualRoutingAndForwardingTable:ManagementVRF
- MemberSAP – Reference to CIM_RemoteServiceAccessPoint:NTP0

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

- VRF - Reference to CIM_VirtualRoutingAndForwardingTable:ManagementVRF
- Route -
  - Embedded instance of CIM_NextHopIPRoute {
    DestinationAddress = 192.0.2.0
    DestinationMask = 255.0.0.0
    NextHopInterface = ...E0\1
    NextHopAddress = 192.0.2.4
    AdminDistance = 1 }

The method call would return:

- ResultingRoute – Reference Route0:CIM_NextHopIPRoute.

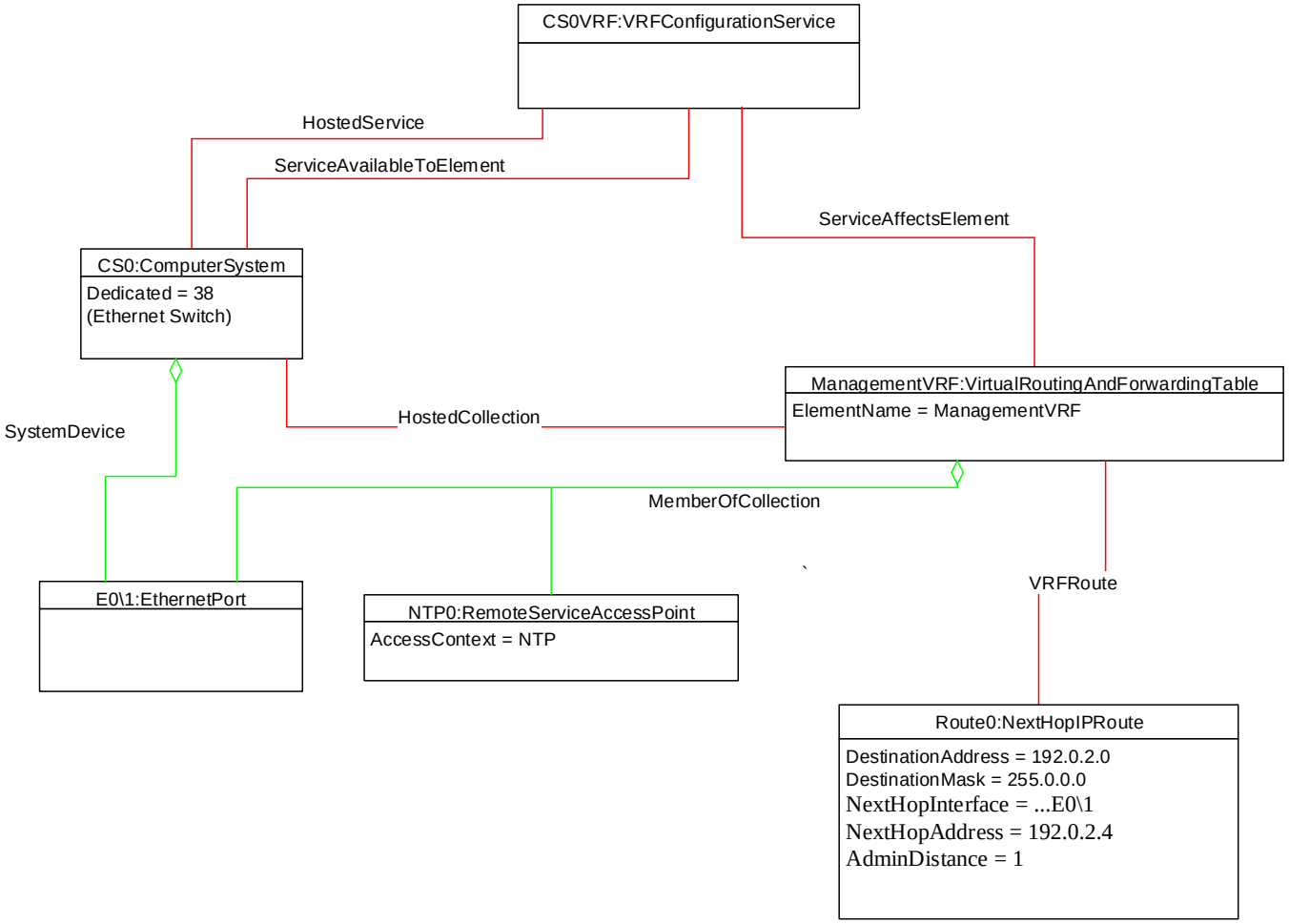

Figure 3 – VRF Instance Diagram

10 CIM Elements

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

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

Element Name	Requirement	Description
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		

10.1 CIM_HostedService

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

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 *

10.2 CIM_VRFRoute

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

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 *

10.3 CIM_RegisteredProfile

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

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 "1.0.0c".
RegisteredOrganization	Mandatory	This property shall have a value of "DMTF".

10.4 CIM_VirtualForwardingAndRoutingTable

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

Table 20 – Class: CIM_VirtualForwardingAndRoutingTable

Elements	Requirement	Description
InstanceID	Mandatory	**Key:** This property shall contain a unique ID to uniquely identify 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.

10.5 CIM_NextHopRoute

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

Table 21 – Class: CIM_NextHopRoute

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

10.6 CIM_NextHopIPRoute

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

Table 22 – Class: CIM_NextHopIPRoute

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

ANNEX A
(informative)

Change log

Version	Date	Description
1.0.0	2018-09-04	