TECHNICAL REPORT

DSL Forum
TR-133

DSLHome™
TR-064 Extensions for Service Differentiation

September 2005

Produced by:
DSLHome-Technical Working Group

Editors:
Jeff Bernstein, 2Wire
Barbara Stark, BellSouth

Working Group Chair:
Greg Bathrick, Texas Instruments

Abstract:

This document specifies the TR-064 extensions needed to provide differentiated service treatment. This includes support for classification, policing, shaping, tagging, queuing, and scheduling.

Notice:

The DSL Forum is a non-profit corporation organized to create guidelines for DSL network system development and deployment. This Technical Report has been approved by members of the Forum. This document is not binding on the DSL Forum, any of its members, or any developer or service provider involved in DSL. The document is subject to change, but only with approval of members of the Forum.

Table of Contents

1 Introduction

This document extends TR-064 to include support for classification, policing, shaping, tagging, queuing, and scheduling of traffic. The extensions defined in this document are intended to be fully backward compatible with TR-064 as currently defined.

2 TR-064 Updates

The updates to TR-064 defined by this specification are given in this section.

2.1 Updates to Section 6.1 "The Model"

This section specifies the changes required in section 6.1 of TR-064, incorporating the QueueManagement object in Figure 5, and the subsequent outline and table of devices and services.

Also in this section, the notation of TR-064 is extended to define Conditional items—items contained in optional tables that are conditionally required if the table itself is implemented.

In this section, items to be added to the original text are shown in red.

Begin Change Text

6.1 The Model

The model shown in Figure 5 uses the UPnP IGD model as its base. UPnP devices and services are shown in black (solid line). New services that are defined in this document are shown in blue (dashed line).

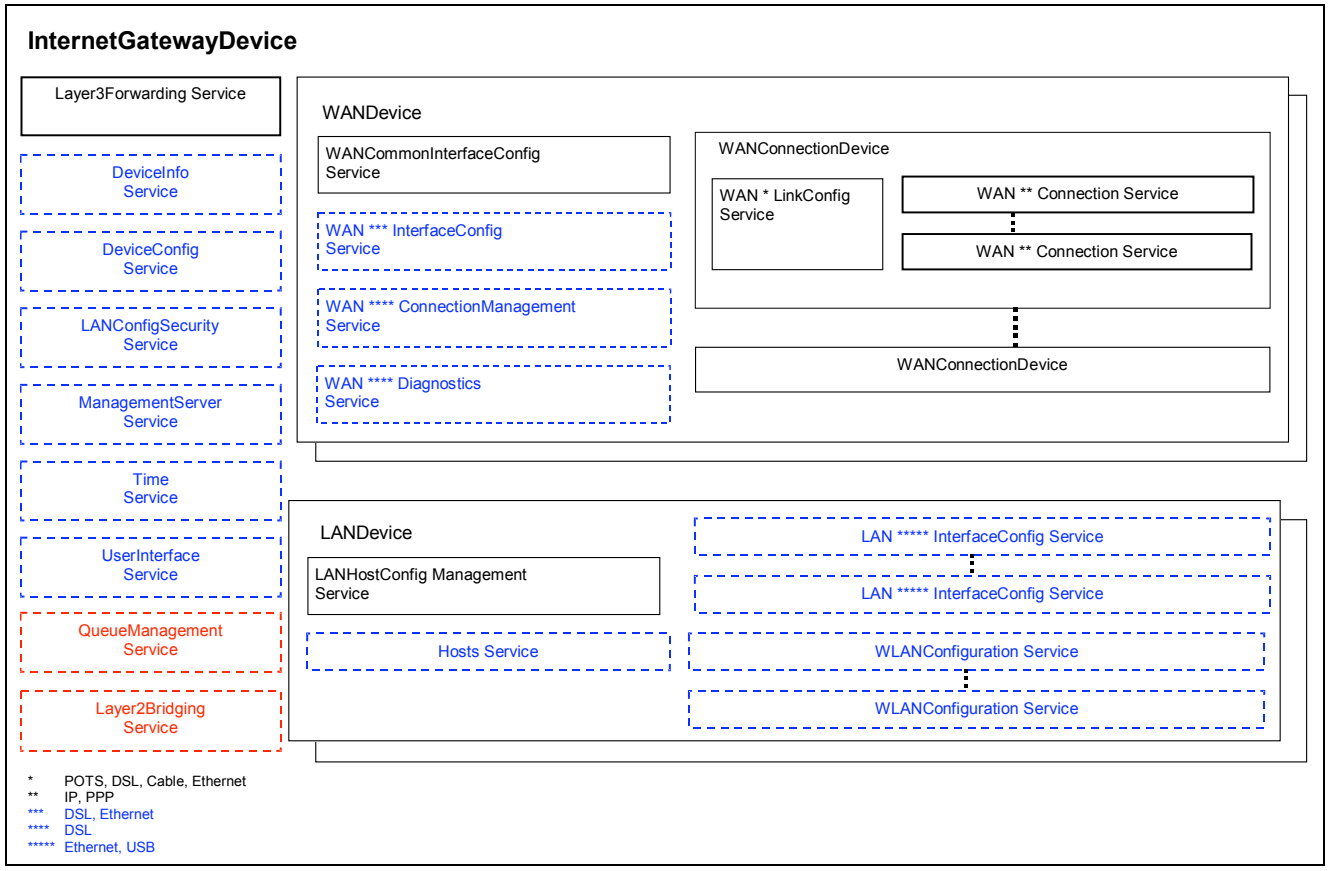

Figure 5 – Extended Internet Gateway Device overview

The following is a tree representation of the additional service templates and their place within the overall UPnP device hierarchy. Devices are indicated in **bold black**; UPnP services are indicated in *black italics*; additional services (Refer to UPnP IGD v1.0 specification for more details on those services) are indicated in blue underline.

InternetGatewayDevice

Layer3Forwarding

DeviceInfo

DeviceConfig

LANConfigSecurity

ManagementServer

Time

UserInterface

QueueManagement

Layer2Bridging

LANDevice

LANHostConfigMgmt

LANEthernetInterfaceConfig

WLANConfiguration

LANUSBInterfaceConfig

Hosts

WANDevice

WANCommonInterfaceConfig

WANDSLInterfaceConfig

WANEthernetInterfaceConfig

WANDSLConnectionManagement

WANDSLDiagnostics

WANConnectionDevice

WANPOTSLinkConfig

WANDSLLinkConfig

WANCableLinkConfig

WANEthernetLinkConfig

WANIPConnection

WANPPPConnection

The following table lists all new or modified devices and services and the implementation requirements.

Device or Service Name	Requirement Status
InternetGatewayDevice	Required for all devices.
Layer3Forwarding	Required for all devices. WT-098 changes required for TR-059 compliance.
DeviceInfo	Required for all devices.
DeviceConfig	Required for all devices.
LANConfigSecurity	Required for all devices.
ManagementServer	Required for CPE that support management via the CPE WAN Management Protocol.
Time	Optional.
UserInterface	Optional.
QueueManagement	Required for TR-059 compliance.
Layer2Bridging	Optional.
LANDevice	Required for all devices.
LANHostConfigMgmt	Required for all devices.
LANEthernetInterfaceConfig	Required for devices that implement a LAN-side Ethernet interface. A managed switch is modelled with 1 LANDevice containing 1 LANEthernetInterfaceConfig for each port of the switch.
WLANConfiguration	Required for devices that implement a LAN-side 802.11abg Wireless Access Point.
LANUSBInterfaceConfig	Required for devices that implement a LAN-side remote NDIS USB Interface.
Hosts	Deferred.
WANDevice	Required for all devices.
WANCommonInterfaceConfig	Required for all devices.
WANDSLInterfaceConfig	Required for devices that implement DSL modem WAN interface.
WANEthernetInterfaceConfig	Required for devices that implement Ethernet WAN interface.
WANDSLConnectionManagement	Required for devices that implement DSL modem WAN interface.
WANDSLDiagnostics	Required for devices with ADSL2 WAN interface, otherwise Optional.
WANConnectionDevice	Required for all devices.
WANPOTSLinkConfig	Required for devices with internal POTS modem or support for external POTS modem through USB or serial interface.
WANDSLLinkConfig	Required for devices the implement DSL modem WAN interface.
WANCableLinkConfig	Required for devices the implement cable modem WAN interface.
WANEthernetLinkConfig	Required for devices the implement WAN-side Ethernet interface.
WANIPConnection	Required for all currently configured IP or bridged connections. WT-098 changes required for TR-059 compliance.
WANPPPConnection	Required for all currently configured PPP connections. WT-098 changes required for TR-059 compliance.

The following abbreviations are used in the tables below:

- R: Required
- O: Optional
- C: Conditional (for items within an optional table, indicates the item is required if the table itself is implemented; otherwise the item is required given the condition specified in the item description)
- D: Deferred (not required by this version of the specification; may be required in future version)
- S: Secured (only applicable to actions), requires authentication; combined with R/O indication
- N/A: Not Applicable

Note: the R/O column of the State Variable and Action tables for each service are conditional on whether the service is implemented at all. For an optional service, for example, an "R" variable or action is required only if the service itself is implemented.

End Change Text

2.2 Updates to Section 6.5.3 "Layer3Forwarding"

This section specifies the changes required in section 6.5.3 of TR-064, which defines the Layer3Forwarding service.

In this section, items to be added to the original text are shown in red, and deletions are shown in ~~blue~~.

Begin Change Text

6.5.3 Layer3Forwarding

6.5.3.1 Overview

This service allows the handling of the routing and forwarding configuration of the device. This service is required.

6.5.3.2 Service Modelling Definitions

ServiceType

urn:dslforum-org:service:Layer3Forwarding:2~~1~~

StateVariables

Variable Name	From IGD[1]	Table	Data Type	Allowed Value	Description	Default Value	R/O
DefaultConnectionService	✓	-	String	String, Max length = 256 characters	Specifies a connection service instance in a WANConnectionDevice. See UPnP Layer3Forwarding:1 v1.01 for more details.	N/A	R
ForwardNumberOfEntries	-	Forwarding (index)	ui2	>=0	Number of forwarding entries.	N/A	R
Enable	-	Forwarding	Boolean	1, 0	Enables or disables the table entry.	N/A	R
Status	-	Forwarding	String	Disabled Enabled Error	Indicates the status of the table entry.	N/A	R
Type	-	Forwarding	String	Default Network Host	Describes the type of route.	N/A	R
DestIPAddress	-	Forwarding KEY	String	IP Address	Destination network address.	N/A	R
DestSubnetMask	-	Forwarding KEY	String	IP Subnet Mask	Destination network subnet mask.	N/A	R
SourceIPAddress	-	Forwarding KEY	String	IP Address	Source network address.	N/A	R
SourceSubnetMask	-	Forwarding KEY	String	IP Subnet Mask	Source network subnet mask.	N/A	R

[1] Indicates whether or not the variable was defined in UPnP IGD 1.0. If not, the variable is specific to this specification.

ForwardingPolicy	-	Forwarding KEY	i2	>=-1	Identifier of a set of classes or flows that have the corresponding ForwardingPolicy value as defined in the QueueManagement service. A value of -1 indicates no ForwardingPolicy is specified. If specified, this forwarding entry is to apply only to traffic associated with the specified classes and flows. Note: Legacy actions that do not include this variable as an input argument MUST assume the default value of -1.	-1	R
GatewayIPAddress	-	Forwarding	String	IP Address	IP address of the gateway.	N/A	R
Interface	-	Forwarding	String	String, nonzero length. Max length = 256 characters	Outgoing interface. 2-tuple referring to WAN**Connection service instance or LANHostConfigManagement service.	N/A	R
ForwardingMetric	-	Forwarding	ui2	>=0	Forwarding metric.	N/A	R
MTU	-	Forwarding	ui2	Between 1 and 1540, inclusive	The maximum allowed size of an Ethernet frame for this route.	N/A	O

Actions, Arguments & Errors

Name	From IGD[2]	Argument	Dir	Related State Variable(s)	Description	Errors	R/O
SetDefaultConnectionService	✓	NewDefaultConnectionService	IN	DefaultConnectionService	Sets the value of the DefaultConnectionService.	402, 501, 720, 721, 723	R S
GetDefaultConnectionService	✓	NewDefaultConnectionService	OUT	DefaultConnectionService	Retrieves the value of the DefaultConnectionService.	402, 501	R
Forwarding Table Actions							
GetForwardNumberOfEntries	-	NewForwardNumberOfEntries	OUT	ForwardNumberOfEntries	Retrieves the value of the ForwardNumberOfEntries state variable.	402, 501	R
AddForwardingEntry	-	NewType NewDestIPAddress NewDestSubnetMask NewSourceIPAddress NewSourceSubnetMask NewGatewayIPAddress NewInterface NewForwardingMetric NewMTU	IN IN IN IN IN IN IN IN IN	Type DestIPAddress DestSubnetMask SourceIPAddress SourceSubnetMask GatewayIPAddress Interface ForwardingMetric MTU	***[Deprecated]*** Inserts an entry in the forwarding table.	402, 501, 701	R S
AddForwardingEntryV2	-	NewType NewDestIPAddress NewDestSubnetMask NewSourceIPAddress NewSourceSubnetMask NewForwardingPolicy NewGatewayIPAddress NewInterface NewForwardingMetric NewMTU	IN IN IN IN IN IN IN IN IN IN	Type DestIPAddress DestSubnetMask SourceIPAddress SourceSubnetMask NewForwardingPolicy GatewayIPAddress Interface ForwardingMetric MTU	Inserts an entry in the forwarding table.	402, 501, 701	R S
DeleteForwardingEntry	-	NewDestIPAddress NewDestSubnetMask NewSourceIPAddress NewSourceSubnetMask	IN IN IN IN	DestIPAddress DestSubnetMask SourceIPAddress SourceSubnetMask	***[Deprecated]*** Deletes an entry in the forwarding table.	402, 501, 702	R S
DeleteForwardingEntryV2	-	NewDestIPAddress NewDestSubnetMask NewSourceIPAddress NewSourceSubnetMask NewForwardingPolicy	IN IN IN IN IN	DestIPAddress DestSubnetMask SourceIPAddress SourceSubnetMask ForwardingPolicy	Deletes an entry in the forwarding table.	402, 501, 702	R S

[2] Indicates whether or not the action was defined in UPnP IGD 1.0. If not, the action is specific to this specification.

GetSpecific ForwardingEntry	-	NewDestIPAddress NewDestSubnetMask NewSourceIPAddress NewSourceSubnetMask NewGatewayIPAddress NewEnable NewStatus NewType NewInterface NewForwardingMetric NewMTU	IN IN IN IN OUT OUT OUT OUT OUT OUT OUT	DestIPAddress DestSubnetMask SourceIPAddress SourceSubnetMask GatewayIPAddress Enable Status Type Interface ForwardingMetric MTU	***[Deprecated]*** Retrieve an entry in the forwarding table.	402, 501, 714	R
GetSpecific ForwardingEntryV2	-	NewDestIPAddress NewDestSubnetMask NewSourceIPAddress NewSourceSubnetMask NewForwardingPolicy NewGatewayIPAddress NewEnable NewStatus NewType NewInterface NewForwardingMetric NewMTU	IN IN IN IN IN OUT OUT OUT OUT OUT OUT OUT	DestIPAddress DestSubnetMask SourceIPAddress SourceSubnetMask ForwardingPolicy GatewayIPAddress Enable Status Type Interface ForwardingMetric MTU	Retrieve an entry in the forwarding table.	402, 501, 714	R
GetGeneric ForwardingEntry	-	NewForwardingIndex NewEnable NewStatus NewType NewDestIPAddress NewDestSubnetMask NewSourceIPAddress NewSourceSubnetMask NewGatewayIPAddress NewInterface NewForwardingMetric NewMTU	IN OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT	ForwardNumberOfEntries Enable Status Type DestIPAddress DestSubnetMask SourceIPAddress SourceSubnetMask GatewayIPAddress Interface ForwardingMetric MTU	***[Deprecated]*** Retrieves Forwarding table one entry at a time. Control points can call this action with an incrementing array index until no more entries are found on the gateway. Index is ranging from 0 to ForwardingNumberOfEntries-1.	402, 501, 713	R
GetGeneric ForwardingEntryV2	-	NewForwardingIndex NewEnable NewStatus NewType NewDestIPAddress NewDestSubnetMask NewSourceIPAddress NewSourceSubnetMask NewForwardingPolicy NewGatewayIPAddress NewInterface NewForwardingMetric NewMTU	IN OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT	ForwardNumberOfEntries Enable Status Type DestIPAddress DestSubnetMask SourceIPAddress SourceSubnetMask ForwardingPolicy GatewayIPAddress Interface ForwardingMetric MTU	Retrieves Forwarding table one entry at a time. Control points can call this action with an incrementing array index until no more entries are found on the gateway. Index is ranging from 0 to ForwardingNumberOfEntries-1.	402, 501, 713	R
SetForwardingEntryEnable	-	NewDestIPAddress NewDestSubnetMask NewSourceIPAddress NewSourceSubnetMask NewEnable	IN IN IN IN IN	DestIPAddress DestSubnetMask SourceIPAddress SourceSubnetMask Enable	Sets the value of the Enable state variable to enable or disable a particular forwarding entry.	402, 501, 702	R S
SetForwardingEntryEnable V2	-	NewDestIPAddress NewDestSubnetMask NewSourceIPAddress NewSourceSubnetMask NewForwardingPolicy NewEnable	IN IN IN IN IN IN	DestIPAddress DestSubnetMask SourceIPAddress SourceSubnetMask ForwardingPolicy Enable	Sets the value of the Enable state variable to enable or disable a particular forwarding entry.	402, 501, 702	R S

6.5.3.3 Theory of Operation

This service models the IP forwarding table of the IGD. Support of the parameters of a "standard" forwarding table entry is required. There are two supported types of forwarding table entries:

- *Destination IP Address Route*
  In this case an entry includes: <DestIPAddress, DestSubnetMask, GatewayIPAddress, Interface, ForwardingMetric>.

  SourceIPAddress and SourceSubnetMask are "0.0.0.0"

- *Source IP Address Route*
  In this case an entry includes: <DestIPAddress, DestSubnetMask, SourceIPAddress, SourceSubnetMask, GatewayIPAddress, Interface, ForwardingMetric>.

When making the forwarding decision, first the source IP address is matched (longest prefix first) followed by destination IP address (longest prefix first) for entries with matching source IP address.

In either case, an entry may also include a ForwardingPolicy value, which restricts the forwarding criteria to classified traffic for only those classes or flows associated with the indicated ForwardingPolicy value. A forwarding entry may include only the ForwardingPolicy as a forwarding criterion, with no specification of source or destination address.

Setting the DefaultConnectionService state variable automatically creates / updates the default interface route to point to the DefaultConnectionService interface.

End Change Text

2.3 Addition of QueueManagement Service

This section specifies a new service to be inserted into TR-064. This service contains queue management functions including classification, queuing, policing, and application-specific flows.

Begin Inserted Text

6.5.x QueueManagement

6.5.x.1 Overview

This service enables the ability to read and set the current classification and hierarchical queuing settings. This service is required for TR-059 compliance. A description of the queuing model associated with this service is provided in Appendix A.

6.5.x.2 Service Modelling Definitions

ServiceType

urn:dslforum-org:service:QueueManagement:1

StateVariables

Variable Name	Table	Data Type	Allowed Value	Description	Default Value	R/O
Enable	-	Boolean	1, 0	Enables or disables all queuing operation.	N/A	R
MaxQueues	-	ui2		The maximum number of queues supported by the CPE. Calculated as the sum of the number of different queues pointed to by Classification table. For each entry in the Classification table, the count includes a queue for each egress interface to which the corresponding classified traffic could reach.	N/A	R
MaxClassificationEntries	-	ui2		The maximum number of entries available in the Classification table.	N/A	R
MaxAppEntries	-	ui2		The maximum number of entries available in the App table.	N/A	R
MaxFlowEntries	-	ui2		The maximum number of entries available in the Flow table.	N/A	R

MaxPolicerEntries	-	ui2		The maximum number of entries available in the Policer table.	N/A	R
MaxQueueEntries	-	ui2		The maximum number of entries available in the Queue table.	N/A	R
DefaultForwardingPolicy	-	ui2	>=0	Identifier of the forwarding policy associated with traffic not associated with any specified classifier.	N/A	R
DefaultPolicer	-	i2	>=-1	Assigned pointer to the Policer array/table for traffic not associated with any specified classifier. A value of -1 indicates a null policer.	N/A	R
DefaultQueue	-	ui2	>=0	Assigned pointer to the Queue array/table for traffic not associated with any specified classifier.	N/A	R
DefaultDSCPMark	-	i2	>=-2	Diffserv Codepoint to mark traffic not associated with any specified classifier. A value of -1 indicates no change from the incoming packet. A value of -2 indicates automatic marking of DSCP based upon the EthernetPriority value as defined in Appendix B.	N/A	R
DefaultEthernetPriorityMark	-	i2	>=-2	Ethernet priority code (as defined in 802.1D) to mark traffic not associated with any specified classifier. A value of -1 indicates no change from the incoming packet. A value of -2 indicates automatic marking of EthernetPriority based upon the DSCP value as defined in Appendix B.	N/A	R
AvailableAppList	-	String	Max length = 1024 characters	Comma-separated list of URNs, each indicating a protocol supported for use as a ProtocolIdentifier in the App table. This list may include any of the URNs defined in Appendix C as well as other URNs defined elsewhere.	N/A	C
ClassificationNumberOfEntries	Classification (index)	ui2	>=0	Number of classification entries	N/A	R
ClassificationKey	Classification KEY	ui2	>=0	Unique key for each classification entry.	N/A	R
ClassificationEnable	Classification	Boolean	1, 0	Enables or disables this classifier.	0	R
ClassificationStatus	Classification	String	Disabled Enabled Error	Indicates the status of this classifier.	"Disabled"	R
ClassificationOrder	Classification	ui2	>=1	Position of the classification entry in the order of precedence. A value of 1 indicates the first entry to be	N/A[3]	R

[3] The value on creation of a Classification table entry MUST be one greater than the largest current value.

				considered. For each packet, the highest ordered entry that matches the classification criteria is applied. All lower order entries are ignored.		
ClassInterface	Classification	String	String Max length = 256 characters	Classification criterion. Ingress interface. 2-tuple referring to a particular WANDevice, WAN-ConnectionDevice, WAN**Connection service, LANDevice, or LAN**InterfaceConfig service. The string "WAN" indicates this entry is to apply to traffic entering from any WAN interface. The string "LAN" indicates this entry is to apply to traffic entering from any LAN interface. The string "Local" indicates this entry is to apply to IP-layer traffic entering from a local source within the Internet Gateway Device. An empty value indicates this classification entry is to apply to all sources.	<Empty>	R
DestIP	Classification	String	IP address	Classification criterion. Destination IP address. An empty value indicates this criterion is not used for classification.	<Empty>	R
DestMask	Classification	String	IP address	Destination IP address Mask. If non-empty, only the indicated network portion of the DestIP address is to be used for classification. An empty value indicates that the full DestIP address is to be used for classification.	<Empty>	R
DestIPExclude	Classification	Boolean	1, 0	If false, the class includes only those packets that match the (masked) DestIP entry, if specified. If true, the class includes all packets except those that match the (masked) DestIP entry, if specified.	0	R
SourceIP	Classification	String	IP address	Classification criterion. Source IP address. An empty value indicates this criterion is not used for classification.	<Empty>	R
SourceMask	Classification	String	IP address	Source IP address Mask. If non-empty, only the indicated network portion of the SourceIP address is to be used for classification. An empty value indicates that the full SourceIP address is to be used for classification.	<Empty>	R
SourceIPExclude	Classification	Boolean	1, 0	If false, the class includes only those packets that match the (masked) SourceIP entry, if specified. If true, the class includes all packets except those that match the (masked) SourceIP entry, if specified.	0	R

Protocol	Classification	i2	>=-1	Classification criterion. Protocol number. A value of -1 indicates this criterion is not used for classification.	-1	R
ProtocolExclude	Classification	Boolean	1, 0	If false, the class includes only those packets that match the Protocol entry, if specified. If true, the class includes all packets except those that match the Protocol entry, if specified.	0	R
DestPort	Classification	i2	>=-1	Classification criterion. Destination port number. A value of -1 indicates this criterion is not used for classification.	-1	R
DestPortRangeMax	Classification	i2	>=-1	Classification criterion. If specified, indicates the classification criterion is to include the port range from DestPort through DestPortRangeMax (inclusive). If specified, DestPortRangeMax MUST be greater than or equal to DestPort. A value of -1 indicates that no port range is specified.	-1	R
DestPortExclude	Classification	Boolean	1, 0	If false, the class includes only those packets that match the DestPort entry (or port range), if specified. If true, the class includes all packets except those that match the DestPort entry (or port range), if specified.	0	R
SourcePort	Classification	i2	>=-1	Classification criterion. Source port number. A value of -1 indicates this criterion is not used for classification.	-1	R
SourcePortRangeMax	Classification	i2	>=-1	Classification criterion. If specified, indicates the classification criterion is to include the port range from SourcePort through SourcePortRangeMax (inclusive). If specified, SourcePortRangeMax MUST be greater than or equal to SourcePort. A value of -1 indicates that no port range is specified.	-1	R
SourcePortExclude	Classification	Boolean	1, 0	If false, the class includes only those packets that match the SourcePort entry (or port range), if specified. If true, the class includes all packets except those that match the SourcePort entry (or port range), if specified.	0	R
SourceMACAddress	Classification	String	MAC Address	Classification criterion. Source MAC Address. An empty value indicates this criterion is not used for classification.	<Empty>	R

SourceMACMask	Classification	String	MAC Address	Bit-mask for the MAC address, where matching of a packet's MAC address with the SourceMAC-Address is only to be done for bit positions set to one in the mask. A mask of FF:FF:FF:FF:FF:FF or an empty string indicates all bits of the SourceMAC-Adress are to be used for classification.	<Empty>	O
SourceMACExclude	Classification	Boolean	1, 0	If false, the class includes only those packets that match the (masked) SourceMACAddress entry, if specified. If true, the class includes all packets except those that match the (masked) SourceMACAddress entry, if specified.	0	R
DestMACAddress	Classification	String	MAC Address	Classification criterion. Destination MAC Address. An empty value indicates this criterion is not used for classification. The use of destination MAC address as a classification criterion is primarily useful only for bridged traffic.	<Empty>	R
DestMACMask	Classification	String	MAC Address	Bit-mask for the MAC address, where matching of a packet's MAC address with the DestMACAddress is only to be done for bit positions set to one in the mask. A mask of FF:FF:FF:FF:FF:FF or an empty string indicates all bits of the DestMAC-Adress are to be used for classification.	<Empty>	O
DestMACExclude	Classification	Boolean	1, 0	If false, the class includes only those packets that match the (masked) Dest-MACAddress entry, if specified. If true, the class includes all packets except those that match the (masked) DestMACAddress entry, if specified.	0	R
Ethertype	Classification	i2	>=-1	Classification criterion. Ethertype as indicated in either the Ethernet or SNAP Type header. A value of -1 indicates this criterion is not used for classification.	-1	O
EthertypeExclude	Classification	Boolean	1, 0	If false, the class includes only those packets that match the Ethertype entry, if specified. If true, the class includes all packets except those that match the Ethertype entry, if specified.	0	O
SSAP	Classification	i1	>=-1	Classification criterion. SSAP element in the LLC header. A value of -1 indicates this criterion is not used for classification.	-1	O
SSAPExclude	Classification	Boolean	1, 0	If false, the class includes only those packets that match the SSAP entry, if specified.	0	O

				If true, the class includes all packets except those that match the SSAP entry, if specified.		
DSAP	Classification	i1	>=-1	Classification criterion. DSAP element in the LLC header. A value of -1 indicates this criterion is not used for classification.	-1	O
DSAPExclude	Classification	Boolean	1, 0	If false, the class includes only those packets that match the DSAP entry, if specified. If true, the class includes all packets except those that match the DSAP entry, if specified.	0	O
LLCControl	Classification	i2	>=-1	Classification criterion. Control element in the LLC header. A value of -1 indicates this criterion is not used for classification.	-1	O
LLCControlExclude	Classification	Boolean	1, 0	If false, the class includes only those packets that match the LLCControl entry, if specified. If true, the class includes all packets except those that match the LLCControl entry, if specified.	0	O
SNAPOUI	Classification	i4	>=-1	Classification criterion. OUI element in the SNAP header. A value of -1 indicates this criterion is not used for classification.	-1	O
SNAPOUIExclude	Classification	Boolean	1, 0	If false, the class includes only those packets that match the SNAPOUI entry, if specified. If true, the class includes all packets except those that match the SNAPOUI entry, if specified.	0	O
SourceVendorClassID	Classification	String	Max length = 256 characters	Classification criterion. Used to identify one or more LAN devices, value of the DHCP Vendor Class Identifier (Option 60) as defined in RFC 2132. An empty value indicates this criterion is not used for classification.	<Empty>	O
SourceVendorClassIDExclude	Classification	Boolean	1, 0	If false, the class includes only those packets sourced from LAN devices that match the SourceVendor-ClassID entry, if specified. If true, the class includes all packets except those sourced from LAN devices that match the Source-VendorClassID entry, if specified.	0	O
DestVendorClassID	Classification	String	Max length = 256 characters	Classification criterion. Used to identify one or more LAN devices, value of the DHCP Vendor Class Identifier (Option 60) as defined in RFC 2132. An empty value indicates this criterion is not used for classification.	<Empty>	O

DestVendorClassIDExclude	Classification	Boolean	1, 0	If false, the class includes only those packets destined for LAN devices that match the Dest-VendorClassID entry, if specified. If true, the class includes all packets except those destined for LAN devices that match the Dest-VendorClassID entry, if specified.	0	O
SourceClientID	Classification	String	Max length = 256 characters	Classification criterion. Used to identify one or more LAN devices, value of the DHCP Client Identifier (Option 61) as defined in RFC 2132. An empty value indicates this criterion is not used for classification.	<Empty>	O
SourceClientIDExclude	Classification	Boolean	1, 0	If false, the class includes only those packets sourced from LAN devices that match the SourceClientID entry, if specified. If true, the class includes all packets except those sourced from LAN devices that match the Source-ClientID entry, if specified.	0	O
DestClientID	Classification	String	Max length = 256 characters	Classification criterion. Used to identify one or more LAN devices, value of the DHCP Client Identifier (Option 61) as defined in RFC 2132. An empty value indicates this criterion is not used for classification.	<Empty>	O
DestClientIDExclude	Classification	Boolean	1, 0	If false, the class includes only those packets destined for LAN devices that match the Dest-ClientID entry, if specified. If true, the class includes all packets except those destined for LAN devices that match the Dest-ClientID entry, if specified.	0	O
SourceUserClassID	Classification	String	Max length = 256 characters	Classification criterion. Used to identify one or more LAN devices, value of the DHCP User Class Identifier (Option 77) as defined in RFC 3004. An empty value indicates this criterion is not used for classification.	<Empty>	O
SourceUserClassIDExclude	Classification	Boolean	1, 0	If false, the class includes only those packets sourced from LAN devices that match the Source-UserClassID entry, if specified. If true, the class includes all packets except those sourced from LAN devices that match the Source-UserClassID entry, if specified.	0	O
DestUserClassID	Classification	String	Max length = 256 characters	Classification criterion. Used to identify one or	<Empty>	O

				more LAN devices, value of the DHCP User Class Identifier (Option 77) as defined in RFC 3004. An empty value indicates this criterion is not used for classification.		
DestUserClassIDExclude	Classification	Boolean	1, 0	If false, the class includes only those packets destined for LAN devices that match the Dest-UserClassID entry, if specified. If true, the class includes all packets except those destined for LAN devices that match the Dest-UserClassID entry, if specified.	0	O
TCPACK	Classification	Boolean	1, 0	Classification criterion. If false, this criterion is not used for classification. If true, this criterion matches with all TCP segments that have the ACK control bit set.	0	O
TCPACKExclude	Classification	Boolean	1, 0	If false, the class includes only those packets that match the TCPACK entry, if specified. If true, the class includes all packets except those that match the TCPACK entry, if specified.	0	O
IPLengthMin	Classification	Ui2	>=0	Classification criterion. Minimum IP Packet Length (including header) in bytes.	0	O
IPLengthMax	Classification	Ui2	>=0	Classification criterion. Maximum IP Packet Length (including header) in bytes. A value of zero indicates that no maximum is specified (an umlimited maximum length).	0	O
IPLengthExclude	Classification	Boolean	1, 0	If false, the class includes only those packets whose length (including header) falls within the inclusive range IPLengthMin through IPLengthMax. A value of zero for both IPLengthMin and IPLengthMax allows any length packet. An equal non-zero value of IPLengthMin and IPLengthMax allows only a packets with the exact length specified. If true, the class includes all packets except those whose length (including header) falls within the inclusive range IPLengthMin through IPLengthMax.	0	O
DSCPCheck	Classification	i2	>=-1	Classification criterion. Diffserv Codepoint (defined in RFC 2474). If set to a Class Selector Codepoint (defined in RFC 2474), all DSCP values that match the first 3 bits will be considered a valid	-1	R

				match. A value of -1 indicates this criterion is not used for classification.		
DSCPExclude	Classification	Boolean	1, 0	If false, the class includes only those packets that match the DSCPCheck entry, if specified. If true, the class includes all packets except those that match the DSCPCheck entry, if specified.	0	R
DSCPMark	Classification	i2	>=-2	Classification result. DiffServ Codepoint to mark traffic with that falls into this Classification entry. A value of -1 indicates no change from the incoming packet. A value of -2 indicates automatic marking of DSCP based upon the EthernetPriority value as defined in Appendix B.	-1	R
EthernetPriorityCheck	Classification	i2	>=-1	Classification criterion. Current Ethernet priority as defined in 802.1D. A value of -1 indicates this criterion is not used for classification.	-1	R
EthernetPriorityExclude	Classification	Boolean	1, 0	If false, the class includes only those packets that match the EthernetPriorityCheck entry, if specified. If true, the class includes all packets except those that match the EthernetPriorityCheck entry, if specified.	0	R
EthernetPriorityMark	Classification	i2	>=-2	Classification result. Ethernet priority code (as defined in 802.1D) to mark traffic with that falls into this Classification entry. A value of -1 indicates no change from the incoming packet. A value of -2 indicates automatic marking of EthernetPriority based upon the DSCP value as defined in Appendix B.	-1	R
VLANIDCheck	Classification	i2	>=-1	Classification criterion. Current VLAN ID. Ethernet VLAN ID as defined in 802.1Q. A value of -1 indicates this criterion is not used for classification.	-1	R
VLANIDExclude	Classification	Boolean	1, 0	If false, the class includes only those packets that match the VLANIDCheck entry, if specified. If true, the class includes all packets except those that match the VLANIDCheck entry, if specified.	0	R
ForwardingPolicy	Classification	ui2	>=0	Identifier of the forwarding policy associated with traffic that falls into this	0	R

				classification.		
ClassPolicer	Classification	i2	>=-1	Classification result. Assigned pointer to the Policer array/table for traffic that falls in this classification. A value of -1 indicates a null policer.	-1	R
ClassQueue	Classification	i2	>=-1	Classification result. Assigned pointer to the Queue array/table for traffic that falls in this classification. A value of -1 indicates a null queue. ClassQueue and ClassApp are mutually exclusive and one of the two must be specified. If ClassQueue is null, ClassApp must be specified, and vice versa.	-1	R
ClassApp	Classification	I2	>=-1	Classification result. Assigned pointer to the App array/table for traffic that falls in this classification. A value of -1 indicates a null app entry. ClassQueue and ClassApp are mutually exclusive and one of the two must be specified. If ClassQueue is null, ClassApp must be specified, and vice versa.	-1	C
AppNumberOfEntries	App (index)	ui2	>=0	Number of entries in the App table.	N/A	C
AppKey	App KEY	ui2	>=0	Unique key for each entry in the App table.	N/A	C
AppEnable	App	Boolean	1, 0	Enables or disables this App table entry.	0	C
AppStatus	App	String	Disabled Enabled Error	Indicates the status of this App table entry.	“Disabled”	C
ProtocolIdentifier	App	String	Max length = 256 characters	URN identifying the protocol associated with the given application. A set of defined URNs is given in Appendix B.	<Empty>	C
AppName	App	String	Max length = 64 characters	Human-readable name associated with this entry in the App table.	<Empty>	C
AppDefaultForwardingPolicy	App	ui2	>=0	Identifier of the forwarding policy associated with this App table entry, but not associated with any specified flow.	0	C
AppDefaultPolicer	App	i2	>=-1	Assigned pointer to the Policer array/table for traffic associated with this App table entry, but not associated with any specified flow. A value of -1 indicates a null policer.	-1	C
AppDefaultQueue	App	i2	>=-1	Assigned pointer to the Queue array/table for traffic associated with this App table entry, but not	-1	C

				associated with any specified flow. A value of -1 indicates a null queue.		
AppDefaultDSCPMark	App	i2	>=-2	DiffServ Codepoint to mark traffic associated with this App table entry, but not associated with any specified flow. A value of -1 indicates no change from the incoming packet. A value of -2 indicates automatic marking of DSCP based upon the EthernetPriority value as defined in Appendix B.	-1	C
AppDefaultEthernetPriorityMark	App	i2	>=-2	Ethernet priority code (as defined in 802.1D) to mark traffic associated with this App table entry, but not associated with any specified flow. A value of -1 indicates no change from the incoming packet. A value of -2 indicates automatic marking of EthernetPriority based upon the DSCP value as defined in Appendix B.	-1	C
FlowNumberOfEntries	Flow (index)	ui2	>=0	Number of entries in the Flow table.	N/A	C
FlowKey	Flow KEY	ui2	>=0	Unique key for each entry in the flow table.	N/A	C
FlowEnable	Flow	Boolean	1, 0	Enables or disables this Flow table entry.	0	C
FlowStatus	Flow	String	Disabled Enabled Error	Indicates the status of this Flow table entry.	“Disabled”	C
FlowType	Flow	String	Max length = 256 characters	URN identifying the type of flow to be associated with the specified queue and policer. A set of defined URNs is given in Appendix C.	<Empty>	C
FlowTypeParameters	Flow	String	Max length = 256 characters	List of name-value pairs representing additional criteria to identify the flow type. The use and interpretation is specific to the particular FlowType URN. Encoded using the “x-www-form-urlencoded” content type defined in [5].	<Empty>	C
FlowName	Flow	String	Max length = 64 characters	Human readable name associated with this entry in the Flowtable.	<Empty>	C
AppIdentifier	Flow	i2	>=-1	Key of the App table entry associated with this flow. A value of -1 indicates the flow table is not associated with any App table entry.	-1	C
FlowForwardingPolicy	Flow	ui2	>=0	Identifier of the forwarding policy associated with this flow.	0	C
FlowPolicer	Flow	i2	>=-1	Assigned pointer to the Policer array/table for traffic that falls in this flow. A value of -1 indicates a	-1	C

				null policer.		
FlowQueue	Flow	i2	>=-1	Assigned pointer to the Queue array/table for traffic that falls in this flow. A value of -1 indicates a null queue.	-1	C
FlowDSCPMark	Flow	i2	>=-2	DiffServ Codepoint to mark traffic with that falls into this flow. A value of -1 indicates no change from the incoming packet. A value of -2 indicates automatic marking of DSCP based upon the EthernetPriority value as defined in Appendix B.	-1	C
FlowEthernetPriorityMark	Flow	i2	>=-2	Ethernet priority code (as defined in 802.1D) to mark traffic with that falls into this flow. A value of -1 indicates no change from the incoming packet. A value of -2 indicates automatic marking of EthernetPriority based upon the DSCP value as defined in Appendix B.	-1	C
PolicerNumberOfEntries	Policer (index)	ui2	>=0	Number of policer entries	N/A	R
PolicerKey	Policer KEY	ui2	>=0	Unique key for each policer entry.	N/A	R
PolicerEnable	Policer	Boolean	1, 0	Enables or disables this policer	0	R
PolicerStatus	Policer	String	Disabled Enabled Error	Indicates the status of this policer.	"Disabled"	R
CommittedRate	Policer	ui4	>=0	Committed rate allowed for this policer in bits-per-second.	0	R
CommittedBurstSize	Policer	ui4	>=0	Committed Burstsize in bytes.	0	R
ExcessBurstSize	Policer	ui4	>=0	Excess Burstsize in bytes. Applied for a SingleRateThreeColor meter.	0	O
PeakRate	Policer	ui4	>=0	Peak rate allowed for this Meter in bits-per-second. Applied for TwoRateThreeColor meters.	0	O
PeakBurstSize	Policer	ui4	>=0	Peak Burstsize in bytes. Applied for TwoRateThreeColor meters.	0	O
MeterType	Policer	String	SimpleTokenBucket SingleRateThreeColor TwoRateThreeColor	Identifies the method of traffic measurement to be used for this policer. SimpleTokenBucket makes use of CommittedRate and CommittedBurstSize. SingleRateThreeColor makes use of	"SimpleToken-Bucket"	R

				CommittedRate, CommittedBurstSize, and ExcessBurstSize as defined in RFC 2697. TwoRateThreeColor makes use of CommittedRate, CommittedBurstSize, PeakRate, and PeakBurstSize as defined in RFC 2698.		
PossibleMeterTypes	Policer	String	SimpleTokenBucket SingleRateThreeColor TwoRateThreeColor	Comma-separated list of supported meter types.	N/A	R
ConformingAction	Policer	String	Null Drop Count <DSCP Value>	Instructions for how to handle traffic that is conforming. Null corresponds with no action. A Count action increases the meter instance count statistics. <DSCP Value> is an unsigned integer that corresponds with a mark action overwriting the traffic's DSCP with the configured DSCP.	"Null"	R
PartialConformingAction	Policer	String	Null Drop Count <DSCP Value>	Instructions for how to handle traffic that is partially conforming (colored yellow). Null corresponds with no action. A Count action increases the meter instance count statistics. <DSCP Value> is an unsigned integer that corresponds with a mark action overwriting the traffic's DSCP with the configured DSCP. Only applies for three-color meters.	"Drop"	O
NonConformingAction	Policer	String	Null Drop Count <DSCP Value>	Instructions for how to handle traffic that is non-conforming. Null corresponds with no action. A Count action increases the meter instance count statistics. <DSCP Value> is an unsigned integer that corresponds with a mark action overwriting the traffic's DSCP with the configured DSCP.	"Drop"	R
CountedPackets	Policer	ui4	>=0	Number of Packets counted as result of a count meter action.	0	R
CountedBytes	Policer	ui4	>=0	Number of Bytes counted as result of a count meter action.	0	R
QueueNumberOfEntries	Queue (index)	ui2	>=0	Indicates the number of queue entries	N/A	R
QueueKey	Queue KEY	ui2	>=0	Unique key for each queue entry.	N/A	R
QueueEnable	Queue	Boolean	1, 0	Enables or disables this queue.	0	R

QueueStatus	Queue	String	Disabled Enabled Error	Indicates the status of this queue.	“Disabled”	R
QueueInterface	Queue	String	String Max length = 256 characters	Egress interfaces for which the specified queue must exist. Packets classified into this queue that exit through any other interface must instead use the default queuing behavior. This parameter must be in one of the following forms: 2-tuple referring to a particular WANDevice, WANConnection-Device, WAN**-Connection service, LANDevice, or LAN**InterfaceConfig service. The string “WAN”, which indicates this entry applies to all WAN interfaces. The string “LAN”, which indicates this entry applies to all LAN interfaces. An empty value, which indicates this classification entry is to apply to all interfaces. Packets classified into this queue that exit through any other interface must instead use the default queuing behavior specified in the Queue table entry referenced by the DefaultQueue variable. For the default queue itself (the Queue table entry referenced by the DefaultQueue variable), the value of the QueueInterface parameter MUST be ignored. That is, the default queue must exist on all egress interfaces.	<Empty>	R
QueueBufferLength	Queue	ui4	>0	Number of bytes in the buffer	N/A	R
QueueWeight	Queue	ui2	>=0	Weight of this queue in case of WFQ or WRR, but only used for queues of equal precedence	0	R
QueuePrecedence	Queue	ui2	>0	Precedence of this queue relative to others. Lower numbers imply greater precedence	1	R
REDThreshold	Queue	ui2		Random Early Discard threshold; in case the DropAlgorithm is RED, this is the threshold to use	0	R
REDPercentage	Queue	ui2		Random Early Discard percentage; in case the DropAlgorithm is RED, this is the percentage to use	0	R
DropAlgorithm	Queue	String	RED DT WRED, BLUE	Dropping algorithm used for this queue if congested)	“DT”	R
SchedulerAlgorithm	Queue	String	WFQ WRR SP (Strict Priority)	Scheduling Algorithm used by scheduler.	“SP”	R

ShapingRate	Queue	i4	>=-1	Rate to shape this queue's traffic. If <= 100, in percent of the rate of the highest rate-constrained layer over which the packet will travel on egress.[4] The rate is limited over the window period specified by ShapeWindow. If > 100, in bits per second. A value of -1 indicates no shaping.	-1	R
ShapingBurstSize	Queue	ui2	>=0	Burst Size in bytes.	0	R

Actions, Arguments & Errors

Name	Argument	Dir	Related State Variable(s)	Description	Errors	R/O
GetInfo	NewEnable	OUT	Enable	Retrieves all of the state variables not associated with a table.	402, 501	R
	NewMaxQueues	OUT	MaxQueues			
	NewMaxClassificationEntries	OUT	MaxClassificationEntries			
	NewMaxAppEntries	OUT	MaxAppEntries			
	NewMaxFlowEntries	OUT	MaxFlowEntries			
	NewMaxPolicerEntries	OUT	MaxPolicerEntries			
	NewMaxQueueEntries	OUT	MaxQueueEntries			
	NewDefaultForwardingPolicy	OUT	DefaultForwardingPolicy			
	NewDefaultPolicer	OUT	DefaultPolicer			
	NewDefaultQueue	OUT	DefaultQueue			
	NewDefaultDSCPMark	OUT	DefaultDSCPMark			
	NewDefaultEthernetPriorityMark	OUT	DefaultEthernetPriorityMark			
	NewAvailableAppList	OUT	AvailableAppList			
SetEnable	NewEnable	IN	Enable	Sets the value of the Enable state variable to enable or disable this service.	402, 501	R S
SetDefaultBehavior	NewDefaultForwardingPolicy	IN	DefaultForwardingPolicy	Sets the default behavior of traffic not associated with any specified classifier.	402, 501	R S
	NewDefaultPolicer	IN	DefaultPolicer			
	NewDefaultQueue	IN	DefaultQueue			
	NewDefaultDSCPMark	IN	DefaultDSCPMark			
	NewDefaultEthernetPriorityMark	IN	DefaultEthernetPriorityMark			
Classification Table Actions						
AddClassificationEntry	NewClassificationEnable	IN	ClassificationEnable	Inserts an entry in the Classification table.	402, 501, 701	R S
	NewClassificationOrder	IN	ClassificationOrder			
	NewClassInterface	IN	ClassInterface			
	NewDestIP	IN	DestIP			
	NewDestMask	IN	DestMask			
	NewDestIPExclude	IN	DestIPExclude			
	NewSourceIP	IN	SourceIP			
	NewSourceMask	IN	SourceMask			
	NewSourceIPExclude	IN	SourceIPExclude			
	NewProtocol	IN	Protocol			
	NewProtocolExclude	IN	ProtocolExclude			
	NewDestPort	IN	DestPort			
	NewDestPortRangeMax	IN	DestPortRangeMax			
	NewDestPortExclude	IN	DestPortExclude			
	NewSourcePort	IN	SourcePort			
	NewSourcePortRangeMax	IN	SourcePortRangeMax			
	NewSourcePortExclude	IN	SourcePortExclude			
	NewSourceMACAddress	IN	SourceMACAddress			
	NewSourceMACMask	IN	SourceMACMask			
	NewSourceMACExclude	IN	SourceMACExclude			
	NewDestMACAddress	IN	DestMACAddress			
	NewDestMACMask	IN	DestMACMask			
	NewDestMACExclude	IN	DestMACExclude			
	NewEthertype	IN	Ethertype			
	NewEthertypeExclude	IN	EthertypeExclude			
	NewSSAP	IN	SSAP			
	NewSSAPExclude	IN	SSAPExclude			
	NewDSAP	IN	DSAP			
	NewDSAPExclude	IN	DSAPExclude			

[4] For example, for packets destined for a WAN DSL interface, if the egress will be on a PPP or IP link with a specified ShapeTo rate, the percentage is calculated relative to this rate. Otherwise, if the ATM layer is rate-constrained, then the rate is calculated relative to this rate. Otherwise, the rate is calculated relative to the physical-layer DSL rate.

	NewLLCControl	IN	LLCControl			
	NewLLCControlExclude	IN	LLCControlExclude			
	NewSNAPOUI	IN	SNAPOUI			
	NewSNAPOUIExclude	IN	SNAPOUIExclude			
	NewSourceVendorClassID	IN	SourceVendorClassID			
	NewSourceVendorClassIDExclude	IN	SourceVendorClassIDExclude			
	NewDestVendorClassID	IN	DestVendorClassID			
	NewDestVendorClassIDExclude	IN	DestVendorClassIDExclude			
	NewSourceClientID	IN	SourceClientID			
	NewSourceClientIDExclude	IN	SourceClientIDExclude			
	NewDestClientID	IN	DestClientID			
	NewDestClientIDExclude	IN	DestClientIDExclude			
	NewSourceUserClassID	IN	SourceUserClassID			
	NewSourceUserClassIDExclude	IN	SourceUserClassIDExclude			
	NewDestUserClassID	IN	DestUserClassID			
	NewDestUserClassIDExclude	IN	DestUserClassIDExclude			
	NewTCPACK	IN	TCPACK			
	NewTCPACKExclude	IN	TCPACKExclude			
	NewIPLengthMin	IN	IPLengthMin			
	NewIPLengthMax	IN	IPLengthMax			
	NewIPLengthExclude	IN	IPLengthExclude			
	NewDSCPCheck	IN	DSCPCheck			
	NewDSCPExclude	IN	DSCPExclude			
	NewDSCPMark	IN	DSCPMark			
	NewEthernetPriorityCheck	IN	EthernetPriorityCheck			
	NewEthernetProirityExclude	IN	EthernetProirityExclude			
	NewEthernetPriorityMark	IN	EthernetPriorityMark			
	NewVLANIDCheck	IN	VLANIDCheck			
	NewVLANIDExclude	IN	VLANIDExclude			
	NewForwardingPolicy	IN	ForwardingPolicy			
	NewClassPolicer	IN	ClassPolicer			
	NewClassQueue	IN	ClassQueue			
	NewClassApp	IN	ClassApp			
	NewClassificationKey	OUT	ClassificationKey			
DeleteClassificationEntry	NewClassificationKey	IN	ClassificationKey	Deletes an entry in the Classification table.	402, 501, 702	R S
GetSpecific ClassificationEntry	NewClassificationKey	IN	ClassificationKey	Retrieve an entry in the Classification table.	402, 501, 714	R
	NewClassificationEnable	OUT	ClassificationEnable			
	NewClassificationStatus	OUT	ClassificationStatus			
	NewClassificationOrder	OUT	ClassificationOrder			
	NewClassInterface	OUT	ClassInterface			
	NewDestIP	OUT	DestIP			
	NewDestMask	OUT	DestMask			
	NewDestIPExclude	OUT	DestIPExclude			
	NewSourceIP	OUT	SourceIP			
	NewSourceMask	OUT	SourceMask			
	NewSourceIPExclude	OUT	SourceIPExclude			
	NewProtocol	OUT	Protocol			
	NewProtocolExclude	OUT	ProtocolExclude			
	NewDestPort	OUT	DestPort			
	NewDestPortRangeMax	OUT	DestPortRangeMax			
	NewDestPortExclude	OUT	DestPortExclude			
	NewSourcePort	OUT	SourcePort			
	NewSourcePortRangeMax	OUT	SourcePortRangeMax			
	NewSourcePortExclude	OUT	SourcePortExclude			
	NewSourceMACAddress	OUT	SourceMACAddress			
	NewSourceMACMask	OUT	SourceMACMask			
	NewSourceMACExclude	OUT	SourceMACExclude			
	NewDestMACAddress	OUT	DestMACAddress			
	NewDestMACMask	OUT	DestMACMask			
	NewDestMACExclude	OUT	DestMACExclude			
	NewEthertype	OUT	Ethertype			
	NewEthertypeExclude	OUT	EthertypeExclude			
	NewSSAP	OUT	SSAP			
	NewSSAPExclude	OUT	SSAPExclude			
	NewDSAP	OUT	DSAP			
	NewDSAPExclude	OUT	DSAPExclude			
	NewLLCControl	OUT	LLCControl			
	NewLLCControlExclude	OUT	LLCControlExclude			
	NewSNAPOUI	OUT	SNAPOUI			
	NewSNAPOUIExclude	OUT	SNAPOUIExclude			
	NewSourceVendorClassID	OUT	SourceVendorClassID			
	NewSourceVendorClassIDExclude	OUT	SourceVendorClassIDExclude			
	NewDestVendorClassID	OUT	DestVendorClassID			
	NewDestVendorClassIDExclude	OUT	DestVendorClassIDExclude			
	NewSourceClientID	OUT	SourceClientID			
	NewSourceClientIDExclude	OUT	SourceClientIDExclude			
	NewDestClientID	OUT	DestClientID			
	NewDestClientIDExclude	OUT	DestClientIDExclude			
	NewSourceUserClassID	OUT	SourceUserClassID			
	NewSourceUserClassIDExclude	OUT	SourceUserClassIDExclude			
	NewDestUserClassID	OUT	DestUserClassID			
	NewDestUserClassIDExclude	OUT	DestUserClassIDExclude			
	NewTCPACK	OUT	TCPACK			
	NewTCPACKExclude	OUT	TCPACKExclude			
	NewIPLengthMin	OUT	IPLengthMin			
	NewIPLengthMax	OUT	IPLengthMax			
	NewIPLengthExclude	OUT	IPLengthExclude			
	NewDSCPCheck	OUT	DSCPCheck			
	NewDSCPExclude	OUT	DSCPExclude			

	NewDSCPMark	OUT	DSCPMark			
	NewEthernetPriorityCheck	OUT	EthernetPriorityCheck			
	NewEthernetProirityExclude	OUT	EthernetProirityExclude			
	NewEthernetPriorityMark	OUT	EthernetPriorityMark			
	NewVLANIDCheck	OUT	VLANIDCheck			
	NewVLANIDExclude	OUT	VLANIDExclude			
	NewForwardingPolicy	OUT	ForwardingPolicy			
	NewClassPolicer	OUT	ClassPolicer			
	NewClassQueue	OUT	ClassQueue			
	NewClassApp	OUT	ClassApp			
GetGeneric ClassificationEntry	NewClassificationIndex	IN	ClassificationNumberOfEntries	Retrieves the Classification table one entry at a time. Control points can call this action with an incrementing array index until no more entries are found on the gateway. Index is ranging from 0 to ClassificationNumberOf-Entries-1.	402, 501, 713	R
	NewClassificationKey	OUT	ClassificationKey			
	NewClassificationEnable	OUT	ClassificationEnable			
	NewClassificationStatus	OUT	ClassificationStatus			
	NewClassificationOrder	OUT	ClassificationOrder			
	NewClassInterface	OUT	ClassInterface			
	NewDestIP	OUT	DestIP			
	NewDestMask	OUT	DestMask			
	NewDestIPExclude	OUT	DestIPExclude			
	NewSourceIP	OUT	SourceIP			
	NewSourceMask	OUT	SourceMask			
	NewSourceIPExclude	OUT	SourceIPExclude			
	NewProtocol	OUT	Protocol			
	NewProtocolExclude	OUT	ProtocolExclude			
	NewDestPort	OUT	DestPort			
	NewDestPortRangeMax	OUT	DestPortRangeMax			
	NewDestPortExclude	OUT	DestPortExclude			
	NewSourcePort	OUT	SourcePort			
	NewSourcePortRangeMax	OUT	SourcePortRangeMax			
	NewSourcePortExclude	OUT	SourcePortExclude			
	NewSourceMACAddress	OUT	SourceMACAddress			
	NewSourceMACMask	OUT	SourceMACMask			
	NewSourceMACExclude	OUT	SourceMACExclude			
	NewDestMACAddress	OUT	DestMACAddress			
	NewDestMACMask	OUT	DestMACMask			
	NewDestMACExclude	OUT	DestMACExclude			
	NewEthertype	OUT	Ethertype			
	NewEthertypeExclude	OUT	EthertypeExclude			
	NewSSAP	OUT	SSAP			
	NewSSAPExclude	OUT	SSAPExclude			
	NewDSAP	OUT	DSAP			
	NewDSAPExclude	OUT	DSAPExclude			
	NewLLCControl	OUT	LLCControl			
	NewLLCControlExclude	OUT	LLCControlExclude			
	NewSNAPOUI	OUT	SNAPOUI			
	NewSNAPOUIExclude	OUT	SNAPOUIExclude			
	NewSourceVendorClassID	OUT	SourceVendorClassID			
	NewSourceVendorClassIDExclude	OUT	SourceVendorClassIDExclude			
	NewDestVendorClassID	OUT	DestVendorClassID			
	NewDestVendorClassIDExclude	OUT	DestVendorClassIDExclude			
	NewSourceClientID	OUT	SourceClientID			
	NewSourceClientIDExclude	OUT	SourceClientIDExclude			
	NewDestClientID	OUT	DestClientID			
	NewDestClientIDExclude	OUT	DestClientIDExclude			
	NewSourceUserClassID	OUT	SourceUserClassID			
	NewSourceUserClassIDExclude	OUT	SourceUserClassIDExclude			
	NewDestUserClassID	OUT	DestUserClassID			
	NewDestUserClassIDExclude	OUT	DestUserClassIDExclude			
	NewTCPACK	OUT	TCPACK			
	NewTCPACKExclude	OUT	TCPACKExclude			
	NewIPLengthMin	OUT	IPLengthMin			
	NewIPLengthMax	OUT	IPLengthMax			
	NewIPLengthExclude	OUT	IPLengthExclude			
	NewDSCPCheck	OUT	DSCPCheck			
	NewDSCPExclude	OUT	DSCPExclude			
	NewDSCPMark	OUT	DSCPMark			
	NewEthernetPriorityCheck	OUT	EthernetPriorityCheck			
	NewEthernetProirityExclude	OUT	EthernetProirityExclude			
	NewEthernetPriorityMark	OUT	EthernetPriorityMark			
	NewVLANIDCheck	OUT	VLANIDCheck			
	NewVLANIDExclude	OUT	VLANIDExclude			
	NewForwardingPolicy	OUT	ForwardingPolicy			
	NewClassPolicer	OUT	ClassPolicer			
	NewClassQueue	OUT	ClassQueue			
	NewClassApp	OUT	ClassApp			
SetClassificationEntryEnable	NewClassificationKey	IN	ClassificationKey	Sets the value of the Enable state variable to enable or disable a particular Classification entry.	402, 501, 702	R S
	NewClassificationEnable	IN	ClassificationEnable			
SetClassificationEntryOrder	NewClassificationKey	IN	ClassificationKey	Sets the value of the ClassificationOrder state variable to re-arrange the order of precedence.	402, 501, 702	R S
	NewClassificationOrder	IN	ClassificationOrder			
App Table Actions						
AddAppEntry	NewAppEnable	IN	AppEnable	Inserts an entry in the App table.	402, 501, 701	R S
	NewProtocolIdentifier	IN	ProtocolIdentifier			
	NewAppName	IN	AppName			
	NewAppDefaultForwardingPolicy	IN	AppDefaultForwardingPolicy			
	NewAppDefaultPolicer	IN	AppDefaultPolicer			

	NewAppDefaultQueue	IN	AppDefaultQueue			
	NewAppDefaultDSCPMark	IN	AppDefaultDSCPMark			
	NewAppDefaultEthernetPriorityMark	IN	AppDefaultEthernetPriorityMark			
	NewAppKey	OUT	AppKey			
DeleteAppEntry	NewAppKey	IN	AppKey	Deletes an entry in the App table.	402, 501, 702	R S
GetSpecific AppEntry	NewAppKey	IN	AppKey	Retrieve an entry in the App table.	402, 501, 714	R
	NewAppEnable	OUT	AppEnable			
	NewAppStatus	OUT	AppStatus			
	NewProtocolIdentifier	OUT	ProtocolIdentifier			
	NewAppName	OUT	AppName			
	NewAppDefaultForwardingPolicy	OUT	AppDefaultForwardingPolicy			
	NewAppDefaultPolicer	OUT	AppDefaultPolicer			
	NewAppDefaultQueue	OUT	AppDefaultQueue			
	NewAppDefaultDSCPMark	OUT	AppDefaultDSCPMark			
	NewAppDefaultEthernetPriorityMark	OUT	AppDefaultEthernetPriorityMark			
GetGeneric AppEntry	NewAppIndex	IN	AppNumberOfEntries	Retrieves the App table one entry at a time. Control points can call this action with an incrementing array index until no more entries are found on the gateway. Index is ranging from 0 to App-NumberOfEntries-1.	402, 501, 713	R
	NewAppKey	OUT	AppKey			
	NewAppEnable	OUT	AppEnable			
	NewAppStatus	OUT	AppStatus			
	NewProtocolIdentifier	OUT	ProtocolIdentifier			
	NewAppName	OUT	AppName			
	NewAppDefaultForwardingPolicy	OUT	AppDefaultForwardingPolicy			
	NewAppDefaultPolicer	OUT	AppDefaultPolicer			
	NewAppDefaultQueue	OUT	AppDefaultQueue			
	NewAppDefaultDSCPMark	OUT	AppDefaultDSCPMark			
	NewAppDefaultEthernetPriorityMark	OUT	AppDefaultEthernetPriorityMark			
SetAppEntryEnable	NewAppKey	IN	AppKey	Sets the value of the Enable state variable to enable or disable a particular App entry.	402, 501, 702	R S
	NewAppEnable	IN	AppEnable			
Flow Table Actions						
AddFlowEntry	NewFlowEnable	IN	FlowEnable	Inserts an entry in the Flow table.	402, 501, 701	R S
	NewFlowType	IN	FlowType			
	NewFlowTypeParameters	IN	FlowTypeParameters			
	NewFlowName	IN	FlowName			
	NewAppIdentifier	IN	AppIdentifier			
	NewFlowForwardingPolicy	IN	FlowForwardingPolicy			
	NewFlowPolicer	IN	FlowPolicer			
	NewFlowQueue	IN	FlowQueue			
	NewFlowDSCPMark	IN	FlowDSCPMark			
	NewFlowEthernetPriorityMark	IN	FlowEthernetPriorityMark			
	NewFlowKey	OUT	FlowKey			
DeleteFlowEntry	NewFlowKey	IN	FlowKey	Deletes an entry in the Flow table.	402, 501, 702	R S
GetSpecific FlowEntry	NewFlowKey	IN	FlowKey	Retrieve an entry in the Flow table.	402, 501, 714	R
	NewFlowEnable	OUT	FlowEnable			
	NewFlowStatus	OUT	FlowStatus			
	NewFlowKey	OUT	FlowKey			
	NewFlowType	OUT	FlowType			
	NewFlowTypeParameters	OUT	FlowTypeParameters			
	NewFlowName	OUT	FlowName			
	NewAppIdentifier	OUT	AppIdentifier			
	NewFlowForwardingPolicy	OUT	FlowForwardingPolicy			
	NewFlowPolicer	OUT	FlowPolicer			
	NewFlowQueue	OUT	FlowQueue			
	NewFlowDSCPMark	OUT	FlowDSCPMark			
	NewFlowEthernetPriorityMark	OUT	FlowEthernetPriorityMark			
GetGeneric FlowEntry	NewFlowIndex	IN	FlowNumberOfEntries	Retrieves the Flow table one entry at a time. Control points can call this action with an incrementing array index until no more entries are found on the gateway. Index is ranging from 0 to FlowNumberOfEntries-1.	402, 501, 713	R
	NewFlowKey	OUT	FlowKey			
	NewFlowEnable	OUT	FlowEnable			
	NewFlowStatus	OUT	FlowStatus			
	NewFlowIndex	OUT	FlowNumberOfEntries			
	NewFlowType	OUT	FlowType			
	NewFlowTypeParameters	OUT	FlowTypeParameters			
	NewFlowName	OUT	FlowName			
	NewAppIdentifier	OUT	AppIdentifier			
	NewFlowForwardingPolicy	OUT	FlowForwardingPolicy			
	NewFlowPolicer	OUT	FlowPolicer			
	NewFlowQueue	OUT	FlowQueue			
	NewFlowDSCPMark	OUT	FlowDSCPMark			
	NewFlowEthernetPriorityMark	OUT	FlowEthernetPriorityMark			
SetFlowEntryEnable	NewFlowKey	IN	FlowKey	Sets the value of the Enable state variable to enable or disable a particular Flow entry.	402, 501, 702	R S
	NewFlowEnable	IN	FlowEnable			
Policer Table Actions						
AddPolicerEntry	NewPolicerEnable	IN	PolicerEnable	Inserts an entry in the Policer table.	402, 501, 701	R S
	NewCommittedRate	IN	CommittedRate			
	NewCommittedBurstSize	IN	CommittedBurstSize			
	NewExcessBurstSize	IN	ExcessBurstSize			
	NewPeakRate	IN	PeakRate			

	NewPeakBurstSize	IN	PeakBurstSize			
	NewMeterType	IN	MeterType			
	NewConformingAction	IN	ConformingAction			
	NewPartialConformingAction	IN	PartialConformingAction			
	NewNonConformingAction	IN	NonConformingAction			
	NewPolicerKey	OUT	PolicerKey			
DeletePolicerEntry	NewPolicerKey	IN	PolicerKey	Deletes an entry in the Policer table.	402, 501, 702	R S
GetSpecific PolicerEntry	NewPolicerKey	IN	PolicerKey	Retrieve an entry in the Policer table.	402, 501, 714	R
	NewPolicerEnable	OUT	PolicerEnable			
	NewPolicerStatus	OUT	PolicerStatus			
	NewCommittedRate	OUT	CommittedRate			
	NewCommittedBurstSize	OUT	CommittedBurstSize			
	NewExcessBurstSize	OUT	ExcessBurstSize			
	NewPeakRate	OUT	PeakRate			
	NewPeakBurstSize	OUT	PeakBurstSize			
	NewMeterType	OUT	MeterType			
	NewPossibleMeterTypes	OUT	PossibleMeterTypes			
	NewConformingAction	OUT	ConformingAction			
	NewPartialConformingAction	OUT	PartialConformingAction			
	NewNonConformingAction	OUT	NonConformingAction			
	NewCountedPackets	OUT	CountedPackets			
	NewCountedBytes	OUT	CountedBytes			
GetGeneric PolicerEntry	NewPolicerIndex	IN	PolicerNumberOfEntries	Retrieves the Policer table one entry at a time. Control points can call this action with an incrementing array index until no more entries are found on the gateway. Index is ranging from 0 to PolicerNumberOfEntries-1.	402, 501, 713	R
	NewPolicerKey	OUT	PolicerKey			
	NewPolicerEnable	OUT	PolicerEnable			
	NewPolicerStatus	OUT	PolicerStatus			
	NewCommittedRate	OUT	CommittedRate			
	NewCommittedBurstSize	OUT	CommittedBurstSize			
	NewExcessBurstSize	OUT	ExcessBurstSize			
	NewPeakRate	OUT	PeakRate			
	NewPeakBurstSize	OUT	PeakBurstSize			
	NewMeterType	OUT	MeterType			
	NewPossibleMeterTypes	OUT	PossibleMeterTypes			
	NewConformingAction	OUT	ConformingAction			
	NewPartialConformingAction	OUT	PartialConformingAction			
	NewNonConformingAction	OUT	NonConformingAction			
	NewCountedPackets	OUT	CountedPackets			
	NewCountedBytes	OUT	CountedBytes			
SetPolicerEntryEnable	NewPolicerKey	IN	PolicerKey	Sets the value of the Enable state variable to enable or disable a particular Policer entry.	402, 501, 702	R S
	NewPolicerEnable	IN	PolicerEnable			
Queue Table Actions						
AddQueueEntry	NewQueueEnable	IN	QueueEnable	Inserts an entry in the Queue table.	402, 501, 701	R S
	NewQueueInterface	IN	QueueInterface			
	NewQueueWeight	IN	QueueWeight			
	NewQueuePrecedence	IN	QueuePrecedence			
	NewREDThreshold	IN	REDThreshold			
	NewREDPercentage	IN	REDPercentage			
	NewDropAlgorithm	IN	DropAlgorithm			
	NewSchedulerAlgorithm	IN	SchedulerAlgorithm			
	NewShapingRate	IN	ShapingRate			
	NewShapingBurstSize	IN	ShapingBurstSize			
	NewQueueKey	OUT	QueueKey			
DeleteQueueEntry	NewQueueKey	IN	QueueKey	Deletes an entry in the Queue table.	402, 501, 702	R S
GetSpecific QueueEntry	NewQueueKey	IN	QueueKey	Retrieve an entry in the Queue table.	402, 501, 714	R
	NewQueueEnable	OUT	QueueEnable			
	NewQueueStatus	OUT	QueueStatus			
	NewQueueInterface	OUT	QueueInterface			
	NewQueueBufferLength	OUT	QueueBufferLength			
	NewQueueWeight	OUT	QueueWeight			
	NewQueuePrecedence	OUT	QueuePrecedence			
	NewREDThreshold	OUT	REDThreshold			
	NewREDPercentage	OUT	REDPercentage			
	NewDropAlgorithm	OUT	DropAlgorithm			
	NewSchedulerAlgorithm	OUT	SchedulerAlgorithm			
	NewShapingRate	OUT	ShapingRate			
	NewShapingBurstSize	OUT	ShapingBurstSize			
GetGeneric QueueEntry	NewQueueIndex	IN	QueueNumberOfEntries	Retrieves the Queue table one entry at a time. Control points can call this action with an incrementing array index until no more entries are found on the gateway. Index is ranging from 0 to QueueNumberOfEntries-1.	402, 501, 713	R
	NewQueueKey	OUT	QueueKey			
	NewQueueEnable	OUT	QueueEnable			
	NewQueueStatus	OUT	QueueStatus			
	NewQueueInterface	OUT	QueueInterface			
	NewQueueBufferLength	OUT	QueueBufferLength			
	NewQueueWeight	OUT	QueueWeight			
	NewQueuePrecedence	OUT	QueuePrecedence			
	NewREDThreshold	OUT	REDThreshold			
	NewREDPercentage	OUT	REDPercentage			
	NewDropAlgorithm	OUT	DropAlgorithm			
	NewSchedulerAlgorithm	OUT	SchedulerAlgorithm			
	NewShapingRate	OUT	ShapingRate			

	NewShapingBurstSize	OUT	ShapingBurstSize			
SetQueueEntryEnable	NewQueueKey NewQueueEnable	IN IN	QueueKey QueueEnable	Sets the value of the Enable state variable to enable or disable a particular Queue entry.	402, 501, 702	R S

End Inserted Text

2.4 Addition of Layer2Bridging Service

This section specifies a new service to be inserted into TR-064. This service specifies layer-2 bridges between LAN and/or WAN interfaces. Bridges can be defined to include layer-2 filter criteria to selectively bridge traffic between interfaces.

Begin Inserted Text

6.5.x Layer2Bridging

6.5.x.1 Overview

This service specifies bridges between layer-2 LAN and/or WAN interfaces. Bridges can be defined to include layer-2 filter criteria to selectively bridge traffic between interfaces. A description of the bridging model associated with this service is provided in Appendix A.

6.5.x.2 Service Modelling Definitions

ServiceType

urn:dslforum-org:service:Layer2Bridging:1

StateVariables

Variable Name	Table	Data Type	Allowed Value	Description	Default Value	R/O
MaxBridgeEntries	-	ui2	>=0	The maximum number of entries available in the Bridge table.	N/A	R
MaxFilterEntries	-	ui2	>=0	The maximum number of entries available in the Filter table.	N/A	R
MaxMarkingEntries	-	ui2	>=0	The maximum number of entries available in the Marking table.	N/A	R
BridgeNumberOfEntries	Bridge (index)	ui2	>=0	Number of entries in the Bridge table.	N/A	R
BridgeKey	Bridge KEY	ui2	>=0	Unique key for each Bridge table entry.	N/A	R
BridgeEnable	Bridge	Boolean	1, 0	Enables or disables this Bridge table entry.	0	R
BridgeStatus	Bridge	String	Disabled Enabled Error	Indicates the status of this Bridge table entry.	"Disabled"	R
BridgeName	Bridge	String	Max length = 64 characters	Human-readable name for this Bridge table entry.	<Empty>	R

VLANID	Bridge	ui2	>=0 < 4095	The 802.1Q VLAN ID associated with this Bridge. A value of 0 indicates either Untagged or PriorityOnly tagging, which are treated identically.	0	R
FilterNumberOfEntries	Filter (index)	ui2	>=0	Number of entries in the Filter table.	N/A	R
FilterKey	Filter KEY	ui2	>=0	Unique key for each Filter table entry.	N/A	R
FilterEnable	Filter	Boolean	1, 0	Enables or disables this Filter table entry.	0	R
FilterStatus	Filter	String	Disabled Enabled Error	Indicates the status of this Filter table entry.	"Disabled"	R
FilterBridgeReference	Filter	i2	>=-1	The BridgeKey value of the Bridge table entry associated with this Filter. A value of -1 indicates the Filter table entry is not associated with a Bridge (and has no effect).	-1	R
ExclusivityOrder	Filter	ui2	>=0	Whether or not the Filter definition is exclusive of all others. And if the entry is exclusive, order of precedence. A value of 1 or greater indicates an Exclusive Filter, where the value 1 indicates the first entry to be considered (highest precedence). A value of 0 indicates a Non-Exclusive Filter. For each packet, if the packet matches any Exclusive Filters, the packet is assigned to the Bridge associated with the highest precedence Exclusive Filter to which it matches (lowest ExclusivityOrder value). If and only if the packet does not match any Exclusive Filters, the packet is assigned to all Bridges associated with each Non-Exclusive Filter for which it matches the defining criteria. If a packet matches no Filter, it is discarded. When the ExclusivityOrder is set to match that of an existing Exclusive Filter (1 or greater), the value for the existing entry and all higher numbered entries is incremented (lowered in precedence) to ensure uniqueness of this value. A deletion or change in ExclusivityOrder of an Exclusive Filter causes ExclusivityOrder values of other Exclusive Filters (values 1 or greater) to be compacted. Note that the use of Exclusive Filters to associate a layer-3 router interface with LAN and/or WAN interfaces via a Bridge entry overrides the default association between layer-3 and layer-2 objects implied by the Internet-GatewayDevice object hierarchy.	0	R
FilterInterface	Filter	String		The interface or interfaces associated with this Filter table entry. The bridge corresponding to this Filter table entry is defined to admit packets on ingress to the bridge from the specified interfaces that meet all of the criteria specified in the Filter table entry. The following values are defined. To associate this Filter with a single interface listed in the Available-Interface table, the FilterInterface value is set to the value of corresponding AvailableInterfaceKey. "AllInterfaces" indicates that this	<Empty>	R

				Filter is associated with all LAN and WAN interfaces listed in the AvailableInterface table (all entries of InterfaceType LANInterface or WANInterface). "LANInterfaces" indicates that this Filter is associated with all LAN interfaces listed in the Available-Interface table (all entries of InterfaceType LANInterface). "WANInterfaces" indicates that this Filter is associated with all WAN interfaces listed in the Available-Interface table (all entries of InterfaceType WANInterface). An empty string indicates the Filter table entry is not associated with any interface (and has no effect)		
VLANIDFilter	Filter	i2	>=-1 < 4095	The 802.1Q VLAN ID of packets to admit to the specified Bridge through the interfaces specified for this Filter. A value of -1 indicates that the default VLAN ID for the Bridge should be used instead (as specified by the VLANID variable in the Bridge table entry associated with this Filter table entry).	-1	R
AdmitOnlyVLANTagged	Filter	Boolean	1, 0	If true, on ingress to the interfaces associated with this Filter, the Bridge admits only packets tagged with a VLAN ID that matches the VLANIDFilter parameter (or instead, the VLAN ID for the Bridge if VLANIDFilter is unspecified). If false, on ingress to the interfaces associated with this Filter, the Bridge admits both packets tagged with a VLAN ID that matches the VLANIDFilter parameter (or instead, the VLAN ID for the Bridge if VLANIDFilter is unspecified), and any Untagged or PriorityOnly packets. All Untagged or PriorityOnly packets are tagged on ingress with the value of the VLANID parameter. If the VLANIDFilter parameter (or instead, the VLAN ID for the Bridge if VLANIDFilter is unspecified) is equal to 0, then this parameter is ignored, and only packets that are Untagged or PriorityOnly packets are admitted.	0	R
EthertypeFilterList	Filter	String	Max length = 256 characters	Comma-separated list of unsigned integers, each representing an Ethertype value.	<Empty>	R
EthertypeFilterExclude	Filter	Boolean	1, 0	If false, on ingress to the interfaces associated with this Filter, the Bridge is defined to admit only those packets that match one of the EthertypeFilter-List entries (in either the Ethernet or SNAP Type header). If the Ethertype-FilterList is empty, no packets are admitted. If true, on ingress to the interfaces associated with this Filter, the Bridge is defined to admit all packets except those packets that match one of the -EthertypeFilterList entries (in either the Ethernet or SNAP Type header). If the EthertypeFilterList is empty, packets are admitted regardless of Ethertype.	1	R
SourceMACAddressFilterList	Filter	String	Max length = 512 characters	Comma-separated list of MAC Addresses. Each list entry may optionally specify a bit-mask, where matching of a packet's MAC address is only to be done for bit positions set to one in the mask. If no mask is specified, all bits of the MAC Address are to be used for matching.	<Empty>	R

				For example, the list may be: 01:02:03:04:05:06, 11:22:33:00:00:00/FF:FF:FF:00:00:00, 88:77:66:55:44:33		
SourceMACAddressFilterExclude	Filter	Boolean	1, 0	If false, on ingress to the interfaces associated with this Filter, the Bridge admits only those packets whose source MAC Address matches one of the SourceMACAddressFilterList entries. If the SourceMACAddress-FilterList is empty, no packets are admitted. If true, on ingress to the interfaces associated with this Filter, the Bridge admits all packets except those packets whose source MAC Address matches one of the SourceMACAddressFilter-List entries. If the SourceMAC-AddressFilterList is empty, packets are admitted regardless of MAC address.	1	R
DestMACAddressFilterList	Filter	String	Max length = 512 characters	Comma-separated list of MAC Addresses. Each list entry may optionally specify a bit-mask, where matching of a packet's MAC address is only to be done for bit positions set to one in the mask. If no mask is specified, all bits of the MAC Address are to be used for matching. For example, list may be: 01:02:03:04:05:06, 11:22:33:00:00:00/FF:FF:FF:00:00:00, 88:77:66:55:44:33	<Empty>	R
DestMACAddressFilterExclude	Filter	Boolean	1, 0	If false, on ingress to the interfaces associated with this Filter, the Bridge admits only those packets whose destination MAC Address matches one of the DestMACAddressFilterList entries. If the DestMACAddress-FilterList is empty, no packets are admitted. If true, on ingress to the interfaces associated with this Filter, the Bridge admits all packets except those packets whose destination MAC Address matches one of the DestMACAddress-FilterList entries. If the DestMAC-AddressFilterList is empty, packets are admitted regardless of MAC address.	1	R
SourceMACFromVendorClassIDFilter	Filter	String	Max length = 256 characters	A string used to identify one or more devices via DHCP for which MAC address filtering would subsequently apply. A device is considered matching if it its DHCP Vendor Class Identifier (Option 60 as defined in RFC 2132) in the most recent DHCP lease acquisition or renewal was equal to the specified value.	<Empty>	O
SourceMACFromVendorClassIDFilterExclude	Filter	Boolean	1, 0	If false, on ingress to the interfaces associated with this Filter, the Bridge admits only those packets whose source MAC Address matches that of a LAN device previously identified as described in SourceMACFrom-VendorClassIDFilter. If SourceMAC-FromVendorClassIDFilter is empty, no packets are admitted. If true, on ingress to the interfaces associated with this Filter, the Bridge admits all packets except those packets whose source MAC Address matches that of a LAN device previously identified as described in Source-MACFromVendorClassIDFilter. If the SourceMACFromVendorClassID-Filter is empty, packets are admitted regardless of MAC address.	1	O
DestMACFromVendorClassIDFilter	Filter	String	Max length = 256 characters	A string used to identify one or more devices via DHCP for which MAC address filtering would subsequently apply. A device is considered	<Empty>	O

				matching if it its DHCP Vendor Class Identifier (Option 60 as defined in RFC 2132) in the most recent DHCP lease acquisition or renewal was equal to the specified value.		
DestMACFromVendorClassIDFilterExclude	Filter	Boolean	1, 0	If false, on ingress to the interfaces associated with this Filter, the Bridge admits only those packets whose destination MAC Address matches that of a LAN device previously identified as described in DestMAC-FromVendorClassIDFilter. If Dest-MACFromVendorClassIDFilter is empty, no packets are admitted. If true, on ingress to the interfaces associated with this Filter, the Bridge admits all packets except those packets whose destination MAC Address matches that of a LAN device previously identified as described in DestMACFromVendorClassIDFilter. If the DestMACFromVendorClassID-Filter is empty, packets are admitted regardless of MAC address.	1	O
SourceMACFromClientIDFilter	Filter	String	Max length = 256 characters	A string used to identify one or more devices via DHCP for which MAC address filtering would subsequently apply. A device is considered matching if it its DHCP Client Identifier (Option 61 as defined in RFC 2132) in the most recent DHCP lease acquisition or renewal was equal to the specified value.	<Empty>	O
SourceMACFromClientIDFilterExclude	Filter	Boolean	1, 0	If false, on ingress to the interfaces associated with this Filter, the Bridge admits only those packets whose source MAC Address matches that of a LAN device previously identified as described in SourceMACFrom-ClientIDFilter. If SourceMACFrom-ClientIDFilter is empty, no packets are admitted. If true, on ingress to the interfaces associated with this Filter, the Bridge admits all packets except those packets whose source MAC Address matches that of a LAN device previously identified as described in Source-MACFromClientIDFilter. If the SourceMACFromClientIDFilter is empty, packets are admitted regardless of MAC address.	1	O
DestMACFromClientIDFilter	Filter	String	Max length = 256 characters	A string used to identify one or more devices via DHCP for which MAC address filtering would subsequently apply. A device is considered matching if it its DHCP Client Identifier (Option 61 as defined in RFC 2132) in the most recent DHCP lease acquisition or renewal was equal to the specified value.	<Empty>	O
DestMACFromClientIDFilterExclude	Filter	Boolean	1, 0	If false, on ingress to the interfaces associated with this Filter, the Bridge admits only those packets whose destination MAC Address matches that of a LAN device previously identified as described in DestMAC-FromClientIDFilter. If DestMAC-FromClientIDFilter is empty, no packets are admitted. If true, on ingress to the interfaces associated with this Filter, the Bridge admits all packets except those packets whose destination MAC Address matches that of a LAN device previously identified as described in DestMACFromClientIDFilter. If the DestMACFromClientIDFilter is empty, packets are admitted regardless of MAC address.	1	O
SourceMACFromUserClassIDFilter	Filter	String	Max length = 256 characters	A string used to identify one or more devices via DHCP for which MAC address filtering would subsequently apply. A device is considered matching if it its DHCP User Class	<Empty>	O

				Identifier (Option 77 as defined in RFC 3004) in the most recent DHCP lease acquisition or renewal was equal to the specified value.		
SourceMACFromUserClassIDFilterExclude	Filter	Boolean	1, 0	If false, on ingress to the interfaces associated with this Filter, the Bridge admits only those packets whose source MAC Address matches that of a LAN device previously identified as described in SourceMACFromUser-ClassIDFilter. If SourceMACFrom-UserClassIDFilter is empty, no packets are admitted. If true, on ingress to the interfaces associated with this Filter, the Bridge admits all packets except those packets whose source MAC Address matches that of a LAN device previously identified as described in Source-MACFromUserClassIDFilter. If the SourceMACFromUserClassIDFilter is empty, packets are admitted regardless of MAC address.	1	O
DestMACFromUserClassIDFilter	Filter	String	Max length = 256 characters	A string used to identify one or more devices via DHCP for which MAC address filtering would subsequently apply. A device is considered matching if it its DHCP User Class Identifier (Option 77 as defined in RFC 3004) in the most recent DHCP lease acquisition or renewal was equal to the specified value.	<Empty>	O
DestMACFromUserClassIDFilterExclude	Filter	Boolean	1, 0	If false, on ingress to the interfaces associated with this Filter, the Bridge admits only those packets whose destination MAC Address matches that of a LAN device previously identified as described in DestMAC-FromUserClassIDFilter. If DestMAC-FromUserClassIDFilter is empty, no packets are admitted. If true, on ingress to the interfaces associated with this Filter, the Bridge admits all packets except those packets whose destination MAC Address matches that of a LAN device previously identified as described in DestMACFromUserClassIDFilter. If the DestMACFromUserClassIDFilter is empty, packets are admitted regardless of MAC address.	1	O
MarkingNumberOfEntries	Marking (index)	ui2	>=0	Number of entries in the Marking table.	N/A	R
MarkingKey	Marking KEY	ui2	>=0	Unique key for each Marking table entry.	N/A	R
MarkingEnable	Marking	Boolean	1, 0	Enables or disables this Marking table entry.	0	R
MarkingStatus	Marking	String	Disabled Enabled Error	Indicates the status of this Marking table entry.	“Disabled”	R
MarkingBridgeReference	Marking	i2	>=-1	The BridgeKey value of the Bridge table entry associated with this Marking table entry. A value of □1 indicates the Marking table entry is not associated with a Bridge (and has no effect). The effect of a Marking table entry applies only to packets that have been admitted to the specified bridge (regardless of the ingress interface).	-1	R
MarkingInterface	Marking	String		The interface or interfaces associated with this Marking table entry for which the specified marking behavior is to apply on egress from the associated bridge. The following values are defined. To associate this Marking table entry with a single interface listed in the AvailableInterface table, the	<Empty>	R

				MarkingInterface value is set to the value of corresponding AvailableInterfaceKey. "AllInterfaces" indicates that this Marking table entry is associated with all LAN and WAN interfaces listed in the AvailableInterface table (all entries of InterfaceType LANInterface or WANInterface). "LANInterfaces" indicates that this Marking table entry is associated with all LAN interfaces listed in the AvailableInterface table (all entries of InterfaceType LANInterface). "WANInterfaces" indicates that this Marking table entry is associated with all WAN interfaces listed in the AvailableInterface table (all entries of InterfaceType WANInterface). An empty string indicates the Marking table entry table entry is not associated with any interface (and has no effect) If there is more than one enabled Marking table entry that specifies one or more of the same interfaces for the same bridge (identical values of MarkingBridgeReference), then for packets on egress from the specified bridge to those interfaces, the applied marking MUST be that specified in the Marking table entry among those in conflict with the lowest MarkingKey value. If an interface in a given bridge does not have a corresponding Marking table entry, the marking is left unchanged on egress.		
VLANIDUntag	Marking	Boolean	1, 0	If true, on egress to the interfaces associated with this Marking table entry, all packets are Untagged. That is, the VLAN tag is stripped from the packet. If false, on egress to the interfaces associated with this Marking table entry, all VLAN tags are left intact (including those added on ingress).	0	R
VLANIDMark	Marking	i2	>=-1 < 4095	The 802.1Q VLAN ID to be used on egress to the interfaces associated with this Marking table entry (if VLANIDUntag is false). A value of -1 indicates that the default VLAN ID for the Bridge should be used instead (as specified by the VLANID variable in the Bridge table entry associated with this Marking table entry).	-1	R
EthernetPriorityMark	Marking	i2	>=-1	Ethernet priority code (as defined in 802.1D) to mark traffic with that falls into this Bridge on egress to the interfaces associated with this Marking table entry. A value of -1 indicates no change from the incoming packet or the mark assigned by the classifier.	-1	R
EthernetPriorityOverride	Marking	Boolean	1, 0	If false, on egress to the interfaces associated with this Marking table entry, the EthernetPriorityMark, if specified, is applied only to packets of priority 0. If true, on egress to the interfaces associated with this Marking table entry, the EthernetPriorityMark, if specified, is to be applied to all packets on this Bridge. If VLANIDUntag is true, then no priority marking is done since the tag containing the Ethernet priority is removed.	0	R

AvailableInterfaceNumberOfEntries	AvailableInterface (index)	ui2	>=0	Number of entries in the Interface table.	N/A	R
AvailableInterfaceKey	AvailableInterface KEY	ui2	>=0	Unique key for each Interface entry.	N/A	R
InterfaceType	AvailableInterface	string	LANInterface WANInterface LANRouterConnection WANRouterConnection	Whether the interface is a LAN-side or WAN-side interface, or a LAN-side or WAN-side connection to the Gateway's IP router.	N/A	R
InterfaceReference	AvailableInterface	string	Max length = 256 characters	This table should contain a single entry for each *available* LAN and WAN interface. For a WAN interface, this parameter is a 2-tuple referring to a particular WANConnectionDevice. A WAN-ConnectionDevice is considered available (included in this table) *only* if it supports layer-2 bridged traffic. That is, this table MUST include only WANConnectionDevices that contain either a WANEthernetLinkConfig object, or that contain a WANDSL-LinkConfig object for which the LinkType is "EoA". For a LAN interface, a 2-tuple referring to a particular LAN**-InterfaceConfig object, or a WLAN-Configuration object. This table SHOULD include one entry for each such object. For a WAN-side connection to the Gateway's IP router, a 2-tuple referring to a particular WAN**Connection service. This table SHOULD include an entry for each layer-3 WAN connection. For a LAN-side connection to the Gateway's IP router, a 2-tuple referring to a particular LANDevice. This table SHOULD include an entry for each LANDevice, each of which is associated with a LAN-side layer-3 connection to the Gateway's IP router.	N/A	R

Actions, Arguments & Errors

Name	Argument	Dir	Related State Variable(s)	Description	Errors	R/O
GetInfo	NewMaxBridgeEntries NewMaxFilterEntries NewMaxMarkingEntries NewBridgeNumberOfEntries NewFilterNumberOfEntries NewMarkingNumberOfEntries NewAvailableInterfaceNumberOfEntries	OUT OUT OUT OUT OUT OUT OUT	MaxBridgeEntries MaxFilterEntries MaxMarkingEntries BridgeNumberOfEntries FilterNumerOfEntries MarkingNumberOfEntries AvailableInterfaceNumberOfEntries	Retrieves all of the state variables not associated with a table.	402, 501	R
Bridge Table Actions						
AddBridgeEntry	NewBridgeEnable NewBridgeName NewVLANID NewBridgeKey	IN IN IN OUT	BridgeEnable BridgeName VLANID BridgeKey	Inserts an entry in the Bridge table.	402, 501, 701	R S
DeleteBridgeEntry	NewBridgeKey	IN	BridgeKey	Deletes an entry in the Bridge table.	402, 501, 702	R S
GetSpecific BridgeEntry	NewBridgeKey NewBridgeEnable NewBridgeStatus NewVLANID	IN OUT OUT OUT	BridgeKey BridgeEnable BridgeStatus VLANID	Retrieve an entry in the Bridge table.	402, 501, 714	R
GetGeneric BridgeEntry	NewBridgeIndex NewBridgeKey NewBridgeEnable NewBridgeStatus NewVLANID	IN OUT OUT OUT OUT	BridgeNumberOfEntries BridgeKey BridgeEnable BridgeStatus VLANID	Retrieves the Bridge table one entry at a time. Control points can call this action with an incrementing array index until no more entries are found on the gateway. Index is ranging from 0 to BridgeNumberOfEntries-1.	402, 501, 713	R

SetBridgeEntryEnable	NewBridgeKey NewBridgeEnable	IN IN	BridgeKey BridgeEnable	Sets the value of the Enable state variable to enable or disable a particular Bridge entry.	402, 501, 702	R S
Filter Table Actions						
AddFilterEntry	NewFilterEnable NewFilterBridgeReference NewExclusivityOrder NewFilterInterface NewVLANIDFilter NewAdmitOnlyVLANTagged NewEthertypeFilterList NewEthertypeFilterExclude NewSourceMACAddressFilterList NewSourceMACAddressFilterExclude NewDestMACAddressFilterList NewDestMACAddressFilterExclude NewSourceMACFromVendorClassIDFilter NewSourceMACFromVendorClassIDFilterExclude NewDestMACFromVendorClassIDFilter NewDestMACFromVendorClassIDFilterExclude NewSourceMACFromClientIDFilter NewSourceMACFromClientIDFilterExclude NewDestMACFromClientIDFilter NewDestMACFromClientIDFilterExclude NewSourceMACFromUserClassIDFilter NewSourceMACFromUserClassIDFilterExclude NewDestMACFromUserClassIDFilter NewDestMACFromUserClassIDFilterExclude NewFilterKey	IN IN IN IN IN IN IN IN IN IN IN IN IN IN IN IN IN IN IN IN IN IN IN IN OUT	FilterEnable FilterBridgeReference ExclusivityOrder FilterInterface VLANIDFilter AdmitOnlyVLANTagged EthertypeFilterList EthertypeFilterExclude SourceMACAddressFilterList SourceMACAddressFilterExclude DestMACAddressFilterList DestMACAddressFilterExclude SourceMACFromVendorClassIDFilter SourceMACFromVendorClassIDFilterExclude DestMACFromVendorClassIDFilter DestMACFromVendorClassIDFilterExclude SourceMACFromClientIDFilter SourceMACFromClientIDFilterExclude DestMACFromClientIDFilter DestMACFromClientIDFilterExclude SourceMACFromUserClassIDFilter SourceMACFromUserClassIDFilterExclude DestMACFromUserClassIDFilter DestMACFromUserClassIDFilterExclude FilterKey	Inserts an entry in the Filter table.	402, 501, 701	R S
DeleteFilterEntry	NewFilterKey	IN	FilterKey	Deletes an entry in the Filter table.	402, 501, 702	R S
GetSpecific FilterEntry	NewFilterKey NewFilterEnable NewFilterStatus NewFilterBridgeReference NewExclusivityOrder NewFilterInterface NewVLANIDFilter NewAdmitOnlyVLANTagged NewEthertypeFilterList NewEthertypeFilterExclude NewSourceMACAddressFilterList NewSourceMACAddressFilterExclude NewDestMACAddressFilterList NewDestMACAddressFilterExclude NewSourceMACFromVendorClassIDFilter NewSourceMACFromVendorClassIDFilterExclude NewDestMACFromVendorClassIDFilter NewDestMACFromVendorClassIDFilterExclude NewSourceMACFromClientIDFilter NewSourceMACFromClientIDFilterExclude NewDestMACFromClientIDFilter NewDestMACFromClientIDFilterExclude NewSourceMACFromUserClassIDFilter NewSourceMACFromUserClassIDFilterExclude NewDestMACFromUserClassIDFilter NewDestMACFromUserClassIDFilterExclude	IN OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT	FilterKey FilterEnable FilterStatus FilterBridgeReference ExclusivityOrder FilterInterface VLANIDFilter AdmitOnlyVLANTagged EthertypeFilterList EthertypeFilterExclude SourceMACAddressFilterList SourceMACAddressFilterExclude DestMACAddressFilterList DestMACAddressFilterExclude SourceMACFromVendorClassIDFilter SourceMACFromVendorClassIDFilterExclude DestMACFromVendorClassIDFilter DestMACFromVendorClassIDFilterExclude SourceMACFromClientIDFilter SourceMACFromClientIDFilterExclude DestMACFromClientIDFilter DestMACFromClientIDFilterExclude SourceMACFromUserClassIDFilter SourceMACFromUserClassIDFilterExclude DestMACFromUserClassIDFilter DestMACFromUserClassIDFilterExclude	Retrieve an entry in the Filter table.	402, 501, 714	R
GetGeneric FilterEntry	NewFilterIndex NewFilterKey NewFilterEnable NewFilterStatus NewFilterBridgeReference NewExclusivityOrder NewFilterInterface NewVLANIDFilter NewAdmitOnlyVLANTagged NewEthertypeFilterList NewEthertypeFilterExclude NewSourceMACAddressFilterList NewSourceMACAddressFilterExclude NewDestMACAddressFilterList NewDestMACAddressFilterExclude NewSourceMACFromVendorClassIDFilter NewSourceMACFromVendorClassIDFilterExclude NewDestMACFromVendorClassIDFilter NewDestMACFromVendorClassIDFilterExclude NewSourceMACFromClientIDFilter NewSourceMACFromClientIDFilterExclude NewDestMACFromClientIDFilter NewDestMACFromClientIDFilterExclude NewSourceMACFromUserClassIDFilter NewSourceMACFromUserClassIDFilterExclude NewDestMACFromUserClassIDFilter NewDestMACFromUserClassIDFilterExclude	IN OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT	FilterNumberOfEntries FilterKey FilterEnable FilterStatus FilterBridgeReference ExclusivityOrder FilterInterface VLANIDFilter AdmitOnlyVLANTagged EthertypeFilterList EthertypeFilterExclude SourceMACAddressFilterList SourceMACAddressFilterExclude DestMACAddressFilterList DestMACAddressFilterExclude SourceMACFromVendorClassIDFilter SourceMACFromVendorClassIDFilterExclude DestMACFromVendorClassIDFilter DestMACFromVendorClassIDFilterExclude SourceMACFromClientIDFilter SourceMACFromClientIDFilterExclude DestMACFromClientIDFilter DestMACFromClientIDFilterExclude SourceMACFromUserClassIDFilter SourceMACFromUserClassIDFilterExclude DestMACFromUserClassIDFilter DestMACFromUserClassIDFilterExclude	Retrieves the Filter table one entry at a time. Control points can call this action with an incrementing array index until no more entries are found on the gateway. Index is ranging from 0 to Filter-NumberOfEntries-1.	402, 501, 713	R

SetFilterEntryEnable	NewFilterKey NewFilterEnable	IN IN	FilterKey FilterEnable	Sets the value of the Enable state variable to enable or disable a particular Filter entry.	402, 501, 702	R S
SetFilterExclusivityOrder	NewFilterKey NewExclusivityOrder	IN IN	FilterKey ExclusivityOrder	Sets the value of the ExclusivityOrder state variable to re-arrange the order of precedence.	402, 501, 702	R S
Marking Table Actions						
AddMarkingEntry	NewMarkingEnable NewMarkingBridgeReference NewMarkingInterface NewVLANIDUntag NewVLANIDMark NewEthernetPriorityMark NewEthernetPriorityOverride NewMarkingKey	IN IN IN IN IN IN IN OUT	MarkingEnable MarkingBridgeReference MarkingInterface VLANIDUntag VLANIDMark EthernetPriorityMark EthernetPriorityOverride MarkingKey	Inserts an entry in the Marking table.	402, 501, 701	R S
DeleteMarkingEntry	NewMarkingKey	IN	MarkingKey	Deletes an entry in the Marking table.	402, 501, 702	R S
GetSpecific MarkingEntry	NewMarkingKey NewMarkingEnable NewMarkingStatus NewMarkingBridgeReference NewMarkingInterface NewVLANIDUntag NewVLANIDMark NewEthernetPriorityMark NewEthernetPriorityOverride	IN OUT OUT OUT OUT OUT OUT OUT OUT	MarkingKey MarkingEnable MarkingStatus MarkingBridgeReference MarkingInterface VLANIDUntag VLANIDMark EthernetPriorityMark EthernetPriorityOverride	Retrieve an entry in the Marking table.	402, 501, 714	R
GetGeneric MarkingEntry	NewMarkingIndex NewMarkingKey NewMarkingEnable NewMarkingStatus NewMarkingBridgeReference NewMarkingInterface NewVLANIDUntag NewVLANIDMark NewEthernetPriorityMark NewEthernetPriorityOverride	IN OUT OUT OUT OUT OUT OUT OUT OUT OUT	MarkingIndex MarkingKey MarkingEnable MarkingStatus MarkingBridgeReference MarkingInterface VLANIDUntag VLANIDMark EthernetPriorityMark NewEthernetPriorityOverride	Retrieves the Marking table one entry at a time. Control points can call this action with an incrementing array index until no more entries are found on the gateway. Index is ranging from 0 to MarkingNumberOfEntries-1.	402, 501, 713	R
SetFilterEntryEnable	NewFilterKey NewFilterEnable	IN IN	FilterKey FilterEnable	Sets the value of the Enable state variable to enable or disable a particular Filter entry.	402, 501, 702	R S
SetFilterExclusivityOrder	NewFilterKey NewExclusivityOrder	IN IN	FilterKey ExclusivityOrder	Sets the value of the ExclusivityOrder state variable to re-arrange the order of precedence.	402, 501, 702	R S
AvailableInterfaces Table Actions						
GetSpecific AvailableInterfacesEntry	NewAvailableInterfacesKey NewInterfaceType NewInterfaceReference	IN OUT OUT	AvailableInterfacesKey InterfaceType InterfaceReference	Retrieve an entry in the AvailableInterfaces table.	402, 501, 714	R
GetGeneric AvailableInterfacesEntry	NewAvailableInterfacesIndex NewAvailableInterfacesKeyNewInterfaceType NewInterfaceReference	IN OUT OUT OUT	AvailableInterfacesNumberOfEntries AvailableInterfacesKey InterfaceType InterfaceReference	Retrieves the AvailableInterfaces table one entry at a time. Control points can call this action with an incrementing array index until no more entries are found on the gateway. Index is ranging from 0 to AvailableInterfacesNumber-OfEntries-1.	402, 501, 713	R

End Inserted Text

2.5 Updates to Section 6.8.4 "WANIPConnection"

This section specifies the changes required in section 6.8.4 of TR-064, which defines the WANIPConnection service.

In this section, items to be added to the original text are shown in red, and deletions are shown in ~~blue~~.

Begin Change Text

6.8.4 WANIPConnection

6.8.4.1 Overview

WANIPConnection is a standard UPnP service that enables a UPnP control point to configure and control IP connections on the WAN interface of a UPnP compliant IGD. This service is required for all WANConnection Devices not employing PPP addressing. This service must not be active for WANConnection Devices that do employ PPP addressing. The allowed values of the PossibleConnectionType and ConnectionType state variables must reflect the supported ConnectionTypes of the device.

6.8.4.2 Service Modelling Definitions

ServiceType

urn:dslforum-org:service:WANIPConnection:2~~1~~

StateVariables

Variable Name	From IGD	Table	Data Type	Allowed Value	Description	Default Value	R/O
Enable	-	-	Boolean	1, 0	Enables or disables the connection.	N/A	R
ConnectionType	✓	-	String	One of PossibleConnectionType	Refer to UPnP WANIPConnection v1.01 service definition for more details.	N/A	R
PossibleConnectionTypes	✓	-	String	Unconfigured IP_Routed IP_Bridged	Refer to UPnP WANIPConnection v1.01 service definition for more details.	N/A	R
ConnectionStatus	✓	-	String	Unconfigured Connecting Connected PendingDisconnect Disconnetiong Disconnected	Refer to UPnP WANIPConnection v1.01 service definition for more details.	N/A	R
Name	-	-	String	String	Friendly Name of the connection. This name must be unique within the scope of the IGD.	N/A	R
Uptime	✓	-	ui2	>=0	Refer to UPnP WANIPConnection v1.01 service definition for more details.	(Seconds)	R
LastConnectionError	✓	-	String	One of the following: ERROR_NONE ERROR_COMMAND_ABORTED ERROR_NOT_ENABLED_FOR_INTERNET ERROR_USER_DISCONNECT ERROR_ISP_DISCONNECT ERROR_IDLE_DISCONNECT ERROR_FORCED_DISCONNECT ERROR_NO_CARRIER ERROR_IP_CONFIGURATION ERROR_UNKNOWN	Refer to UPnP WANIPConnection v1.01 service definition for more details.	N/A	R R O O O O O O O O R R
AutoDisconnectTime	✓	-	ui4	>=0	Refer to UPnP WANIPConnection v1.01 service definition for more details.	(Seconds)	O

IdleDisconnectTime	✓	-	ui4	>=0	Refer to UPnP WANIPConnection v1.01 service definition for more details.	(Seconds)	O
WarnDisconnectDelay	✓	-	ui4	>=0	Refer to UPnP WANIPConnection v1.01 service definition for more details.	(Seconds)	O
RSIPAvailable	✓	-	Boolean	0, 1	Refer to UPnP WANIPConnection v1.01 service definition for more details.	N/A	R
NATEnabled	✓	-	Boolean	0, 1	Refer to UPnP WANIPConnection v1.01 service definition for more details.	N/A	R
ExternalIPAddress	✓	-	String	IP Address	Refer to UPnP WANIPConnection v1.01 service definition for more details.	N/A	R
SubnetMask	-	-	String	IP Subnet Mask	Subnet mask of the WAN interface. This variable will only be configurable from the LAN-side if the AddressingType is Static.	N/A	R
PortMappingNumberOfEntries	✓	PortMapping (index)	ui2	>=0	Refer to UPnP WANIPConnection v1.01 service definition for more details.	N/A	R
PortMappingEnabled	✓	PortMapping	Boolean	0, 1	Refer to UPnP WANIPConnection v1.01 service definition for more details.	N/A	R
PortMappingLeaseDuration	✓	PortMapping	ui4	>=0	Refer to UPnP WANIPConnection v1.01 service definition for more details.	N/A	R
RemoteHost	✓	PortMapping KEY	String	IP Address or Empty String	Refer to UPnP WANIPConnection v1.01 service definition for more details.	Empty String	R
ExternalPort	✓	PortMapping KEY	ui2	Between 0 & 65535 inclusive	Refer to UPnP WANIPConnection v1.01 service definition for more details.	N/A	R
InternalPort	✓	PortMapping	ui2	Between 1 & 65535 inclusive	Refer to UPnP WANIPConnection v1.01 service definition for more details.	N/A	R
PortMappingProtocol	✓	PortMapping KEY	String	TCP UDP	Refer to UPnP WANIPConnection v1.01 service definition for more details.	Empty String	R
InternalClient	✓	PortMapping	String	IP Address	Refer to UPnP WANIPConnection v1.01 service definition for more details.	Empty String	R
PortMappingDescription	✓	PortMapping	String	String	Refer to UPnP WANIPConnection v1.01 service definition for more details.	Empty String	R

AddressingType	-	-	String	DHCP, Static	Represents the method used to assign an address to the WAN side interface of the IGD.	N/A	R
DefaultGateway	-	-	String	IP Address	The default gateway of the WAN interface. This variable will only be configurable from the LAN-side if the AddressingType is Static.	N/A	R
MACAddress	-	-	String	MAC Address	The physical address of the WANIPConnection if applicable.	N/A	R
MACAddressOverride	-	-	Boolean	1, 0	Whether the value of MACAddress state variable can be overridden. If true (1), the action SetMACAddress may be used to set the value of MAC-Address. If false (0), the CPE's default value is used (or restored if it had previously been overridden).	N/A	O
MaxMTUSize	-	-	ui2	Between 1 and 1540, inclusive	The maximum allowed size of an Ethernet frame from LAN-side devices.	N/A	O
DNSEnabled	-	-	Boolean	0, 1	Defines whether or not the device should attempt to query a DNS server across this connection.	1	R
DNSOverrideAllowed	-	-	Boolean	0, 1	Defines whether or not a manually set, "non-zero" DNS address can be overridden by a DNS entry received from the WAN.	0	R
DNSServers	-	-	String	String	Comma separated list of DNS servers configured on the WANIPConnection.	N/A	R
ConnectionTrigger	-	-	String	OnDemand AlwaysOn Manual	Defines the trigger used to establish the IP connection.	OnDemand	R
RouteProtocolRx	-	-	String	Off, (R) RIPv1 (O) RIPv2, (O) OSPF (O)	Defines the Rx protocol to be used.	Off	R
ShapingRate	-	-	i4	>=-1	Rate to shape this connection's egress traffic. If <= 100, in percent of the rate of the highest rate-constrained layer over which the packet will travel on egress.[5] The rate is limited over the window period specified by ShapeWindow. If > 100, in bits per second.	-1	C

[5] For example, for packets destined for a WAN DSL interface, if the ATM layer is rate-constrained, then the rate is calculated relative to this rate. Otherwise, the rate is calculated relative to the physical-layer DSL rate.

					A value of -1 indicates no shaping. Support for this variable is required only if the QueueManagement service is present.		
ShapingBurstSize		ui2	>=0		Burst Size in bytes. Support for this variable is required only if the QueueManagement service is present.	0	C

Actions, Arguments & Errors

Name	From IGD	Argument	Dir	Related State Variable(s)	Description	Errors	R/O
SetEnable	-	NewEnable	IN	Enable	Sets the value of the Enable state variable to enable or disable the connection.	402, 501	R S
GetInfo	-	NewEnable NewConnectionType NewPossibleConnectionTypes NewConnectionStatus NewName NewUptime NewLastConnectionError NewAutoDisconnectTime NewIdleDisconnectTime NewWarnDisconnectDelay NewRSIPAvailable NewNATEnabled NewExternalIPAddress NewSubnetMask NewAddressingType NewDefaultGateway NewMACAddress NewMACAddressOverride NewMaxMTUSize NewDNSEnabled NewDNSOverrideAllowed NewDNSServers NewConnectionTrigger NewRouetProtocolRx	OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT	Enable ConnectionType PossibleConnectionTypes ConnectionStatus Name Uptime LastConnectionError AutoDisconnectTime IdleDisconnectTime WarnDisconnectDelay RSIPAvailable NATEnabled ExternalIPAddress SubnetMask AddressingType DefaultGateway MACAddress MACAddressOverride MaxMTUSize DNSEnabled DNSOverrideAllowed DNSServers ConnectionTrigger RouteProtocolRx	Retrieves all of the state variables not associated with a table.	402, 501	R
SetConnectionType	✓	NewConnectionType	IN	ConnectionType	Refer to UPnP WANIPConnection v1.01 service definition for more details.	402, 501, 703	R S
GetConnectionTypeInfo	✓	NewConnectionType NewPossibleConnectionType	OUT OUT	ConnectionType PossibleConnectionTypes	Refer to UPnP WANIPConnection v1.01 service definition for more details.	402, 501	R
RequestConnection	✓	*None*	-	-	Refer to UPnP WANIPConnection v1.01 service definition for more details.	402, 704, 705, 706, 707, 708, 709, 710	R S
RequestTermination	✓	*None*	-	-	Refer to UPnP WANIPConnection v1.01 service definition for more details.	402, 501, 707, 710, 711	O S
ForceTermination	✓	*None*	-	-	Refer to UPnP WANIPConnection v1.01 service definition for more details.	402, 501, 707, 710, 711	R S
SetAutoDisconnectTime	✓	NewAutoDisconnectTime	IN	AutoDisconnectTime	Refer to UPnP WANIPConnection v1.01 service definition for more details.	402, 501	O S
SetIdleDisconnectTime	✓	NewIdleDisconnectTime	IN	IdleDisconnectTime	Refer to UPnP WANIPConnection v1.01 service definition for more details.	402, 501	O S

SetWarnDisconnectDelay	✓	NewWarnDisconnectDelay	IN	WarnDisconnectDelay	Refer to UPnP WANIPConnection v1.01 service definition for more details.	402, 501	O S
GetStatusInfo	✓	NewConnectionStatus NewLastConnectionError NewUpTime	OUT OUT OUT	ConnectionStatus LastConnectionError UpTime	Refer to UPnP WANIPConnection v1.01 service definition for more details.	402,	R
GetAutoDisconnectTime	✓	NewAutoDisconnectTime	OUT	AutoDisconnectTime	Refer to UPnP WANIPConnection v1.01 service definition for more details.	402	O
GetIdleDisconnectTime	✓	NewIdleDisconnectTime	OUT	IdleDisconnectTime	Refer to UPnP WANIPConnection v1.01 service definition for more details.	402	O
GetWarnDisconnectDelay	✓	NewWarnDisconnectDelay	OUT	WarnDisconnectDelay	Refer to UPnP WANIPConnection v1.01 service definition for more details.	402	O
GetNATRSIPStatus	✓	NewRSIPAvailavle NewNATEnabled	OUT OUT	RSIPAvailavle NATEnabled	Refer to UPnP WANIPConnection v1.01 service definition for more details.	402	R
PortMapping Table Actions							
GetPortMappingNumberOfEntries	-	NewPortMappingNumberOfEntries	OUT	PortMappingNumberOfEntries	Retrieves the value of the PortMappingNumberOfEntries state variable.	402, 501	R
GetGeneric-PortMappingEntry	-	NewPortMappingIndex NewRemoteHost NewExternalPort NewProtocol NewInternalPort NewInternalClient NewEnabled NewPortMappingDescription NewLeaseDuration	IN OUT OUT OUT OUT OUT OUT OUT OUT	PortMappingNumberOfEntries RemoteHost ExternalPort PortMappingProtocol InternalPort InternalClient PortMappingEnabled PortMappingDescription PortMappingLeaseDuration	Refer to UPnP WANIPConnection v1.01 service definition for more details.	402, 713	R
GetSpecificPortMappingEntry	✓	NewRemoteHost NewExternalPort NewProtocol NewInternalPort NewInternalClient NewEnabled NewPortMappingDescription NewLeaseDuration	IN IN IN OUT OUT OUT OUT OUT	RemoteHost ExternalPort PortMappingProtocol InternalPort InternalClient PortMappingEnabled PortMappingDescription PortMappingLeaseDuration	Refer to UPnP WANIPConnection v1.01 service definition for more details.	402, 714	R
AddPortMapping	✓	NewRemoteHost NewExternalPort NewProtocol NewInternalPort NewInternalClient NewEnabled NewPortMappingDescription NewLeaseDuration	IN IN IN IN IN IN IN IN	RemoteHost ExternalPort PortMappingProtocol InternalPort InternalClient PortMappingEnabled PortMappingDescription PortMappingLeaseDuration	Refer to UPnP WANIPConnection v1.01 service definition for more details.	402, 501, 715, 716, 718, 724, 725, 726, 727	R S
DeletePortMapping	✓	NewRemoteHost NewExternalPort NewProtocol	IN IN IN	RemoteHost ExternalPort PortMappingProtocol	Refer to UPnP WANIPConnection v1.01 service definition for more details.	402, 714	R S
GetExternalIPAddress	✓	NewExternalIPAddress	OUT	ExternalIPAddress	Refer to UPnP WANIPConnection v1.01 service definition for more details.	402, 501	R
SetIPInterfaceInfo	-	NewAddressingType NewExternalIPAddress NewSubnetMask NewDefaultGateway	IN IN IN IN	AddressingType ExternalIPAddress SubnetMask DefaultGateway	Set the WAN side IP interface. NewExternalIPAddress, NewSubnetMask, NewDefaultGateway are ignored if NewAddressingType is not "Static."	402, 501, 702	O S
SetMACAddress	-	NewMACAddress	IN	MACAddress	Sets the value of MACAddress. Valid only if MACAddress-Override is present and true (1).	402, 501, 728	O S
SetMaxMTUSize	-	NewMaxMTUSize	IN	MaxMTUSize	Set the WAN side Max MTU size.	402, 501, 702	O S

SetConnectionTrigger	-	NewConnectionTrigger	IN	ConnectionTrigger	Sets the value of the Connection Trigger variable.	402, 501	R S
SetRouteProtocolRx	-	NewRouteProtocolRx	IN	RouteProtocolRx	Sets the value of the Rx route protocol.	402, 501	R S
SetRateShaping	-	NewShapingRate NewShapingBurstSize	IN IN	ShapingRate ShapingBurstSize	Sets rate-shaping parameters. Support for this action is required only if the QueueManagement service is present.	402, 501	C S
GetRateShaping	-	NewShapingRate NewShapingBurstSize	OUT OUT	ShapingRate ShapingBurstSize	Gets rate-shaping parameters. Support for this action is required only if the QueueManagement service is present.	402, 501	C

End Change Text

2.6 Updates to Section 6.8.5 "WANPPPConnection"

This section specifies the changes required in section 6.8.5 of TR-064, which defines the WANPPPConnection service.

In this section, items to be added to the original text are shown in red, and deletions are shown in ~~blue~~.

Begin Change Text

6.8.5 WANPPPConnection

6.8.5.1 Overview

WANPPPConnection is a standard UPnP service that enables a UPnP control point to configure and control PPP connections on the WAN interface of a UPnP compliant IGD. This service is required if the WANConnection Device employs PPP connections. This service must NOT be active if the WANConnection Device does not support PPP connections. The allowed values of the PossibleConnectionType and ConnectionType state variables must reflect the supported ConnectionTypes of the device.

6.8.5.2 Service Modelling Definitions

ServiceType

urn:dslforum-org:service:WANPPPConnection:2~~1~~

StateVariables

Variable Name	From IGD	Table	Data Type	Allowed Value	Description	Default Value	R/O
Enable	-	-	Boolean	1, 0	Enables or disables the connection.	N/A	R
ConnectionType	✓	-	String	One of PossibleConnectionTypes	Refer to UPnP WANPPPConnection v1.01 service definition for more details.	N/A	R
PossibleConnectionTypes	✓	-	String	Unconfigured IP_Routed DHCP_Spoofed PPPoE_Bridged PPTP_Relay L2TP_Relay PPPoE_Relay	Refer to UPnP WANPPPConnection v1.01 service definition for more details.	N/A	R

ConnectionStatus	✓	-	String	Unconfigured Connecting Authenticating Connected PendingDisconnect Disconnecting Disconnected	Refer to UPnP WANPPPConnection v1.01 service definition for more details.	N/A	R
Name	-	-	String	String	Friendly Name of the connection. This name must be unique within the scope of the IGD.	N/A	R
Uptime	✓	-	ui4	>=0	Refer to UPnP WANPPPConnection v1.01 service definition for more details.	(Seconds)	R
UpstreamMaxBitRate	✓	-	ui4	>=0	Refer to UPnP WANPPPConnection v1.01 service definition for more details.	(bits/second)	R
DownstreamMaxBitRate	✓	-	ui4	>=0	Refer to UPnP WANPPPConnection v1.01 service definition for more details.	(bits/second)	R
LastConnectionError	✓	-	String	One of the following: ERROR_NONE ERROR_ISP_TIME_OUT ERROR_COMMAND_ABORTED ERROR_NOT_ENABLED_FOR_INTERNET ERROR_BAD_PHONE_NUMBER ERROR_USER_DISCONNECT ERROR_ISP_DISCONNECT ERROR_IDLE_DISCONNECT ERROR_FORCED_DISCONNECT ERROR_SERVER_OUT_OF_RESOURCES ERROR_RESTRICTED_LOGON_HOURS ERROR_ACCOUNT_DISABLED ERROR_ACCOUNT_EXPIRED ERROR_PASSWORD_EXPIRED ERROR_AUTHENTICATION_FAILURE ERROR_NO_DIALTONE ERROR_NO_CARRIER ERROR_NO_ANSWER ERROR_LINE_BUSY ERROR_UNSUPPORTED_BITSPERSECOND ERROR_TOO_MANY_LINE_ERRORS ERROR_IP_CONFIGURATION ERROR_UNKNOWN	Refer to UPnP WANPPPConnection v1.01 service definition for more details.	N/A	R R O O O O O O O O O O O O O O R O O O O O O O R R
AutoDisconnectTime	✓	-	ui4	>=0	Refer to UPnP WANPPPConnection v1.01 service definition for more details.	(Seconds)	O
IdleDisconnectTime	✓	-	ui4	>=0	Refer to UPnP WANPPPConnection v1.01 service definition for more details.	(Seconds)	O
WarnDisconnectDelay	✓	-	ui4	>=0	Refer to UPnP WANPPPConnection v1.01 service definition for more details.	(Seconds)	O
ConnectionTrigger	-	-	String	OnDemand AlwaysOn Manual	Defines the trigger used to establish the PPP connection.	OnDemand	R
RSIPAvailable	✓	-	Boolean	0, 1	Refer to UPnP WANPPPConnection v1.01 service definition for more details.	N/A	R
NATEnabled	✓	-	Boolean	0, 1	Refer to UPnP WANPPPConnection v1.01 service definition for more details.	N/A	R
UserName	✓	-	String	Alphanumeric Text	Refer to UPnP WANPPPConnection v1.01 service definition for more details.	Empty String	R
Password	✓	-	String	Alphanumeric Text	Refer to UPnP WANPPPConnection v1.01 service definition for more details.	Empty String	R

PPPEncryptionProtocol	✓	-	String	MPPE, ...	Refer to UPnP WANPPPConnection v1.01 service definition for more details.	N/A	O
PPPCompressionProtocol	✓	-	String	STAC LZS Van Jacobsen	Refer to UPnP WANPPPConnection v1.01 service definition for more details.	N/A	O
PPPAuthenticationProtocol	✓	-	String	PAP CHAP MS-CHAP	Refer to UPnP WANPPPConnection v1.01 service definition for more details.	N/A	O
ExternalIPAddress	✓	-	String	IP Address	Refer to UPnP WANPPPConnection v1.01 service definition for more details.	N/A	R
RemoteIPAddress	-	-	String	IP Address	The remote IP address of the WANPPPConneciton.	N/A	O
MaxMRUSize	-	-	ui2	Between 1 and 1540, inclusive	The maximum allowed size of frames sent from the remote peer.	N/A	O
CurrentMRUSize	-	-	ui2	Between 1 and 1540, inclusive	The current MRU in use over this connection.	N/A	O
PortMappingNumberOfEntries	✓	PortMapping (index)	ui2	>=0	Refer to UPnP WANPPPConnection v1.01 service definition for more details.	N/A	R
PortMappingEnabled	✓	PortMapping	Boolean	0,1	Refer to UPnP WANPPPConnection v1.01 service definition for more details.	N/A	R
PortMappingLeaseDuration	✓	PortMapping	ui4	0 to maximum value of ui4 (in seconds)	Refer to UPnP WANPPPConnection v1.01 service definition for more details.	N/A	R
RemoteHost	✓	PortMapping KEY	String	IP Address or empty string	Refer to UPnP WANPPPConnection v1.01 service definition for more details.	N/A	R
ExternalPort	✓	PortMapping KEY	ui2	Between 1 and 65535 inclusive	Refer to UPnP WANPPPConnection v1.01 service definition for more details.	N/A	R
InternalPort	✓	PortMapping	ui2	Between 1 and 65535 inclusive	Refer to UPnP WANPPPConnection v1.01 service definition for more details.	N/A	R
PortMappingProtocol	✓	PortMapping KEY	String	TCP UDP	Refer to UPnP WANPPPConnection v1.01 service definition for more details.	N/A	R
InternalClient	✓	PortMapping	String	IP Address	Refer to UPnP WANPPPConnection v1.01 service definition for more details.	N/A	R
PortMappingDescription	✓	PortMapping	String	String	Refer to UPnP WANPPPConnection v1.01 service definition for more details.	N/A	R
MACAddress	-	-	String	MAC Address	The physical address of the WANPPPConnection if applicable.	N/A	R
MACAddressOverride	-	-	Boolean	1, 0	Whether the value of MACAddress state variable can be overridden. If true (1), the action Set-MACAddress may be used to set the value of MACAddress. If false (0), the CPE's default value is used (or restored if it had previously been	N/A	O

					overridden).		
DNSEnabled	-	-	Boolean	0, 1	Defines whether or not the device should attempt to query a DNS server across this connection.	1	R
DNSOverrideAllowed	-	-	Boolean	0, 1	Defines whether or not a manually set, "non-zero" DNS address can be overridden by a DNS entry received from the WAN.	0	R
DNSServers	-	-	String	String	Comma separated list of DNS servers configured on the WANIPConnection.	N/A	R
TransportType	-	-	String	PPPoA PPPoE L2TP (for future use) PPTP (for future use)	PPP Transport type of the WANPPPConnection.	N/A	R
PPPoEACName	-	-	String	String	PPPoE Access Concentrator.	N/A	R
PPPoEServiceName	-	-	String	String	PPPoE Service Name.	N/A	R
RouteProtocolRx	-	-	String	Off, (R) RIPv1 (O) RIPv2, (O) OSPF (O)	Defines the Rx protocol to be used.	Off	R
PPPLCPEcho	-	-	ui2	-	PPP LCP Echo period in seconds.	N/A	O
PPPLCPEchoRetry	-	-	ui2	-	Number of PPP LCP Echo retries within an echo period.	N/A	O
ShapingRate	-	-	i4	>=-1	Rate to shape this connection's egress traffic. If <= 100, in percent of the rate of the highest rate-constrained layer over which the packet will travel on egress.[6] The rate is limited over the window period specified by ShapeWindow. If > 100, in bits per second. A value of -1 indicates no shaping. Support for this variable is required only if the QueueManagement service is present.	-1	R
ShapingBurstSize			ui2	>=0	Burst Size in bytes. Support for this variable is required only if the QueueManagement service is present.	0	R

Actions, Arguments & Errors

Name	From IGD	Argument	Dir	Related State Variable(s)	Description	Errors	R/O

[6] For example, for packets destined for a WAN DSL interface, if the ATM layer is rate-constrained, then the rate is calculated relative to this rate. Otherwise, the rate is calculated relative to the physical-layer DSL rate.

SetEnable	-	NewEnable	IN	Enable	Sets the value of the Enable state variable to enable or disable the connection.	402, 501	R S
GetInfo	-	NewEnable NewConnectionType NewPossibleConnectionTypes NewConnectionStatus NewName NewUptime NewUpstreamMaxBitRate NewDownstreamMaxBitRate NewLastConnectionError NewAutoDisconnectTime NewIdleDisconnectTime NewWarnDisconnectDelay NewConnectionTrigger NewRSIPAvailable NewNATEnabled NewUserName NewPPPEncryptionProtocol NewPPPCompressionProtocol NewPPPAuthenticationProtocol NewExternalIPAddress NewRemoteIPAddress NewMACAddress NewMACAddressOverride NewMaxMRUSize NewCurrentMRUSize NewDNSEnabled NewDNSOverrideAllowed NewDNSServers NewTransportType NewPPPoEACName NewPPPoEServiceName NewRouteProtocolRx NewPPPLCPEcho NewPPPLCPEchoRetry	OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT OUT	Enable ConnectionType PossibleConnectionTypes ConnectionStatus Name Uptime UpstreamMaxBitRate DownstreamMaxBitRate LastConnectionError AutoDisconnectTime IdleDisconnectTime WarnDisconnectDelay ConnectionTrigger RSIPAvailable NATEnabled UserName PPPEncryptionProtocol PPPCompressionProtocol PPPAuthenticationProtocol ExternalIPAddress RemoteIPAddress MACAddress MACAddressOverride MaxMRUSize CurrentMRUSize DNSEnabled DNSOverrideAllowed DNSServers TransportType PPPoEACName PPPoEServiceName RouteProtocolRx PPPLCPEcho PPPLCPEchoRetry	Retrieves all of the state variables not associated with a table. Also excludes Password, which is not readable.	402, 501	R
SetConnectionType	✓	NewConnectionType	IN	ConnectionType	Refer to UPnP WANPPPConnection v1.01 service definition for more details.	402, 501, 703	R S
GetConnectionTypeInfo	✓	NewConnectionType NewPossibleConnectionType	OUT OUT	ConnectionType PossibleConnectionTypes	Refer to UPnP WANPPPConnection v1.01 service definition for more details.	402, 501	R
RequestConnection	✓	*None*	-	-	Refer to UPnP WANPPPConnection v1.01 service definition for more details.	402, 704, 705 706, 707 708, 709 710	R S
RequestTermination	✓	*None*	-	-	Refer to UPnP WANPPPConnection v1.01 service definition for more details.	402, 501, 707, 710, 711	O S
ForceTermination	✓	*None*	-	-	Refer to UPnP WANPPPConnection v1.01 service definition for more details.	402, 501, 707, 710, 711	R S
SetAutoDisconnectTime	✓	NewAutoDisconnectTime	IN	AutoDisconnectTime	Refer to UPnP WANPPPConnection v1.01 service definition for more details.	402, 501	O S
SetIdleDisconnectTime	✓	NewIdleDisconnectTime	IN	IdleDisconnectTime	Refer to UPnP WANPPPConnection v1.01 service definition for more details.	402, 501	O S
SetWarnDisconnectDelay	✓	NewWarnDisconnectDelay	IN	WarnDisconnectDelay	Refer to UPnP WANPPPConnection v1.01 service definition for more details.	402, 501	O S
GetStatusInfo	✓	NewConnectionStatus NewLastConnectionError NewUpTime	OUT OUT OUT	ConnectionStatus LastConnectionError UpTime	Refer to UPnP WANPPPConnection v1.01 service definition for more details.	402,	R
GetLinkLayerMaxBitRates	✓	NewUpstreamMaxBitRate NewDownstreamMaxBitRate	OUT OUT	UpstreamMaxBitRate DownstreamMaxBitRate	Refer to UPnP WANPPPConnection v1.01 service definition for more details.	402,	R
GetPPPEncryptionProtocol	✓	NewPPPEncryptionProtocol	OUT	PPPEncryptionProtocol	Refer to UPnP WANPPPConnection v1.01 service definition for more	402,	O

					details.		
GetPPPCompressionProtocol	✓	NewPPPCompressionProtocol	OUT	PPPCompressionProtocol	Refer to UPnP WANPPPConnection v1.01 service definition for more details.	402,	O
GetPPPAuthenticationProtocol	✓	NewPPPAuthenticationProtocol`	OUT	PPPAuthenticationProtocol	Refer to UPnP WANPPPConnection v1.01 service definition for more details.	402,	O
GetUserName	✓	NewUserName	OUT	UserName	Refer to UPnP WANPPPConnection v1.01 service definition for more details.	402,	R S
SetUserName	-	NewUserName	IN	UserName	Sets the value of the related state variable.	402, 501, 702, 703	R S
SetPassword	-	NewPassword	IN	Password	Sets the value of the related state variable.	402, 501, 702, 703	R S
GetAutoDisconnectTime	✓	NewAutoDisconnectTime	OUT	AutoDisconnectTime	Refer to UPnP WANPPPConnection v1.01 service definition for more details.	402	O
GetIdleDisconnectTime	✓	NewIdleDisconnectTime	OUT	IdleDisconnectTime	Refer to UPnP WANPPPConnection v1.01 service definition for more details.	402	O
GetWarnDisconnectDelay	✓	NewWarnDisconnectDelay	OUT	WarnDisconnectDelay	Refer to UPnP WANPPPConnection v1.01 service definition for more details.	402	O
GetNATRSIPStatus	✓	NewRSIPAvailavle NewNATEnabled	OUT OUT	RSIPAvailavle NATEnabled	Refer to UPnP WANPPPConnection v1.01 service definition for more details.	402	R
SetPPPoEService	-	NewPPPoEACName NewPPPoEServiceName	IN IN	PPPoEACName PPPoEServiceName	Sets the values of the PPPoEACName and PPPoEServiceName state variables.	402, 501	O S
SetConnectionTrigger	-	NewConnectionTrigger	IN	ConnectionTrigger	Sets the value of the Connection Trigger variable.	402, 501	R S
PortMappingTable Actions							
GetPortMappingNumberOfEntries	-	NewPortMappingNumberOfEntries	OUT	PortMappingNumberOfEntries	Retrieves the value of the PortMappingNumberOfEntries state variable.	402, 501	R
GetGeneric-PortMappingEntry	-	NewPortMappingIndex NewRemoteHost NewExternalPort NewProtocol NewInternalPort NewInternalClient NewEnabled NewPortMappingDescription NewLeaseDuration	IN OUT OUT OUT OUT OUT OUT OUT OUT	PortMappingNumberOfEntries RemoteHost ExternalPort PortMappingProtocol InternalPort InternalClient PortMappingEnabled PortMappingDescription PortMappingLeaseDuration	Refer to UPnP WANIPConnection v1.01 service definition for more details.	402, 713	R
GetSpecificPortMappingEntry	✓	NewRemoteHost NewExternalPort NewProtocol NewInternalPort NewInternalClient NewEnabled NewPortMappingDescription NewLeaseDuration	IN IN IN OUT OUT OUT OUT OUT	RemoteHost ExternalPort PortMappingProtocol InternalPort InternalClient PortMappingEnabled PortMappingDescription PortMappingLeaseDuration	Refer to UPnP WANIPConnection v1.01 service definition for more details.	402, 714	R
AddPortMapping	✓	NewRemoteHost NewExternalPort NewProtocol NewInternalPort NewInternalClient NewEnabled NewPortMappingDescription NewLeaseDuration	IN IN IN IN IN IN IN IN	RemoteHost ExternalPort PortMappingProtocol InternalPort InternalClient PortMappingEnabled PortMappingDescription PortMappingLeaseDuration	Refer to UPnP WANIPConnection v1.01 service definition for more details.	402, 501, 715, 716, 718, 724, 725, 726, 727	R S
DeletePortMapping	✓	NewRemoteHost NewExternalPort NewProtocol	IN IN IN	RemoteHost ExternalPort PortMappingProtocol	Refer to UPnP WANIPConnection v1.01 service definition for more	402, 714	R S

					details.		
GetExternalIPAddress	✓	NewExternalIPAddress	OUT	ExternalIPAddress	Refer to UPnP WANIPConnection v1.01 service definition for more details.	402, 501	R
SetMACAddress	-	NewMACAddress	IN	MACAddress	Sets the value of MACAddress. Valid only if MACAddress-Override is present and true (1).	402, 501, 728	O S
SetMaxMRUSize	-	NewMaxMRUSize	IN	MaxMRUSize	Set the WAN side Max MRU size.	402, 501, 702	O S
SetRouteProtocolRx	-	NewRouteProtocolRx	IN	RouteProtocolRx	Sets the value of the Rx route protocol.	402, 501	R S
SetRateShaping	-	NewShapingRate NewShapingBurstSize	IN IN	ShapingRate ShapingBurstSize	Sets rate-shaping parameters. Support for this action is required only if the QueueManagement service is present.	402, 501	C S
GetRateShaping	-	NewShapingRate NewShapingBurstSize	OUT OUT	ShapingRate ShapingBurstSize	Gets rate-shaping parameters. Support for this action is required only if the QueueManagement service is present.	402, 501	C

End Change Text

2.7 Addition of Appendix on Queuing and Bridging

This section specifies a new Appendix to be added to TR-064 that describes the queuing and bridging model association with this specification.

Begin Inserted Text

Appendix A. Queuing and Bridging

Figure A-1 shows the queuing and bridging model for an Internet Gateway Device. This model relates to the QueueManagement object as well as the Layer2Bridging and Layer3Forwarding objects. The elements of this model are described in the following sections.

Note – the queuing model described in this Appendix is meant strictly as a model to clarify the intended behavior of the related data objects. There is no implication intended that an implementation must be structured to conform to this model.

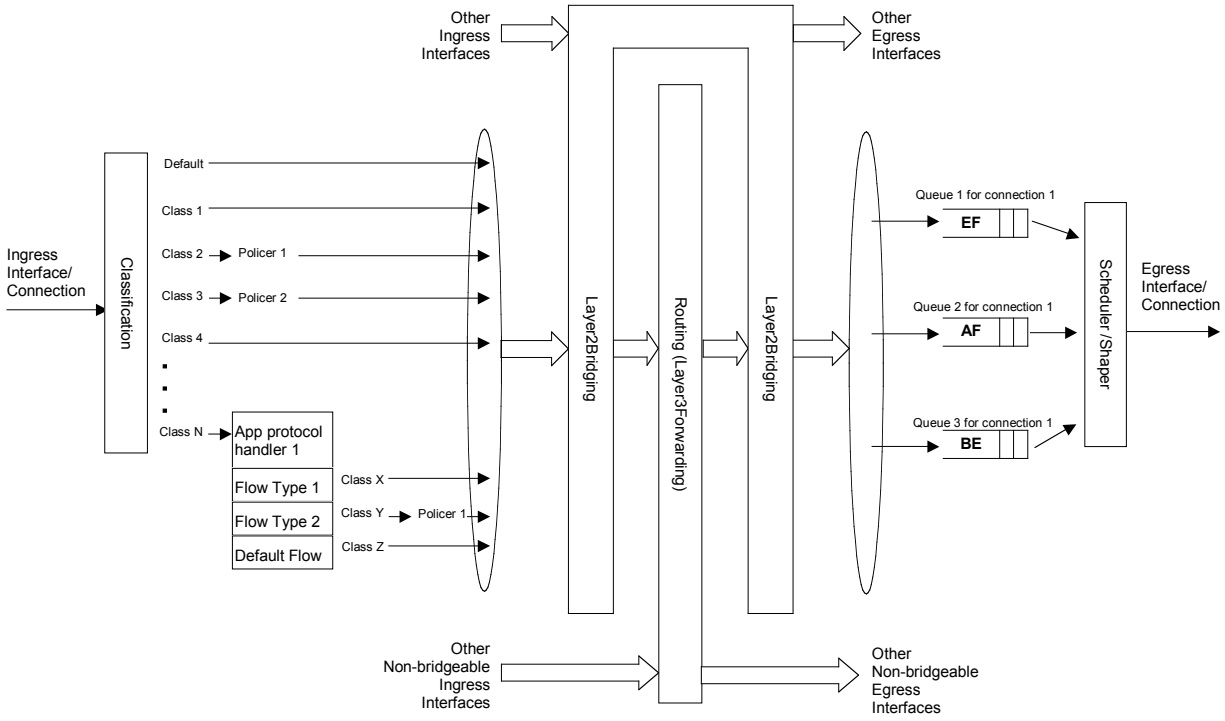

Figure A-1 – Queuing model of an Internet Gateway Device

A.1 Packet Classification

The Classification table within the QueueManagement object specifies the assignment of each packet arriving at an ingress interface to a specific internal class. This classification may be based on a number of matching criteria, such as destination and source IP address, destination and source port, and protocol.

Each entry in the Classification table includes a series of elements, each indicated to be a Classification Criterion. Each classification criterion can be set to a specified value, or can be set to a value that indicates that criterion is not to be used. A packet is defined to match the classification criteria for that table entry only the packet matches all of the specified criteria. That is, a logical AND operation is applied across all classification criteria within a given Classification table entry.

> *Note – to apply a logical OR to sets of classification criteria, multiple entries in the Classification table can be created that specify the same resulting queuing behavior.*

For each classification criterion, the Classification table also includes a corresponding "exclude" flag. This flag may be used to invert the sense of the associated classification criterion. That is, if this flag is false for a given criterion, the classifier is to include only packets that meet the specified criterion (as well as all others). If this flag is true for a given criterion, the classifier is to include all packets except those that meet the associated criterion (in addition to meeting all other criteria).

For a given entry in the Classification table, the classification is to apply only to those interfaces specified by the ClassInterface element. This element may specify a particular ingress interface, all LAN-side interfaces, all WAN-side interfaces, a local IP-layer source within the Internet Gateway Device, or all sources. Depending on the particular interface, not all classification criteria may be applicable. For example, Ethernet layer classification criteria would not apply to packets arriving on a non-bridged ATM VC.

Packet classification is modeled to include all ingress packets regardless of whether they ultimately will be bridged or routed through the Internet Gateway Device. The packet classifier is not modeled to apply to

packets that are embedded in a tunnelled connection (such as, PPPoE, L2TP, or tunnelled IPsec). In such cases, classification would apply only to the outer tunnel packets, but not the embedded packets contained within. An exception is for tunnels that terminate in the Internet Gateway Device itself. That is, for connections that terminate in the Internet Gateway Device, such as a PPP connection, the classification is applied to the IP packets contained within.

A.1.1 Classification Order

The class assigned to a given packet corresponds to the first entry in the Classification table (given the specified order of the entries in the table) whose matching criteria match the packet. If there is no entry that matches the packet, the packet is assigned to a default class.

Classification rules are sensitive to the order in which they are applied because certain traffic may meet the criteria of more than one Classification table entry. The ClassificationOrder parameter is responsible for identifying the order in which the Classification entries are to be applied.

The following rules apply to the use and setting of the ClassificationOrder parameter:

- ClassificationOrder goes in order from 1 to n, where n is equal to the number of entries in the Classification table. 1 is the highest precedence, and n the lowest. For example, if entries with ClassificationOrder of 4 and 7 both have rules that match some particular traffic, the traffic will be classified according to the entry with the 4.
- The CPE is responsible for ensuring that all ClassificationOrder values are unique and sequential.
    - If an entry is added (number of entries becomes n+1), and the value specified for ClassificationOrder is greater than n+1, then the CPE will set ClassificationOrder to n+1.
    - If an entry is added (number of entries becomes n+1), and the value specified for ClassificationOrder is less than n+1, then the CPE will create the entry with that specified value, and increment the ClassificationOrder value of all existing entries with ClassificationOrder equal to or greater than the specified value.
    - If an entry is deleted, the CPE will decrement the ClassificationOrder value of all remaining entries with ClassificationOrder greater than the value of the deleted entry.
    - If the ClassificationOrder value of an entry is changed, then the value will also be changed for other entries greater than or equal to the lower of the old and new values, and less than the larger of the old and new values. If the new value is less than the old, then these other entries will all have ClassificationOrder incremented. If the new value is greater than the old, then the other entries will have ClassificationOrder decremented and the changed entry will be given a value of <new value>-1. For example, an entry is changed from 8 to 5. The existing 5 goes to 6, 6 to 7, and 7 to 8. If the entry goes from 5 to 8, then 6 goes to 5, 7 to 6, and the changed entry is 7. This is consistent with the behavior that would occur if the change were considered to be an Add of a new entry with the new value, followed by a Delete of the entry with the old value.

A.1.2 Dynamic Application Specific Classification

In some situations, traffic to be classified cannot be identified by a static set of classification criteria. Instead, identification of traffic flows may require explicit application awareness. The model accommodates such situations via the App and Flow tables in the QueueManagement object.

Each entry in the App table is associated with an application-specific protocol handler, identified by the ProtocolIdentifier, which contains a URN. For a particular CPE, the AvailableAppList parameter indicates which protocol handlers that CPE is capable of supporting, if any. A list of standard protocol handlers and their associated URNs is specified in Appendix B, though a CPE may also support vendor-specific protocol handlers as well. Multiple App table entries may refer to the same ProtocolIdentifier.

The role of the protocol handler is to identify and classify flows based on application awareness. For example, a SIP protocol handler might identify a call-control flow, an audio flow, and a video flow. The App and Flow tables are used to specify the classification outcome associated with each such flow.

For each App table entry there may be one or more associated Flow table entries. Each flow table identifies a type of flow associated with the protocol handler. The FlowType element is used to identify the specific type of flow associated with each entry. For example, a Flow table entry for a SIP protocol handler might refer only to the audio flows associated with that protocol handler. A list of standard FlowType values is given in Appendix B, though a CPE may also support vendor-specific flow types.

A protocol handler may be defined as being fed from the output of a Classification table entry. That is, a Classification entry may be used to single out control traffic to be passed to the protocol handler, which then subsequently identifies associated flows. Doing so allows more than one instance of a protocol handler associated with distinct traffic. For example, one could define two App table entries associated with SIP protocol handlers. If the classifier distinguished control traffic to feed into each handler based on the destination IP address of the SIP server, this could be used to separately classify traffic for different SIP service providers. In this case, each instance of the protocol handler would identify only those flows associated with a given service. Note that the Classification table entry that feeds each protocol handler wouldn't encompass all of the flows; only the traffic needed by the protocol handler to determine the flows—typically only the control traffic.

A.1.3 Classification Outcome

Each Classification entry specifies a tuple composed of either:

- A Queue and optional Policer, or
- An App table entry

Each entry also specifies:

- Outgoing DiffServ and Ethernet priority marking behavior
- A ForwardingPolicy tag that may be referenced in the Layer3Forwarding table to affect packet routing (note that the ForwardingPolicy tag affects only routed traffic)

Note that the information associated with the classification outcome is modeled as being carried along with each packet as it flows through the system.

If there a packet does not match any Classification table entry, the DefaultQueue, DefaultPolicer, default markings, and default ForwardingPolicy are used.

If a Queue/Policer tuple is specified, classification is complete. If, however, an App is specified, the packet is passed to the protocol handler specified by the ProtocolIdentifier in the specified App table entry for additional classification (see section A.1.2). If any of the identified flows match the FlowType specified in any Flow table entry corresponding to the given App table entry (this correspondence is indicated by the App identifier), the specified tuple and markings for that Flow table entry is used for packets in that flow. Other flows associated with the application, but not explicitly identified, use the default tuple and markings specified for that App table entry.

A.2 Policing

The Policer table defines the policing parameters for ingress packets identified by either a Classification table entry (or the default classification) or a dynamic flow identified by a protocol handler identified in the App table.

Each Policer table entry specifies the packet handling characteristics, including the rate requirements and behavior when these requirements are exceeded.

A.3 Queuing and Scheduling

The Queue table specifies the number and types of queues, queue parameters, shaping behavior, and scheduling algorithm to use. Each Queue table entry specifies a set of egress interfaces for which a queue with the corresponding characteristics must exist.

Note – If the CPE can determine that among the interfaces specified for a queue to exist, packets classified into that queue cannot egress to a subset of those interfaces (from knowledge of the current routing and bridging configuration), the CPE may choose not to instantiate the queue on those interfaces.

Note – Packets classified into a queue that exit through an interface for which the queue is not specified to exist, must instead use the default queuing behavior. The default queue itself must exist on all egress interfaces.

The model defined here is not intended to restrict where the queuing is implemented in an actual implementation. In particular, it is up to the particular implementation to determine at what protocol layer it is most appropriate to implement the queuing behavior (IP layer, Ethernet MAC layer, ATM layer, etc.). In some cases, however, the QueueManagement configuration would restrict the choice of layer where queueing can be implemented. For example, if a queue is specified to carry traffic that is bridged, then it could not be implemented as an IP-layer queue.

Note – care should be taken to avoid having multiple priority queues multiplexed onto a single connection that is rate shaped. In such cases, it the possibility exists that high priority traffic can be held back due to rate limits of the overall connection exceeded by lower priority traffic. Where possible, each priority queue should be shaped independently using the shaping parameters in the Queue table.

The scheduling parameters defined in the Queue table apply to the first level of what may be a more general scheduling hierarchy. This specification does not specify the rules that an implementation must apply to determine the most appropriate scheduling hierarchy given the scheduling parameters defined in the Queue table.

As an example, take a situation where the output of four distinct queues is to be multiplexed into a single connection, and two entries share one set of scheduling parameters while the other two entries share a different set of scheduling parameters. In this case, it may be appropriate to implement this as a scheduling hierarchy with the first two queues multiplexed with a scheduler defined by the first pair, and the second two queues being multiplexed with a scheduler defined by the second pair. The lower layers of this scheduling hiearchy cannot be directly determined from the content of the Queue table.

A.4 Bridging

For each interface, the output of the classifier is modeled to feed a set of layer-2 bridges as specified by the Layer2Bridging object. Each bridge specifies layer-2 connectivity between one or more layer-2 LAN and/or WAN interfaces, and optionally one or more layer-3 connections to the local router.

Each bridge corresponds to a single entry in the Bridge table of the Layer2Bridging. Each entry contains (by reference) one or more Filter table entries. Each Filter table entry specifies an interface or set of interfaces to include in the bridge, and may also specify layer-2 filter criteria to selectively bridge traffic among the specified interfaces.

Each Filter table entry selects one or more interfaces among those listed in the AvailableInterfaces table. This table would normally include all layer-2 interfaces that include an Ethernet MAC layer. This would exclude, for example, a non-bridged ATM VC carrying IPoA or PPPoA. A given entry may refer to a specific layer-2 interface, all available LAN interfaces, all available WAN interfaces, or all available LAN and WAN interfaces. A Filter table entry may also include LAN-side or WAN-side layer-3 connections to the local router, such as PPP or IP connections. When including a layer-3 connection in a bridge, this overrides the default association of that connection with a layer-2 object as indicated by the connection object hierarchy.

Note – from the point of view of a bridge, packets arriving into the bridge from the local router (either LAN-side or WAN-side) are treated as ingress packets, even though the same packets, which just left the router, are treated as egress from the point of the router. For example, a Filter table entry might admit packets on ingress to the bridge from a particular WANIPConnection, which means that it admits packets on their way out of the router over this layer-3 connection.

A.4.1 Filtering

Traffic from a given interface (or set of interfaces) may be selectively admitted to a given Bridge, rather than bridging all traffic from that interface. Each entry in the Filter table includes a series of filter criteria. Each filter criterion can be set to a specified value, or can be set to a value that indicates that criterion is not to be used. A packet is admitted to the Bridge only if the packet matches all of the specified criteria. That is, a logical AND operation is applied across all filter criteria within a given Filter table entry.

> *Note – to apply a logical OR to sets of filter criteria, multiple entries in the Filter table can be created that refer to the same interfaces and the same Bridge table entry.*

For each filter criterion, the Filter table also includes a corresponding "exclude" flag. This flag may be used to invert the sense of the associated filter criterion. That is, if this flag is false for a given criterion, the Bridge will admit only packets that meet the specified criterion (as well as all others). If this flag is true for a given criterion, the Bridge will admit all packets except those that meet the associated criterion (in addition to meeting all other criteria).

Note that because the filter criteria are based on layer-2 packet information, if the selected interface for a given Filter table entry is a layer-3 connection from the local router, the layer-2 filter criteria do not apply.

A.4.2 Exclusivity Order

Each Filter table entry is defined as either exclusive or non-exclusive. Any packet that matches the filter criteria of one or more exclusive filters is admitted to the Bridge associated with the first exclusive entry in the Filter table (relative to the specified ExclusivityOrder).

If there is no exclusive filter that matches a packet, then the packet is admitted to all Bridges associated with non-exclusive filters that match the packet.

The following rules apply to the use and setting of the ExclusivityOrder parameter:

- If the ExclusivityOrder is zero, the filter is defined to be non-exclusive.
- If the ExclusivityOrder is one or greater, the filter is defined to be exclusive.
- Among exclusive filters, the ExclusivityOrder goes in order from 1 to n, where n is equal to the number of exclusive filters. 1 is the highest precedence, and n the lowest.
- The CPE is responsible for ensuring that all ExclusivityOrder values among exclusive filters are unique and sequential.
    - If an exclusive filter is added (number of exclusive filters becomes n+1) or a non-exclusive filter is changed to be exclusive, and the value specified for ExclusivityOrder is greater than n+1, then the CPE will set ExclusivityOrder to n+1.
    - If an exclusive filter is added (number of entries becomes n+1) or a non-exclusive filter is changed to be exclusive, and the value specified for ExclusivityOrder is less than n+1, then the CPE will create the entry with that specified value, and increment the ExclusivityOrder value of all existing exclusive filters with ExclusivityOrder equal to or greater than the specified value.
    - If an exclusive filter is deleted or an exclusive filter is changed to non-exclusive, the CPE will decrement the ExclusivityOrder value of all remaining exclusive filter with ExclusivityOrder greater than the value of the deleted entry.
    - If the ExclusivityOrder value of an exclusive filter is changed, then the value will also be changed for other exclusive filters greater than or equal to the lower of the old and new values, and less than the larger of the old and new values. If the new value is less than the old, then these other entries will all have ExclusivityOrder incremented. If the new value is greater than the old, then the other entries will have ExclusivityOrder decremented and the changed entry will be given a value of <new value>-1. For example, an entry is changed from 8 to 5. The existing 5 goes to 6, 6 to 7, and 7 to 8. If the entry goes from 5 to 8, then 6 goes to 5, 7 to 6, and the changed entry is 7. This is consistent with the behavior that would occur if the

change were considered to be an Add of a new exclusive filter with the new value, followed by a Delete of the exclusive filter with the old value.

A.4.2 Egress from a Bridge

Packets admitted to a bridge from any interface are bridged across all of the interfaces considered part of that bridge. An interface is considered part of a bridge if it is specified by any of the Filter table or Marking table entries that are associated with the bridge. That is, the union of all interfaces specified either for potential admission into the bridge or for special marking treatment on egress are considered part of the bridge. This may include both layer-2 interfaces as well as layer-3 connections to the local router.

For a given bridge, packets on egress may optionally be marked distinctly for specific interfaces. The Marking table allows the CPE to be configured to selective either remove all VLANID/priority marking from a packet on egress, or modify the VLANID and/or Ethernet priority marking on egress. This may be done selectively per interface, across a class of interfaces, or for all interfaces.

A.5 Example Queuing Architecture for RG (from TR-059)

The queuing and scheduling discipline envisioned upstream for the RG is shown in Figure A-2.

There are multiple access sessions supported in this model, however, all traffic is classified and scheduled in a monolithic system. So, while it might appear at first that the Diffserv queuing and scheduling might apply only to IP-aware access – in fact all access, IP, Ethernet, or PPP is managed by the same system that adheres to the Diffserv model.

For example, at the bottom of the figure, BE treatment is given to the non-IP-aware access sessions (PPPoE started behind the RG or delivered to an L2TP tunnel delivery model). This queue might be repeated several times in order to support fairness among multiple PPPoE accesses – or it may be a monolithic queue with separate rate limiters applied to the various access sessions.

The PTA access is a single block of queues. This is done because NSP access typically works with a single default route to the NSP, and managing more than one simultaneously at the RG would be perilous. The ∑ rate limiter would limit the overall access traffic for a service provider.

Rate limiters are also shown within the EF and AF service classes because the definition of those Diffserv types is based on treating the traffic differently when it falls into various rates.

Finally, at the top of the diagram is the ASP access block of queues. In phase 1A, these queues are provisioned and provide aggregate treatment of traffic mapped to them. In phase 1B, it will become possible to assign AF queues to applications to give them specific treatment instead of aggregate treatment. The EF service class may also require a high degree of coordination among the applications that make use of it so that its maximum value is not exceeded.

Notable in this architecture is that all the outputs of the EF, AF, and BE queues are sent to a scheduler (**S**) that pulls traffic from them in a strict priority fashion. In this configuration EF traffic is, obviously, given highest precedence and BE is given the lowest. The AF service classes fall in-between.

Note that there is significant interest in being able to provide a service arrangement that would allow general Internet access to have priority over other (bulk rate) services.[7] Such an arrangement would be accomplished by assigning the bulk rate service class to BE and by assigning the default service class (Internet access) as AF with little or no committed information rate.

Given this arrangement, the precedence of traffic shown in the figure is arranged as:

1. EF – red dotted line
2. AF – blue dashed line (with various precedence among AF classes as described in RFC2597)

[7] This "bulk rate" service class would typically be used for background downloads and potentially for peer-to-peer applications as an alternative to blocking them entirely.

3. BE – black solid line

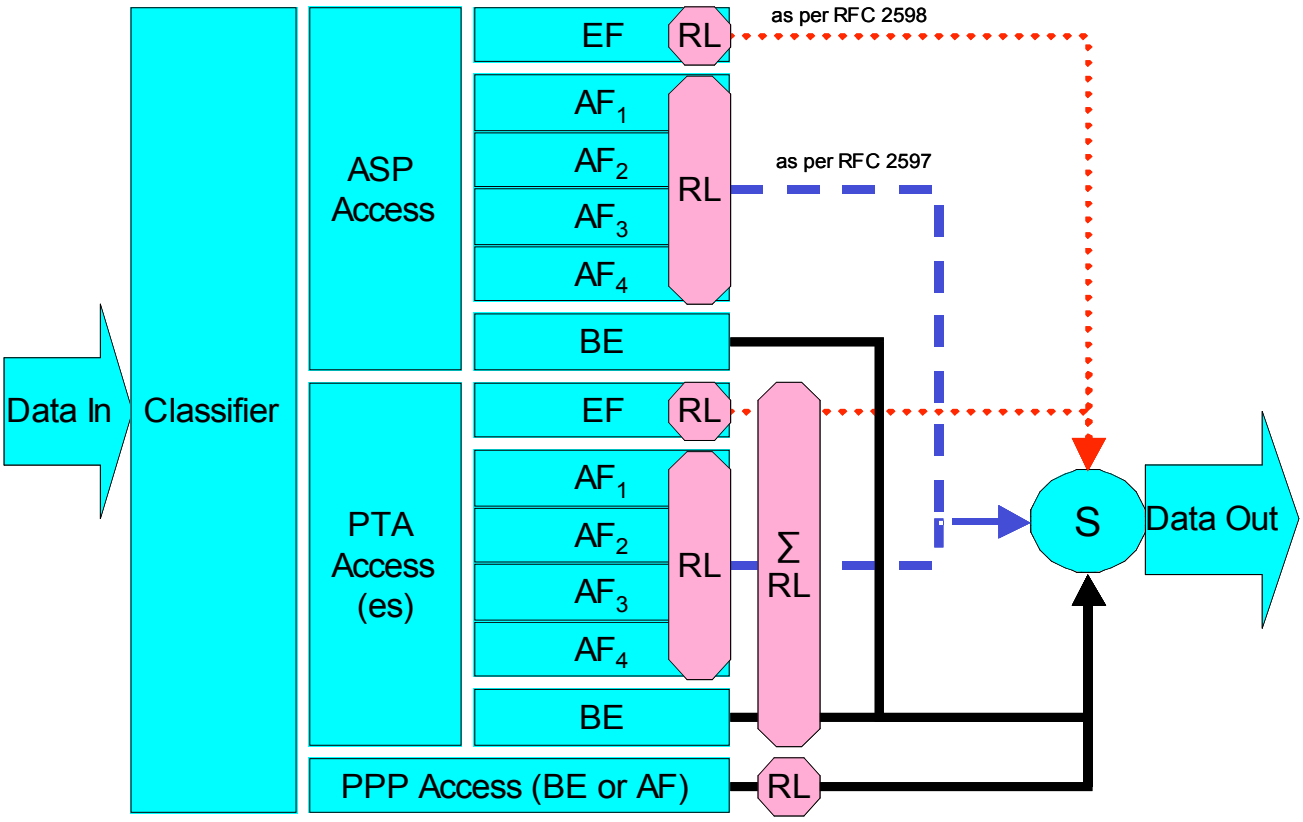

Figure A-2 – Queuing and Scheduling Example for RG

In Figure A-2 the following abbreviations apply:

ASP – Application Service Provider

PTA – PPP Terminated Aggregation

PPP – Point-to-Point Protocol

EF – Expedited Forwarding – as defined in RFC 3246

AF – Assured Forwarding – as defined in RFC 2597

BE – Best Effort forwarding

RL – Rate Limiter

∑RL – Summing Rate Limiter (limits multiple flows)

S – Scheduler

End Inserted Text

2.8 Addition of Appendix with Default Layer 2/3 QoS Mapping

This section specifies a new Appendix to be added to TR-064 that specifies the default Layer 2/3 QoS mapping.

Begin Inserted Text

Appendix B. Default Layer 2/3 QoS Mapping

Table X presents a "default" mapping between layer 2 and layer 3 QoS. In practice, it is a guideline for automatic marking of DSCP (layer 3) based upon EthernetPriority (layer 2) and the other way around. Please refer to the QueueManagement service DSCPMark and EthernetPriorityMark parameters for configuration of a default automatic DSCP/EthernetPriority mapping.

Table X – Default Layer 2/3 QoS Mapping

Layer 2					Layer 3			
ATM	VLAN		WMM / 802.11e		DiffServ		IP Precedence	
Class	VLAN User Priority	Designation	WME Access Category	Designation	DSCP	PHB	IP Precedence (Obsolete)	Designation
UBR	001	BK	AC_BK	Background	000000	Default	000	Routine
	010	spare						
UBR	000	BE	AC_BE	Best Effort	000000 000000	Default CS0	000	Routine
UBR	011	EE	AC_BE	Best Effort	001110 001100 001010 001000	AF13 AF12 AF11 CS1	001	Priority
VBR-nrt	100	CL	AC_VI	Video	010110 010100 010010 010000	AF23 AF22 AF21 CS2	010	Immediate
VBR-nrt	101	VI	AC_VI	Video	011110 011100 011010 011000	AF33 AF32 AF31 CS3	011	Flash
VBR-rt	110	VO	AC_VO	Voice	100110 100100 100010 100000	AF43 AF42 AF41 CS4	100	Flash Override
CBR	110	VO	AC_VO	Voice	101110 101000	EF CS5	101	CRITIC/ECP
CBR	111	NC	AC_VO	Voice	110000 111000	CS6 CS7	110 111	Internetwork Control Network Control

Note: grayed items are added to allow two-way mapping between layer-2 and layer-3 QoS.

End Inserted Text

2.9 Addition of Appendix with URN Definitions

This section specifies a new Appendix to be added to TR-064 that describes the syntax of URNs used for the Protocol Identifier and FlowType variables in the QueueManagement service.

Begin Inserted Text

Appendix C. URN Definitions

C.1ProtocolIdentifier

Table X lists the URNs defined for the ProtocolIdentifier parameter in the App table of the QueueManagement service. Additional standard or vendor-specific URNs may be defined following the standard synax for forming URNs.

Table X – ProtocolIdentifer URNs

URN	Description
urn:dslforum-org:sip	Session Initiaion Protocol (SIP) as defined by RFC 3261.
urn:dslforum-org:h.323	ITU-T Recommendation H.323
urn:dslforum-org:h.248	ITU-T Recommendation H.248 (MEGACO)
urn:dslforum-org:mgcp	Media Gateway Control Protocol (MGCP) as defined by RFC 3435
urn:dslforum-org:pppoe	Bridged sessions of PPPoE

C.2FlowType

A syntax for forming URNs for the FlowType parameter in the Flow table of the QueueManagement service are defined for the Session Description Protocol (SDP) as defined by RFC 2327. Additional standard or vendor-specific URNs may be defined following the standard synax for forming URNs.

A URN to specify an SDP flow is formed as follows:

```
urn:dslforum-org:sdp-[MediaType]-[Transport]
```

[MediaType] corresponds to the “media” sub-field of the “m” field of an SDP session description.

[Transport] corresponds to the “transport” sub-field of the “m” field of an SDP session description.

Non-alphanumeric characters in either field are removed (e.g., “rtp/avp” becomes “rtpavp”).

For example, the following would be valid URNs referring to SDP flows:

```
urn:dslforum-org:sdp-audio-rtpavp
urn:dslforum-org:sdp-video-rtpavp
urn:dslforum-org:sdp-data-udp
```

For FlowType URNs following this convention, there is no defined use for FlowTypeParameters, which should be left empty.

For the ProtocolIdentifier urn:dslforum-org:pppoe, a single flow type is defined referring to the entire PPPoE session. The URL for this FlowType is:

```
urn:dslforum-org:pppoe
```

C.3FlowTypeParameters

For the FlowType urn:dslforum-org:pppoe, the following FlowTypeParameter:

Name	Description of Value
ServiceName	The PPPoE service name. If specified, only bridged PPPoE sessions designated for the named service would be considered part of this flow. If this parameter is not specified, or is empty, bridged PPPoE associated with any service considered part of this flow.
ACName	The PPPoE access concentrator name. If specified, only bridged PPPoE sessions designated for the named access concentrator would be considered part of this flow. If this parameter is not specified, or is empty, bridged PPPoE associated with any access concentrator considered part of this flow.
PPPDomain	The domain part of the PPP username. If specified, only bridged PPPoE sessions in which the domain portion of the PPP username matches this value are considered part of this flow. If this parameter is not specified, or is empty, all bridged PPPoE sessions are considered part of this flow.

End Inserted Text

2.10 Addition of Appendix with Bridging Use Case

This section specifies a new Appendix to be added to TR-064 that describes a use case for the Layer2Bridging capability.

Begin Inserted Text

Appendix D. Layer2Bridging Use Case

D.1 Interface Based Bridging

In an ITU-H.610 architecture using multi-VC and multi-edges to offer multi-services (high speed Internet, TVoDSL, etc.), one VC or a group of VCs are associated with each service. Regarding the CPE, some services may be layer-2 based if the service provider needs to have a layer-2 view of the home devices (for example, set-top boxes). If the services are offered by different service providers, and shared Internet access is also provided via the Internet Gateway, conflict between the local DHCP server and remote DHCP servers can occur. If there is no QoS on the home network there may also be issues regarding the priority of different streams. One solution is to associate one or more physical ports of the Internet Gateway with a specific service associated with one or more VCs.

As an example, Ethernet port 1 may be dedicated to a TVoDSL service and this port would be included in the same bridge with the VCs supporting the TVoDSL service. In this case, the other home network ports would be associated with the shared Internet access service. To achieve this, an interface-based bridge would be created using the Layer2Bridging object. A Bridge table entry would be created along with associated Filter table entries for Ethernet port 1, and each VC associated with the TVoDSL service. In this case no filter criteria would be used in each Filter table entry. If the subscriber's services are modified, the Layer2Bridging configuration may need to be modified accordingly.

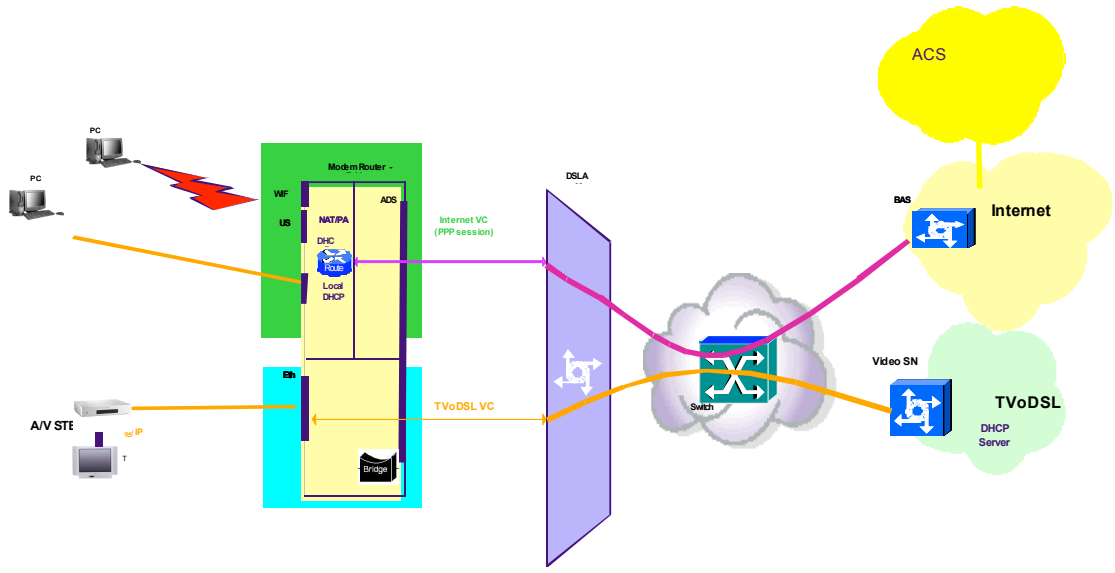

Figure D-1 – Example of interface-based bridging

End Inserted Text

2.11 Updates to Normative References

The Normative References, in section 9 of TR-064 should be updated as shown below.

In this section, items to be added to the original text are shown in red.

Begin Change Text

9 Normative References

[1] UPnP InternetGatewayDevice, Template Versions 1.01, For Universal Plug and Play Version 1.0, November 12, 2001, http://www.upnp.org/download/UPnP_IGD_DCP_v1.zip.

[2] UPnP Device Architecture, Version 1.0.1, May 6, 2003, http://www.upnp.org/download/Clean%20UPnPDA101-20030506.doc.

[3] M. Gudgin, M. Hadley, N. Mendelsohn, J. Moreau, H. Frystyk Nielsen "Simple Object Access Protocol (SOAP) 1.1", http://www.w3.org/TR/2000/NOTE-SOAP-20000508/, W3C Note, May 8, 2000.

[4] UPnP WLAN configuration service http://www.upnp.org/download/WLANConfigurationService%201.0.pdf

[5] HTML 4.01 Specification, http://www.w3.org/TR/html4

End Change Text