

Introduction

This proposal is provided by the editor of IEEE P802.1Qcr, Johannes Specht, for discussion during resolution of rogue comment #101 on 802.1Qcr/D1.0. The major motivation of this comment is to enhance clause 48 for extension by new YANG modules of ongoing and future IEEE 802.1 projects, have a common style across the related contents added by such projects to IEEE 802.1Q, and enhance readability due to more symmetry.

Remarks:

- a) Editor's Notes are used throughout this proposal for commenting observations, make suggestions, describe alternatives, and similar.
- b) Like in IEEE Std 802.1Qcp-2018, there is no strong relationship between the structuring of clause 48.3 of IEEE Std 802.1Qcp-2018 and the structuring found in subsequent clauses of this proposal (see comment #101). However, level 3 subclauses in 48.4 are re-used in this proposal to provide the "glue" in a unified manner. Contents from clause 48.3 are not part of this document, but can be found in IEEE Std 802.1Qcp-2018 and the latest draft of 802.1Qcr.
- c) The subdivision of subsequent clauses is close to IEEE 802.1Qcp-2018, although slightly flattened. Contents related YANG modules `ieee802-dot1q-tpmr`, `ieee802-dot1q-vlan-bridge`, and `ieee802-dot1q-pb` are not level 4 subclauses of an enclosing level 3 sub-clause of the `ieee802-dot1q-bridge`, but are implemented as subsequent level 3 subclauses instead. While the level 3 subclauses in the subsequent shown level 2 subclauses 48.5, 48.6, and 48.7 are symmetrically structured (one level 3 subclause per YANG module), an alternative would be to bundle the level 3 subclauses of a particular module together.
- d) Level 3 subclauses for which the content was not available during creation of this document are marked with "N/A". This content may be provided via maintenance, contributions, or similar.

48. YANG Data Model

48.4 Structure of the YANG model

IEEE 802.1Q YANG models are divided into a number of YANG modules. A summary of the modules contained in this clause is represented in Table 48-1.

Table 48-1—Summary of YANG modules

Module	References	Description
ieee802-types	48.5.1, 48.6.1, 48.7.1	General type definitions used within IEEE 802 standards.
ieee802-dot1q-types	48.5.2, 48.6.2, 48.7.2	General type definitions used by IEEE 802.1Q standard.
ieee802-dot1q-bridge	48.5.3, 48.6.3, 48.7.3	Generic IEEE 802.1Q Bridge YANG model, which is augmented by specific IEEE 802.1Q bridges.
ieee802-dot1q-tpmr	48.5.4, 48.6.4, 48.7.4	Two-Port MAC Relay YANG model, which augments the generic bridge YANG module.
ieee802-dot1q-vlan-bridge	48.5.5, 48.6.5, 48.7.5	Customer VLAN Bridge YANG model, which augments the generic bridge YANG module.
ieee802-dot1q-pb	48.5.6, 48.6.6, 48.7.6	Provider Bridges YANG model, which augments the generic bridge YANG module.
ieee802-dot1q-stream-filters-gates	48.5.7, 48.6.7, 48.7.7	Stream Filters and Stream Gates common to all applications in 8.6.5.4.
ieee802-dot1q-ats	48.5.8, 48.6.8, 48.7.8	ATS-specific extensions to the ieee802-dot1q-stream-filters-gates and ieee802-dot1q-bridge modules.

The relationship between the models listed in clause 48.3 and the YANG modules listed in Table 48-1 is described in the following subclauses.

<<Editor's Note: The new 48.4.x subclauses may be referred to from conformance clauses, etc. The "Notes" column in subsequently shown tables may be used to describe partial implementation (e.g., full implementation of ieee802-types is typically not required). If such details are not needed, the tables could be replaced by lettered lists.>>

48.4.1 Generic Bridge model

The generic Bridge model provides basic bridging capabilities and allows for augmentation by specific YANG models (e.g., Two-Port MAC Relay model, Customer Bridge YANG model, Provider Bridge model).

A system implementing the generic Bridge model implements the YANG modules as described Table 48-3.

Table 48-2—YANG module dependencies for the generic Bridge model

YANG module	Notes
<i>ieee802-types</i>	N/A
<i>ieee802-dot1q-types</i>	N/A
<i>ieee802-dot1q-bridge</i>	N/A

48.4.2 Two-Port MAC Relay model

A system implementing the TPMR YANG model (48.3.2.1) implements the YANG modules as described Table 48-3.

Table 48-3—YANG module dependencies for the Two-Port MAC Relay model

YANG module	Notes
<i>ieee802-types</i>	N/A
<i>ieee802-dot1q-types</i>	N/A
<i>ieee802-dot1q-bridge</i>	N/A
<i>ieee802-dot1q-tpmr</i>	N/A

48.4.3 Customer VLAN Bridge model

A system implementing the Customer VLAN Bridge YANG model (48.3.2.2) implements the YANG modules as described in Table 48-4.

Table 48-4—YANG module dependencies for the Customer VLAN Bridge model

YANG module	Notes
<i>ieee802-types</i>	N/A
<i>ieee802-dot1q-types</i>	N/A
<i>ieee802-dot1q-bridge</i>	N/A
<i>ieee802-dot1q-vlan-bridge</i>	N/A

48.4.4 Provider Bridge model

A system implementing the Provider Bridge YANG model (48.3.2.3) implements the YANG modules as described in Table 48-5.

Table 48-5—YANG module dependencies for the Provider Bridge model

YANG module	Notes
<i>ieee802-types</i>	N/A
<i>ieee802-dot1q-types</i>	N/A
<i>ieee802-dot1q-bridge</i>	N/A
<i>ieee802-dot1q-pb</i>	N/A

48.4.5 Stream Filter and Stream Gates model

The Stream Filter and Stream Gates model (48.3.3) provides basic stream filter (8.6.5.1) and stream gate (8.6.5.2) capabilities and allows for augmentation by specific YANG models (e.g., ATS model).

A system implementing the Stream Filter and Stream Gates model implements the YANG modules as described Table 48-6.

Table 48-6—YANG module dependencies for the Stream Filter and Stream Gates model

YANG module	Notes
<i>ieee802-types</i>	N/A
<i>ieee802-dot1q-types</i>	N/A
<i>ieee802-dot1q-bridge</i>	N/A
<i>ieee802-dot1q-stream-filters-gates</i>	N/A

48.4.6 Asynchronous Traffic Shaping (ATS) model

A system implementing the ATS model (48.3.4) implements the YANG modules as described Table 48-7.

Table 48-7—YANG module dependencies for the Stream Filter and Stream Gates model

YANG module	Notes
<i>ieee802-types</i>	N/A
<i>ieee802-dot1q-types</i>	N/A
<i>ieee802-dot1q-bridge</i>	N/A
<i>ieee802-dot1q-stream-filters-gates</i>	N/A
<i>ieee802-dot1q-ats</i>	N/A

48.5 Relationship to IEEE 802.1Q managed objects

<<Editor's Note: The subsequent introduction copied from IEEE Std 802.1Qcp-2018 and needs adjustments>>

This standard specifies a Unified Modeling Language (UML) [B78] information model and a YANG data model that allows configuration and status reporting for bridges and bridge components including Media Access Control (MAC) Bridges, Two-Port MAC Relays (TPMRs), Customer Virtual Local Area Network (VLAN) Bridges, and Provider Bridges (as specified by this standard) with the capabilities currently specified in 12.4 to 12.8, 12.10, 12.13, and 12.19 of this standard.

In support of this standard, the YANG data model extends the IETF Interface Management YANG model (as specified in IETF RFC 8343).

The Bridge Port YANG node augments the Interface Management YANG model. The specific Bridge (e.g., TPMR, Customer VLAN, Provider Bridge) YANG models are augmentations from the Bridge YANG model. A system implementing these YANG models shall implement the *ieee802-dot1q-bridge*, *ieee802-types*, and *ieee802-dot1q-types* YANG models.

48.5.1 Relationship of the *ieee802-types* YANG module

N/A

<<Editor's Note: The treatment of *ieee802-types* YANG module needs discussion (e.g., scope?!), given that it contains IEEE802-wide types. However, These are simple and we may omit descriptions for these in this clause.>>

48.5.2 Relationship of the *ieee802-dot1q-types* YANG module

N/A

<<Editor's Note: Contents of *ieee802-dot1q-types* are simple, we may omit descriptions for these in this clause. Alternatively (which may even be better), we may craft an explicit text that can be copied into all new level 3 subclauses of this clause (48.5) in case the relationship table is not provided (i.e., stating the criteria, etc.)>>

48.5.3 Relationship of the ieee802-dot1q-bridge YANG module

<<I've made some minor tweaks, however, the base content is copied from multiple tables in IEEE 802.1Qcp-2018. The tweaks are as follows:
 - read/write attributes ("r-w" and "r") removed (already visible at other places and thus less prone to break, MIB tables don't show these either)
 - slight adjustments to the bold cell contents in the left column
 - Indentations simplified>>

Table 48-8—Cross-reference table of the ieee802-dot1q-bridge YANG module

Bridge management information	YANG node(s)
* Bridge	ieee802-dot1q-bridge:bridges:bridge
name (12.4)	name — KEY
address (12.4)	address
type	bridge-type
ports (12.4)	ports
upTime (12.4)	up-time
components (12.3)	components
* Bridge Component	ieee802-dot1q-bridge:bridges:bridge:component
—	name — KEY
id (12.3)	id
type (12.3)	type
address (8.13.8, 13.24)	address
trafficClassEnabled (12.4.1.5.1)	traffic-class-enabled
ports (12.4.1.1.3)	ports
* bridgePorts (—)	* bridge-ports
Bridge Component Capabilities	ieee802-dot1q-bridge:bridges:bridge:component:capabilities
extendedFiltering (12.4.1.5.2)	extended-filtering
trafficClasses (12.4.1.5.2)	traffic-classes
staticEntryIndividualPort (12.4.1.5.2)	static-entry-individual-port
ivlCapable (12.4.1.5.2)	ivl-capable
svlCapable (12.4.1.5.2)	svl-capable
hybridCapable (12.4.1.5.2)	hybrid-capable
configurablePvidTagging (12.4.1.5.2)	configurable-pvid-tagging
localVlanCapable (12.4.1.5.2)	local-vlan-capable

Table 48-8—Cross-reference table of the ieee802-dot1q-bridge YANG module (continued)

Bridge management information	YANG node(s)
Filtering Database	ieee802-dot1q-bridge:bridges:bridge:component:filtering-database
agingTime (12.7, 8.8.3)	aging-time
size (12.7)	size
staticEntries (12.7, 8.8.1)	static-entries
dynamicEntries (12.7, 8.8.3)	dynamic-entries
staticVlanRegistrationEntries (12.7, 8.8.2)	static-vlan-registration-entries
dynamicVlanRegistrationEntries (12.7, 8.8.5)	dynamic-vlan-registration-entries
macAddressRegistrationEntries (12.7, 8.8.4)	mac-address-registration-entries
Filtering Entries	ieee802-dot1q-bridge:bridges:bridge:component:filtering-database:filtering-entries
databaseId (12.7.7)	database-id — KEY
address (12.7.7)	address — KEY
vid (12.7.7)	vid — KEY
entryType (12.7.7)	entry-type
portMap (8.8.1, 8.8.2)	port-map
status (—)	status
VLAN Registration Entries	ieee802-dot1q-bridge:bridges:bridge:component:filtering-database:vlan-registration-entries
databaseId (12.7.7)	database-id — KEY
vid (12.7.7)	vid — KEY
entryType (12.7.7)	entry-type
portMap (8.8.1, 8.8.2)	port-map
Permanent Database	ieee802-dot1q-bridge:bridges:bridge:component:permanent-database
size (12.7.6)	size
staticEntries (12.7.6)	static-entries
staticVlanRegistrationEntries (12.7.6)	static-vlan-registration-entries
Permanent Filtering Entries	ieee802-dot1q-bridge:bridges:bridge:component:permanent-database:filtering-entries
databaseId (12.7.7)	database-id — KEY
address (12.7.7)	address — KEY
vid (12.7.7)	vid — KEY
portMap (8.8.1, 8.8.2)	port-map

Table 48-8—Cross-reference table of the ieee802-dot1q-bridge YANG module (continued)

Bridge management information	YANG node(s)
Bridge VLAN	ieee802-dot1q-bridge:bridges:bridge:component:bridge-vlan
version (12.10.1.3)	version
maxVids (12.10.1.3)	max-vids
overrideDefaultPvid (12.10.1.3)	override-default-pvid
protocolTemplate (12.10.1.7)	protocol-template
maxMsti (12.10.1.7)	max-msti
Bridge VLAN ID Entries	ieee802-dot1q-bridge:bridges:bridge:component:bridge-vlan:vlan-id
vid (12.10.2)	vid — KEY
name (12.10.2)	name
vid (12.10.2)	vid
* untaggedPorts (8.8.2, 12.10.2.1.3)	* untagged-ports
* egressPorts (8.8.10, 12.10.2.1.3)	* egress-ports
Protocol Group Database	ieee802-dot1q-bridge:bridges:bridge:component:bridge-vlan:protocol-group-database
frameFormatType (12.10.1.7)	frame-format-type
protocolGroupId (6.12.2)	protocol-group-id
VID to FID	ieee802-dot1q-bridge:bridges:bridge:component:bridge-vlan:vid-to-fid
vid (12.10.3.4)	vid
fid (12.10.3.4)	fid
VID to FID Allocations	ieee802-dot1q-bridge:bridges:bridge:component:bridge-vlan:vid-to-fid-allocation
vid (12.10.3.2)	vid — KEY
fid (12.10.3.2)	fid
allocationType (12.10.3.2)	allocation-type
FID to VID Allocations	ieee802-dot1q-bridge:bridges:bridge:component:bridge-vlan:fid-to-vid-allocation
fid (12.10.3.3)	fid — KEY
* vid (12.10.3)	* vid
* allocationType (12.10.3)	* allocation-type

Table 48-8—Cross-reference table of the ieee802-dot1q-bridge YANG module (continued)

Bridge management information	YANG node(s)
Bridge MST	ieee802-dot1q-bridge:bridges:bridge:component:bridge-mst
* MSTID (12.12.1)	ieee802-dot1q-bridge:bridges:bridge:component:bridge-mst:mstid
FID to MSTID	ieee802-dot1q-bridge:bridges:bridge:component:bridge-mst:fid-to-mstid
fid (12.12.2)	fid — KEY
mstid (12.12.2)	mstid
FID to MSTID Allocation	ieee802-dot1q-bridge:bridges:bridge:component:bridge-mst:fid-to-mstid-allocation
fids (12.12.2)	fids — KEY
mstid (12.12.2)	mstid

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54

Table 48-8—Cross-reference table of the ieee802-dot1q-bridge YANG module (continued)

Bridge management information	YANG node(s)
Bridge Port	ietf-interfaces:interfaces:interface:ieee802-dot1q-bridge:bridge-port
componentName	component-name
pvid (5.4, 12.10.1)	pvid
defaultPriority (12.6.2)	default-priority
priorityRegenerationTable (12.6.2, 6.9.4)	priority-regeneration-table
pcpSelection (12.6.2, 6.9.3)	pcp-selection
pcpDecodingTable (12.6.2, 6.9.3)	pcp-decoding-table
pcpEncodingTable (12.6.2)	pcp-encoding-table
useDei (12.6.2, 6.9.3)	use-dei
dropEncoding (12.6.2, 8.6.6)	drop-encoding
serviceAccessPrioritySelection (12.6.2, 6.13)	service-access-priority-selection
serviceAccessPriority (12.6.2, 6.13.1)	service-access-priority
trafficClass (11.2.3.2.3, 8.6.6)	traffic-class
acceptableFrame (12.10.1.3, 6.9)	acceptable-frame
enableIngressFiltering (12.10.1.4, 8.6.2)	enable-ingress-filtering
restrictedVlanRegistration (12.10.1.6, 11.2.3.2.3)	enable-restricted-vlan-registration
vidTranslationTable (12.10.1.8, 6.9)	enable-vid-translation-table
egressVidTranslationTable (12.10.1.9, 6.9)	enable-egress-vid-translation-table
protocolGroupId (6.12.2)	protocol-group-id
protocolGroupDatabaseContents (12.10.1.7)	protocol-group-vid-set
adminPointToPoint (6.8.2, 12.4.2)	admin-point-to-point
* vidTranslations (12.10.1.8, 6.9)	* vid-translations
* egressVidTranslations (12.10.1.9, 6.9)	* egress-vid-translations
protocolBasedVlanClassification (5.4.1.2)	protocol-based-vlan-classification
maxVidSetEntries (12.10.1.1.3)	max-vid-set-entries
portNumber (13.25, 12.4.2)	port-number
portType (12.4.2.1)	port-type
address (12.4.2)	address
capabilities (12.4.2, 12.10.1.1.3)	capabilities
typeCapabilities (12.4.2)	type-capabilities
external (12.4.2)	external
operPointToPoint (12.4.2)	oper-point-to-point
mediaDependentOverhead (12.4.2)	media-dependent-overhead

Table 48-8—Cross-reference table of the ieee802-dot1q-bridge YANG module (continued)

Bridge management information	YANG node(s)
Bridge Port Statistics	ietf-interfaces:interfaces:interface:ieee802-dot1q-bridge:bridge-port:statistics
delayExceededDiscard (12.6.1.1.3, 8.6.6)	delay-exceeded-discard
mtuExceededDiscards (12.6.1.1.3)	mtu-exceeded-discards
frameRx (12.6.1.1.3)	frame-rx
octetsRx (12.6.1.1.3)	octets-rx
frameTx ()	frame-tx
octetsTx ()	octets-tx
discardInbound (12.6.1.1.3)	discard-inbound
forwardOutbound (12.6.1.1.3)	forward-outbound
discardLackOfBuffers (12.6.1.1.3)	discard-lack-of-buffers
discardTransitDelayExceeded (12.6.1.1.3)	discard-transit-delay-exceeded
discardOnError (12.6.1.1.3)	discard-on-error
discardOnIngressFiltering (12.6.1.1.3)	discard-on-ingress-filtering

48.5.4 Relationship of the ieee802-dot1q-tpmr YANG module

N/A

48.5.5 Relationship of the ieee802-vlan-bridge YANG module

N/A

48.5.6 Relationship of the ieee802-dot1q-pb YANG module

N/A

48.5.7 Relationship of the ieee802-dot1q-stream-filters-gates YANG module

Table 48-13—Cross-reference table of the ieee802-dot1q-stream-filters-gates YANG module

Bridge management information	YANG node(s)
Stream Filters	ieee802-dot1q-bridge:bridges:bridge:component:stream-filters
MaxStreamFilterInstances (12.31.1.1)	max-stream-filter-instances
Stream Filter Instance Table (Table 12-30)	ieee802-dot1q-bridge:bridges:bridge:component:stream-filters:stream-filter-instance-table
StreamFilterInstance (12.31.2.1)	stream-filter-instance-id—KEY
StreamHandleSpec (12.31.2.2)	stream-handle-spec
PrioritySpec (12.31.2.3)	priority-spec
StreamGateInstanceID (Table 12-30)	stream-gate-ref
FilterSpecificationList (12.31.2.5)	ieee802-dot1q-bridge:bridges:bridge:component:stream-filters:stream-filter-instance-table:filter-specification-list
—	index—KEY
Maximum SDU Size Filters	ieee802-dot1q-bridge:bridges:bridge:component:stream-filters:stream-filter-instance-table:filter-specification-list:maximum-sdu-size
An Integer value representing a Maximum SDU size (12.31.2.5, item a)	maximum-sdu-size
StreamBlockedDueToOversizeFrameEnable (12.31.2)	stream-blocked-due-to-oversize-frame-enabled
StreamBlockedDueToOversizeFrame (12.31.2)	stream-blocked-due-to-oversize-frame
Stream Gates	ieee802-dot1q-bridge:bridges:bridge:component:stream-gates
MaxStreamGateInstances (12.31.1.2)	max-stream-gate-instances
Stream Gate Instance Table (Table 12-31)	ieee802-dot1q-bridge:bridges:bridge:component:stream-gates:stream-gate-instance-table
StreamGateInstance (12.31.2.4)	stream-gate-instance-id—KEY
StreamGateEnabled (Table 12-31)	stream-handle-spec
PrioritySpec (Table 12-31,12.31.2.3)	priority-spec
AdminGateStates (Table 12-31)	admin-gate-states
AdminIPV (12.31.2.5)	admin-ipv

48.5.8 Relationship of the ieee802-dot1q-ats YANG module

Table 48-14—Cross-reference table of the ieee802-dot1q-ats YANG module

Generic bridge management information	YANG node(s)
ATS Schedulers	ieee802-dot1q-bridge:bridges:bridge:component:schedulers
MaxSchedulerInstances (12.31.1.5)	max-scheduler-instances
Scheduler Instance Table (Table 12-33)	ieee802-dot1q-bridge:bridges:bridge:component:schedulers:scheduler-instance-table
SchedulerInstanceID (12.31.5.1)	scheduler-instance-id—KEY
CommittedBurstSize (12.31.5.2)	committed-burst-size
CommittedInformationRate (12.31.5.3)	committed-information-rate
SchedulerGroupInstanceID (12.31.5.4)	scheduler-group-ref
ATS Scheduler Groups	ieee802-dot1q-bridge:bridges:bridge:component:scheduler-groups
MaxSchedulerGroupInstances (12.31.1.6)	max-scheduler-group-instances
Scheduler Group Instance Table (Table 12-34)	ieee802-dot1q-bridge:bridges:bridge:component:scheduler-groups:scheduler-group-instance-table
SchedulerGroupInstanceID (12.31.6.1)	scheduler-group-instance-id—KEY
MaxResidenceTime (12.31.6.2)	max-residence-time
ATS Scheduler Timing Characteristics Table (Table 12-36)	ieee802-dot1q-bridge:bridges:bridge:component:scheduler-timing-characteristics:scheduler-timing-characteristics-table
ReceptionPortNumber (12.31.8.1)	reception-port—KEY
TransmissionPortNumber (12.31.8.2)	transmission-port—KEY
ClockOffsetVariationMax (12.31.8.3)	clock-offset-variation-max
ClockRateDeviationMax (12.31.8.4)	clock-rate-deviation-max
ArrivalRecognitionDelayMax (12.31.8.5)	arrival-recognition-delay-max
ProcessingDelayMin (12.31.8.6)	processing-delay-min
ProcessingDelayMax (12.31.8.7)	processing-delay-max

48.6 YANG data scheme tree definitions

A simplified graphical representation of the data model is used in this document. The meaning of the symbols in these diagrams is as follows:

- Brackets “[” and “]” enclose list keys.
- Abbreviations before data node names: “rw” means configuration (read-write), and “ro” means state data (read-only).
- Symbols after data node names: “?” means an optional node, “!” means a presence container, and “*” denotes a list and leaf-list.
- Parentheses enclose choice and case nodes, and case nodes are also marked with a colon (“:”).

Ellipsis (“...”) stands for contents of subtrees that are not shown.

48.6.1 Data scheme definition for the ieee802-types YANG module

N/A

48.6.2 Data scheme definition for the ieee802-dot1q-types YANG module

N/A

48.6.3 Data scheme definition for the ieee802-dot1q-bridge YANG module

```
module: ieee802-dot1q-bridge
  +--rw bridges
    +--rw bridge* [name]
      +--rw name          dot1qtypes:name-type
      +--rw address       ieee:mac-address
      +--rw bridge-type   identityref
      +--ro ports?        uint16
      +--ro up-time?      yang:zero-based-counter32
      +--ro components?   uint32
      +--rw component* [name]
        +--rw name          string
        +--rw id?           uint32
        +--rw type          identityref
        +--rw address?      ieee:mac-address
        +--rw traffic-class-enabled? boolean
        +--ro ports?       uint16
        +--ro bridge-port* if:interface-ref
        +--ro capabilities
          | +--ro extended-filtering?    boolean
          | +--ro traffic-classes?       boolean
          | +--ro static-entry-individual-port? boolean
          | +--ro ivl-capable?           boolean
          | +--ro svl-capable?           boolean
          | +--ro hybrid-capable?       boolean
          | +--ro configurable-pvid-tagging? boolean
          | +--ro local-vlan-capable?    boolean
        +--rw filtering-database
          | +--rw aging-time?            uint32
          | +--ro size?                  yang:gauge32
          | +--ro static-entries?        yang:gauge32
          | +--ro dynamic-entries?       yang:gauge32
          | +--ro static-vlan-registration-entries? yang:gauge32
          | +--ro dynamic-vlan-registration-entries? yang:gauge32
          | +--ro mac-address-registration-entries? yang:gauge32 {extended-
filtering-services}?
          | +--rw filtering-entry* [database-id vids address]
          | | +--rw database-id          uint32
          | | +--rw address              ieee:mac-address
          | | +--rw vids                  dot1qtypes:vid-range-type
```

```

1      | | +--rw entry-type?    enumeration
2      | | +--rw port-map* [port-ref]
3      | | | +--rw port-ref                                port-number-type
4      | | | +--rw (map-type)?
5      | | | | +---:(static-filtering-entries)
6      | | | | | +--rw static-filtering-entries
7      | | | | | | +--rw control-element?    enumeration
8      | | | | | | +--rw connection-identifier? port-number-type
9      | | | | +---:(static-vlan-registration-entries)
10     | | | | | +--rw static-vlan-registration-entries
11     | | | | | | +--rw registrar-admin-control? enumeration
12     | | | | | | +--rw vlan-transmitted?    enumeration
13     | | | | +---:(mac-address-registration-entries)
14     | | | | | +--rw mac-address-registration-entries
15     | | | | | | +--rw control-element?    enumeration
16     | | | | +---:(dynamic-vlan-registration-entries)
17     | | | | | +--rw dynamic-vlan-registration-entries
18     | | | | | | +--rw control-element?    enumeration
19     | | | | +---:(dynamic-reservation-entries)
20     | | | | | +--rw dynamic-reservation-entries
21     | | | | | | +--rw control-element?    enumeration
22     | | | | +---:(dynamic-filtering-entries)
23     | | | | | +--rw dynamic-filtering-entries
24     | | | | | | +--rw control-element?    enumeration
25     | | | +--ro status?    enumeration
26     | | +--rw vlan-registration-entry* [database-id vids]
27     | | | +--rw database-id    uint32
28     | | | +--rw vids          dot1qtypes:vid-range-type
29     | | | +--rw entry-type?    enumeration
30     | | | +--rw port-map* [port-ref]
31     | | | | +--rw port-ref                                port-number-type
32     | | | | +--rw (map-type)?
33     | | | | | +---:(static-filtering-entries)
34     | | | | | | +--rw static-filtering-entries
35     | | | | | | | +--rw control-element?    enumeration
36     | | | | | | | +--rw connection-identifier? port-number-type
37     | | | | | +---:(static-vlan-registration-entries)
38     | | | | | | +--rw static-vlan-registration-entries
39     | | | | | | | +--rw registrar-admin-control? enumeration
40     | | | | | | | +--rw vlan-transmitted?    enumeration
41     | | | | | +---:(mac-address-registration-entries)
42     | | | | | | +--rw mac-address-registration-entries
43     | | | | | | | +--rw control-element?    enumeration
44     | | | | | +---:(dynamic-vlan-registration-entries)
45     | | | | | | +--rw dynamic-vlan-registration-entries
46     | | | | | | | +--rw control-element?    enumeration
47     | | | | | +---:(dynamic-reservation-entries)
48     | | | | | | +--rw dynamic-reservation-entries
49     | | | | | | | +--rw control-element?    enumeration
50     | | | | | +---:(dynamic-filtering-entries)
51     | | | | | | +--rw dynamic-filtering-entries
52     | | | | | | | +--rw control-element?    enumeration
53     | | +--rw permanent-database
54     | | | +--ro size?                                yang:gauge32
55     | | | +--ro static-entries?                    yang:gauge32
56     | | | +--ro static-vlan-registration-entries? yang:gauge32
57     | | | +--rw filtering-entry* [database-id vids address]
58     | | | | +--rw database-id    uint32
59     | | | | +--rw address        ieee:mac-address
60     | | | | +--rw vids          dot1qtypes:vid-range-type
61     | | | | +--ro status?        enumeration
62     | | | | +--rw port-map* [port-ref]
63     | | | | | +--rw port-ref                                port-number-type
64     | | | | | +--rw (map-type)?
65     | | | | | | +---:(static-filtering-entries)
66     | | | | | | | +--rw static-filtering-entries
67     | | | | | | | | +--rw control-element?    enumeration
68     | | | | | | | | +--rw connection-identifier? port-number-type
69     | | | | | +---:(static-vlan-registration-entries)
70     | | | | | | +--rw static-vlan-registration-entries
71     | | | | | | | +--rw registrar-admin-control? enumeration
72     | | | | | | | +--rw vlan-transmitted?    enumeration

```

```

1      |           +---:(mac-address-registration-entries)
2      |           | +---rw mac-address-registration-entries
3      |           |           +---rw control-element?  enumeration
4      |           +---:(dynamic-vlan-registration-entries)
5      |           | +---rw dynamic-vlan-registration-entries
6      |           |           +---rw control-element?  enumeration
7      |           +---:(dynamic-reservation-entries)
8      |           | +---rw dynamic-reservation-entries
9      |           |           +---rw control-element?  enumeration
10     +---rw bridge-vlan
11     | +---ro version?                uint16
12     | +---ro max-vids?                uint16
13     | +---ro override-default-pvid?   boolean
14     | +---ro protocol-template?       dot1qtypes:protocol-frame-format-
type {port-and-protocol-based-vlan}?
15     | +---ro max-msti?                uint16
16     | +---rw vlan* [vid]
17     | | +---rw vid                    dot1qtypes:vlan-index-type
18     | | +---rw name?                  dot1qtypes:name-type
19     | | +---ro untagged-ports*        if:interface-ref
20     | | +---ro egress-ports*          if:interface-ref
21     | | +---rw protocol-group-database* [db-index] {port-and-protocol-
based-vlan}?
22     | | +---rw db-index                uint16
23     | | +---rw frame-format-type?     dot1qtypes:protocol-frame-format-
type
24     | | | +---rw (frame-format)?
25     | | | | +---:(ethernet-rfc1042-snap8021H)
26     | | | | +---rw ethertype?        dot1qtypes:ethertype-type
27     | | | | +---:(snap-other)
28     | | | | +---rw protocol-id?      string
29     | | | | +---:(llc-other)
30     | | | | +---rw dsap-ssap-pairs
31     | | | | +---rw llc-address?      string
32     | | | +---rw group-id?            uint32
33     | | +---rw vid-to-fid-allocation* [vids]
34     | | | +---rw vids                  dot1qtypes:vid-range-type
35     | | | +---ro fid?                  uint32
36     | | | +---ro allocation-type?     enumeration
37     | | +---rw fid-to-vid-allocation* [fid]
38     | | | +---rw fid                    uint32
39     | | | +---ro allocation-type?     enumeration
40     | | | +---ro vid*                  dot1qtypes:vlan-index-type
41     | | +---rw vid-to-fid* [vid]
42     | | | +---rw vid                    dot1qtypes:vlan-index-type
43     | | | +---rw fid?                  uint32
44     +---rw bridge-mst
45     | +---rw mstid*                    dot1qtypes:mstid-type
46     | +---rw fid-to-mstid* [fid]
47     | | +---rw fid                    uint32
48     | | +---rw mstid?                  dot1qtypes:mstid-type
49     | +---rw fid-to-mstid-allocation* [fids]
50     | | +---rw fids                    dot1qtypes:vid-range-type
51     | | +---rw mstid?                  dot1qtypes:mstid-type
52     augment /if:interfaces/if:interface:
53     +---rw bridge-port
54     | +---rw component-name?          string
55     | +---rw port-type?                identityref
56     | +---rw pvid?                      dot1qtypes:vlan-index-type
57     | +---rw default-priority?         dot1qtypes:priority-type
58     | +---rw priority-regeneration
59     | | +---rw priority0?              priority-type
60     | | +---rw priority1?              priority-type
61     | | +---rw priority2?              priority-type
62     | | +---rw priority3?              priority-type
63     | | +---rw priority4?              priority-type
64     | | +---rw priority5?              priority-type
65     | | +---rw priority6?              priority-type
66     | | +---rw priority7?              priority-type

```



```
1      +---rw pcp-selection?                               dot1qtypes:pcp-selection-type
2      +---rw pcp-decoding-table
3      |   +---rw pcp-decoding-map* [pcp]
4      |   |   +---rw pcp                               pcp-selection-type
5      |   |   +---rw priority-map* [priority-code-point]
6      |   |   |   +---rw priority-code-point           priority-type
7      |   |   |   +---rw priority?                     priority-type
8      |   |   |   +---rw drop-eligible?                 boolean
9      |   +---rw pcp-encoding-table
10      |   |   +---rw pcp-encoding-map* [pcp]
11      |   |   |   +---rw pcp                           pcp-selection-type
12      |   |   |   +---rw priority-map* [priority dei]
13      |   |   |   |   +---rw priority                 priority-type
14      |   |   |   |   +---rw dei                       boolean
15      |   |   |   |   +---rw priority-code-point?      priority-type
16      |   +---rw use-dei?                               boolean
17      |   +---rw drop-encoding?                         boolean
18      |   +---rw service-access-priority-selection?     boolean
19      |   +---rw service-access-priority
20      |   |   +---rw priority0?   priority-type
21      |   |   +---rw priority1?   priority-type
22      |   |   +---rw priority2?   priority-type
23      |   |   +---rw priority3?   priority-type
24      |   |   +---rw priority4?   priority-type
25      |   |   +---rw priority5?   priority-type
26      |   |   +---rw priority6?   priority-type
27      |   |   +---rw priority7?   priority-type
28      |   +---rw traffic-class
29      |   |   +---rw traffic-class-map* [priority]
30      |   |   |   +---rw priority                 priority-type
31      |   |   |   +---rw available-traffic-class* [num-traffic-class]
32      |   |   |   |   +---rw num-traffic-class   uint8
33      |   |   |   |   +---rw traffic-class?      traffic-class-type
34      |   +---rw acceptable-frame?                 enumeration
35      |   +---rw enable-ingress-filtering?          boolean
36      |   +---rw enable-restricted-vlan-registration? boolean
37      |   +---rw enable-vid-translation-table?      boolean
38      |   +---rw enable-egress-vid-translation-table? boolean
39      |   +---rw protocol-group-vid-set* [group-id] {port-and-protocol-based-vlan}?
40      |   |   +---rw group-id   uint32
41      |   |   +---rw vid*       dot1qtypes:vlanid
42      |   +---rw admin-point-to-point?             enumeration
43      |   +---ro protocol-based-vlan-classification? boolean {port-and-protocol-
44      |   based-vlan}?
45      |   +---ro max-vid-set-entries?                uint16 {port-and-protocol-
46      |   based-vlan}?
47      |   +---ro port-number?                        dot1qtypes:port-number-type
48      |   +---ro address?                            ieee:mac-address
49      |   +---ro capabilities?                       bits
50      |   +---ro type-capabilities?                 bits
51      |   +---ro external?                           boolean
52      |   +---ro oper-point-to-point?                boolean
53      |   +---ro media-dependent-overhead?         uint8
54      |   +---ro statistics
55      |   |   +---ro delay-exceeded-discards?        yang:counter64
56      |   |   +---ro mtu-exceeded-discards?          yang:counter64
57      |   |   +---ro frame-rx?                       yang:counter64
58      |   |   +---ro octets-rx?                      yang:counter64
59      |   |   +---ro frame-tx?                       yang:counter64
60      |   |   +---ro octets-tx?                      yang:counter64
61      |   |   +---ro discard-inbound?                 yang:counter64
62      |   |   +---ro forward-outbound?               yang:counter64
63      |   |   +---ro discard-lack-of-buffers?         yang:counter64
64      |   |   +---ro discard-transit-delay-exceeded? yang:counter64
65      |   |   +---ro discard-on-error?                yang:counter64
66      |   |   +---ro discard-on-ingress-filtering?    yang:counter64 {ingress-
67      |   filtering}?
68      |   +---rw vid-translations* [local-vid]
69      |   |   +---rw local-vid   dot1qtypes:vlanid
70      |   |   +---rw relay-vid?  dot1qtypes:vlanid
71      |   +---rw egress-vid-translations* [relay-vid]
72      |   |   +---rw relay-vid   dot1qtypes:vlanid
```

1 +--rw local-vid? dot1qtypes:vlanid

48.6.4 Data scheme definition for the ieee802-dot1q-tpmr YANG module

```
2
3
4
5 module: ieee802-dot1q-tpmr
6   augment /if:interfaces/if:interface/dot1q:bridge-port:
7     +--rw managed-address?       boolean
8     +--rw mac-status-propagation
9     +--rw link-notify?           boolean
10    +--rw link-notify-wait?       yang:timeticks
11    +--rw link-notify-retry?      yang:timeticks
12    +--rw mac-notify?            boolean
13    +--rw mac-notify-time?       yang:timeticks
14    +--rw mac-recover-time?      yang:timeticks
15  augment /if:interfaces/if:interface/dot1q:bridge-port/dot1q:statistics:
16    +--ro acks-tx?                yang:counter64
17    +--ro add-notificatons-tx?    yang:counter64
18    +--ro loss-notification-tx?   yang:counter64
19    +--ro loss-confirmation-tx?   yang:counter64
20    +--ro acks-rx?                yang:counter64
21    +--ro add-notificatons-rx?    yang:counter64
22    +--ro loss-notification-rx?   yang:counter64
23    +--ro loss-confirmation-rx?   yang:counter64
24    +--ro add-events?             yang:counter64
25    +--ro loss-events?            yang:counter64
26    +--ro mac-status-notifications? yang:counter64
```

48.6.5 Data scheme definition for the ieee802-dot1q-vlan-bridge YANG module

27 N/A

48.6.6 Data scheme definition for the ieee802-dot1q-pb YANG module

```
28
29 module: ieee802-dot1q-pb
30   augment /if:interfaces/if:interface/dot1q:bridge-port:
31     +--rw svid?                   dot1qtypes:vlanid
32     +--rw cvid-registration* [cvid]
33     |   +--rw cvid                dot1qtypes:vlanid
34     |   +--rw svid?               dot1qtypes:vlanid
35     |   +--rw untagged-pep?       boolean
36     |   +--rw untagged-cep?       boolean
37     +--rw service-priority-regeneration* [svid]
38     |   +--rw svid                dot1qtypes:vlanid
39     |   +--rw priority-regeneration
40     |   |   +--rw priority0?      priority-type
41     |   |   +--rw priority1?      priority-type
42     |   |   +--rw priority2?      priority-type
43     |   |   +--rw priority3?      priority-type
44     |   |   +--rw priority4?      priority-type
45     |   |   +--rw priority5?      priority-type
46     |   |   +--rw priority6?      priority-type
47     |   |   +--rw priority7?      priority-type
48     +--rw rcap-internal-interface* [external-svid]
49     |   +--rw external-svid        dot1qtypes:vlanid
50     |   +--rw internal-port-number? dot1qtypes:port-number-type
51     |   +--rw internal-svid?       dot1qtypes:vlanid
52     |   +--rw internal-interface-type? enumeration
```

48.6.7 Data scheme definition for the ieee802-dot1q-stream-filters-gates YANG module

```
53
54 module: ieee802-dot1q-stream-filters-gates
55   augment /dot1q:bridges/dot1q:bridge/dot1q:component:
56     +--rw stream-filters
57     |   +--rw stream-filter-instance-table* [stream-filter-instance-id]
58     |   |   +--rw stream-filter-instance-id   uint32
59     |   |   +--rw (stream-handle-spec)?
```

```
1      | | | +--:(wildcard)
2      | | | | +--rw wildcard?          empty
3      | | | +--:(stream-handle)
4      | | |   +--rw stream-handle      uint32
5      | | +--rw priority-spec          ipv-type
6      | | +--rw stream-gate-ref        stream-gate-ref
7      | | +--rw filter-specification-list* [index]
8      | |   +--rw index                uint8
9      | |   +--rw (filter-specification)?
10     | |     +--:(maximum-sdu-size)
11     | |       +--rw maximum-sdu-size  uint32
12     | |       +--rw stream-blocked-due-to-oversize-frame-enabled?  boolean
13     | |       +--ro stream-blocked-due-to-oversize-frame?          boolean
14     | +--ro max-stream-filter-instances?  uint32
15     +--rw stream-gates
16         +--rw stream-gate-instance-table* [stream-gate-instance-id]
17             | +--rw stream-gate-instance-id  uint32
18             | +--rw gate-enable?            boolean
19             | +--rw admin-gate-states?      gate-state-value-type
20             | +--rw admin-ipv?              ipv-type
21         +--ro max-stream-gate-instances?    uint32
22
```

48.6.8 Data scheme definition for the ieee802-dot1q-ats YANG module

```
23
24
25 module: ieee802-dot1q-ats
26   augment /dot1q:bridges/dot1q:bridge/dot1q:component
27     /sfsg:stream-filters/sfsg:stream-filter-instance-table
28     /sfsg:filter-specification-list/sfsg:filter-specification:
29     +--:(scheduler-ref)
30     +--rw scheduler-ref      ats:scheduler-ref-type
31   augment /if:interfaces/if:interface/dot1q:bridge-port:
32     +--rw ats-port-parameters
33     +--ro discarded-frames-count?  yang:counter64
34   augment /dot1q:bridges/dot1q:bridge/dot1q:component:
35     +--rw schedulers
36         | +--rw scheduler-instance-table* [scheduler-instance-id]
37         | | +--rw scheduler-instance-id      uint32
38         | | +--rw committed-information-rate  uint64
39         | | +--rw committed-burst-size       uint32
40         | | +--rw scheduler-group-ref        ats:scheduler-group-ref-type
41         | +--ro max-scheduler-instances?    uint32
42     +--rw scheduler-groups
43         +--rw scheduler-group-instance-table* [scheduler-group-instance-id]
44         | +--rw scheduler-group-instance-id  uint32
45         | +--rw max-residence-time          uint32
46         +--ro max-scheduler-group-instances?  uint32
47         +--rw scheduler-timing-characteristics
48         +--ro scheduler-timing-characteristics-table* [reception-port
49         transmission-port]
49         +--ro reception-port                dot1qtypes:port-number-type
50         +--ro transmission-port             dot1qtypes:port-number-type
51         +--ro clock-offset-variation-max    uint32
52         +--ro clock-rate-deviation-max     uint32
53         +--ro arrival-recognition-delay-max uint32
54         +--ro processing-delay-min          uint32
55         +--ro processing-delay-max         uint32
```

48.7 YANG modules

<<Editor's Note: Level 3 subclause names are symmetrically phrased to those for the MIB (clause 17.7 in IEEE Std 802.1Q-2018)>>

48.7.1 Definitions for the ieee802-types YANG module

```
module ieee802-types {
  namespace urn:ieee:std:802.1Q:yang:ieee802-types;
  prefix ieee;
  organization
    "IEEE 802.1 Working Group";
  contact
    "WG-URL: http://www.ieee802.org/1/
    WG-EMail: stds-802-1-L@ieee.org

    Contact: IEEE 802.1 Working Group Chair
    Postal: C/O IEEE 802.1 Working Group
           IEEE Standards Association
           445 Hoes Lane
           P.O. Box 1331
           Piscataway
           NJ 08854
           USA

    E-mail: STDS-802-1-L@IEEE.ORG";
  description
    "This module contains a collection of generally useful derived
    data types for IEEE YANG models.";
  revision 2018-03-07 {
    description
      "Published as part of IEEE Std 802.1Q-2018.
      Initial version.";
    reference
      "IEEE Std 802.1Q-2018, Bridges and Bridged Networks.";
  }
  typedef mac-address {
    type string {
      pattern "[0-9a-fA-F]{2}(-[0-9a-fA-F]{2}){5}";
    }
    description
      "The mac-address type represents a MAC address in the canonical
      format and hexadecimal format specified by IEEE Std 802. The
      hexadecimal representation uses uppercase characters.";
    reference
      "3.1 of IEEE Std 802-2014
      8.1 of IEEE Std 802-2014";
  }
}
```

48.7.2 Definitions for the ieee802-dot1q-types YANG module

```
module ieee802-dot1q-types {
  namespace urn:ieee:std:802.1Q:yang:ieee802-dot1q-types;
  prefix dot1q-types;
  import ietf-yang-types {
    prefix yang;
  }
  organization
    "IEEE 802.1 Working Group";
  contact
    "WG-URL: http://www.ieee802.org/1/
    WG-EMail: stds-802-1-L@ieee.org

    Contact: IEEE 802.1 Working Group Chair
    Postal: C/O IEEE 802.1 Working Group
           IEEE Standards Association
           445 Hoes Lane
```

```
1           P.O. Box 1331
2           Piscataway
3           NJ 08854
4           USA
5
6           E-mail: STDS-802-1-L@IEEE.ORG";
7 description
8 "Common types used within dot1q-bridge modules.";
9 revision 2018-03-07 {
10     description
11     "Published as part of IEEE Std 802.1Q-2018.
12     Initial version.";
13     reference
14     "IEEE Std 802.1Q-2018, Bridges and Bridged Networks.";
15 }
16
17 identity dot1q-vlan-type {
18     description
19     "Base identity from which all 802.1Q VLAN tag types are derived
20     from.";
21 }
22 identity c-vlan {
23     base dot1q-vlan-type;
24     description
25     "An 802.1Q Customer VLAN, using the 81-00 EtherType";
26     reference
27     "5.5 of IEEE Std 802.1Q-2018";
28 }
29 identity s-vlan {
30     base dot1q-vlan-type;
31     description
32     "An 802.1Q Service VLAN, using the 88-A8 EtherType originally
33     introduced in 802.1ad, and incorporated into 802.1Q (2011)";
34     reference
35     "5.6 of IEEE Std 802.1Q-2018";
36 }
37
38 typedef name-type {
39     type string {
40         length "0..32";
41     }
42     description
43     "A text string of up to 32 characters, of locally determined
44     significance.";
45 }
46
47 typedef port-number-type {
48     type uint32 {
49         range "1..65535";
50     }
51     description
52     "The port number of the Bridge port for which this entry
53     contains Bridge management information.";
54 }
55
56 typedef priority-type {
57     type uint8 {
58         range "0..7";
59     }
60     description
61     "A range of priorities from 0 to 7 (inclusive). The Priority
62     Code Point (PCP) is a 3-bit field that refers to the class of
63     service associated with an 802.1Q VLAN tagged frame. The field
64     specifies a priority value between 0 and 7, these values can be
65     used by quality of service (QoS) to prioritize different classes
66     of traffic.";
67 }
68
69 typedef vid-range-type {
70     type string {
71         pattern
72         "([1-9]"
73         "[0-9]{0,3}"
74         "(-[1-9][0-9]{0,3})?"
75         "(,[1-9][0-9]{0,3}(-[1-9][0-9]{0,3})?)*)";
76     }
77 }
```

```
1      description
2      "A list of VLAN Ids, or non overlapping VLAN ranges, in
3      ascending order, between 1 and 4094.
4
5      This type is used to match an ordered list of VLAN Ids, or
6      contiguous ranges of VLAN Ids. Valid VLAN Ids must be in the
7      range 1 to 4094, and included in the list in non overlapping
8      ascending order.
9
10     For example: 1,10-100,250,500-1000";
11 }
12 typedef vlanid {
13     type uint16 {
14         range "1..4094";
15     }
16     description
17     "The vlanid type uniquely identifies a VLAN. This is the 12-bit
18     VLAN-ID used in the VLAN Tag header. The range is defined by the
19     referenced specification. This type is in the value set and its
20     semantics equivalent to the VlanId textual convention of the
21     SMiv2.";
22 }
23 typedef vlan-index-type {
24     type uint32 {
25         range "1..4094 | 4096..4294967295";
26     }
27     description
28     "A value used to index per-VLAN tables. Values of 0 and 4095 are
29     not permitted. The range of valid VLAN indices. If the value is
30     greater than 4095, then it represents a VLAN with scope local to
31     the particular agent, i.e., one without a global VLAN-ID
32     assigned to it. Such VLANs are outside the scope of IEEE 802.1Q,
33     but it is convenient to be able to manage them in the same way
34     using this YANG module.";
35     reference
36     "9.6 of IEEE Std 802.1Q-2018";
37 }
38 typedef mstid-type {
39     type uint32 {
40         range "1..4094";
41     }
42     description
43     "In an MSTP Bridge, an MSTID, i.e., a value used to identify a
44     spanning tree (or MST) instance";
45     reference
46     "13.8 of IEEE Std 802.1Q-2018";
47 }
48 typedef pcp-selection-type {
49     type enumeration {
50         enum 8POD {
51             description
52             "8 priorities, 0 drop eligible";
53         }
54         enum 7P1D {
55             description
56             "7 priorities, 1 drop eligible";
57         }
58         enum 6P2D {
59             description
60             "6 priorities, 2 drop eligible";
61         }
62         enum 5P3D {
63             description
64             "5 priorities, 3 drop eligible";
65         }
66     }
67     description
68     "Priority Code Point selection types.";
69     reference
70     "12.6.2.5.3 of IEEE Std 802.1Q-2018
71     6.9.3 of IEEE Std 802.1Q-2018";
72 }
```

```
1     typedef protocol-frame-format-type {
2         type enumeration {
3             enum Ethernet {
4                 description
5                     "Ethernet frame format";
6             }
7             enum rfc1042 {
8                 description
9                     "RFC 1042 frame format";
10            }
11            enum snap8021H {
12                description
13                    "SNAP 802.1H frame format";
14            }
15            enum snapOther {
16                description
17                    "Other SNAP frame format";
18            }
19            enum llcOther {
20                description
21                    "Other LLC frame format";
22            }
23            description
24                "A value representing the frame format to be matched.";
25            reference
26                "12.10.1.7.1 of IEEE Std 802.1Q-2018";
27        }
28    }
29    typedef ethertype-type {
30        type string {
31            pattern "[0-9a-fA-F]{2}-[0-9a-fA-F]{2}";
32        }
33        description
34            "The EtherType value represented in the canonical order defined
35            by IEEE 802. The canonical representation uses uppercase
36            characters.";
37        reference
38            "9.2 of IEEE Std 802-2014";
39    }
40    typedef dot1q-tag-type {
41        type identityref {
42            base dot1q-vlan-type;
43        }
44        description
45            "Identifies a specific 802.1Q tag type";
46        reference
47            "IEEE Std 802.1Q-2018";
48    }
49    typedef traffic-class-type {
50        type uint8 {
51            range "0..7";
52        }
53        description
54            "This is the numerical value associated with a traffic class in
55            a Bridge. Larger values are associated with higher priority
56            traffic classes.";
57        reference
58            "3.239 of IEEE Std 802.1Q-2018";
59    }
60    grouping dot1q-tag-classifier-grouping {
61        description
62            "A grouping which represents an 802.1Q VLAN, matching both the
63            EtherType and a single VLAN Id.";
64        leaf tag-type {
65            type dot1q-tag-type;
66            mandatory true;
67            description
68                "VLAN type";
69        }
70        leaf vlan-id {
71            type vlanid;
72            mandatory true;
73        }
74    }
```

```
1         description
2         "VLAN Id";
3     }
4 grouping dot1q-tag-or-any-classifier-grouping {
5     description
6     "A grouping which represents an 802.1Q VLAN, matching both the
7     EtherType and a single VLAN Id or 'any' to match on any VLAN Id.";
8     leaf tag-type {
9         type dot1q-tag-type;
10        mandatory true;
11        description
12        "VLAN type";
13    }
14    leaf vlan-id {
15        type union {
16            type vlanid;
17            type enumeration {
18                enum any {
19                    value 4095;
20                    description
21                    "Matches 'any' VLAN in the range 1 to 4094 that is not
22                    matched by a more specific VLAN Id match";
23                }
24            }
25        }
26        mandatory true;
27        description
28        "VLAN Id or any";
29    }
30 }
31 grouping dot1q-tag-ranges-classifier-grouping {
32     description
33     "A grouping which represents an 802.1Q VLAN that matches a range
34     of VLAN Ids.";
35     leaf tag-type {
36         type dot1q-tag-type;
37         mandatory true;
38         description
39         "VLAN type";
40     }
41     leaf vlan-ids {
42         type vid-range-type;
43         mandatory true;
44         description
45         "VLAN Ids";
46     }
47 }
48 grouping dot1q-tag-ranges-or-any-classifier-grouping {
49     description
50     "A grouping which represents an 802.1Q VLAN, matching both the
51     EtherType and a single VLAN Id, ordered list of ranges, or 'any'
52     to match on any VLAN Id.";
53     leaf tag-type {
54         type dot1q-tag-type;
55         mandatory true;
56         description
57         "VLAN type";
58     }
59     leaf vlan-id {
60         type union {
61             type vid-range-type;
62             type enumeration {
63                 enum any {
64                     value 4095;
65                     description
66                     "Matches 'any' VLAN in the range 1 to 4094.";
67                 }
68             }
69         }
70     }
71     mandatory true;
72     description
```



```
1         "VLAN Ids or any";
2     }
3 grouping priority-regeneration-table-grouping {
4     description
5     "The priority regeneration table provides the ability to map
6     incoming priority values on a per-Port basis, under management
7     control.";
8     reference
9     "6.9.4 of IEEE Std 802.1Q-2018";
10    leaf priority0 {
11        type priority-type;
12        default "0";
13        description
14        "Priority 0";
15        reference
16        "12.6.2.3 of IEEE Std 802.1Q-2018
17        6.9.4 of IEEE Std 802.1Q-2018";
18    }
19    leaf priority1 {
20        type priority-type;
21        default "1";
22        description
23        "Priority 1";
24        reference
25        "12.6.2.3 of IEEE Std 802.1Q-2018
26        6.9.4 of IEEE Std 802.1Q-2018";
27    }
28    leaf priority2 {
29        type priority-type;
30        default "2";
31        description
32        "Priority 2";
33        reference
34        "12.6.2.3 of IEEE Std 802.1Q-2018
35        6.9.4 of IEEE Std 802.1Q-2018";
36    }
37    leaf priority3 {
38        type priority-type;
39        default "3";
40        description
41        "Priority 3";
42        reference
43        "12.6.2.3 of IEEE Std 802.1Q-2018
44        6.9.4 of IEEE Std 802.1Q-2018";
45    }
46    leaf priority4 {
47        type priority-type;
48        default "4";
49        description
50        "Priority 4";
51        reference
52        "12.6.2.3 of IEEE Std 802.1Q-2018
53        6.9.4 of IEEE Std 802.1Q-2018";
54    }
55    leaf priority5 {
56        type priority-type;
57        default "5";
58        description
59        "Priority 5";
60        reference
61        "12.6.2.3 of IEEE Std 802.1Q-2018
62        6.9.4 of IEEE Std 802.1Q-2018";
63    }
64    leaf priority6 {
65        type priority-type;
66        default "6";
67        description
68        "Priority 6";
69        reference
70        "12.6.2.3 of IEEE Std 802.1Q-2018
71        6.9.4 of IEEE Std 802.1Q-2018";
72    }
73 }
```

```
1      }
2      leaf priority7 {
3          type priority-type;
4          default "7";
5          description
6              "Priority 7";
7          reference
8              "12.6.2.3 of IEEE Std 802.1Q-2018
9              6.9.4 of IEEE Std 802.1Q-2018";
10     }
11 }
12 grouping pcp-decoding-table-grouping {
13     description
14         "The Priority Code Point decoding table enables the decoding of
15         the priority and drop-eligible parameters from the PCP.";
16     reference
17         "6.9.3 of IEEE Std 802.1Q-2018";
18     list pcp-decoding-map {
19         key "pcp";
20         description
21             "This map associates the priority code point field found in
22             the VLAN to a priority and drop eligible value based upon the
23             priority code point selection type.";
24         leaf pcp {
25             type pcp-selection-type;
26             description
27                 "The priority code point selection type.";
28             reference
29                 "12.6.2.7 of IEEE Std 802.1Q-2018
30                 6.9.3 of IEEE Std 802.1Q-2018";
31         }
32         list priority-map {
33             key "priority-code-point";
34             description
35                 "This map associated a priority code point value to priority
36                 and drop eligible parameters.";
37             leaf priority-code-point {
38                 type priority-type;
39                 description
40                     "Priority associated with the pcp.";
41                 reference
42                     "12.6.2.7 of IEEE Std 802.1Q-2018
43                     6.9.3 of IEEE Std 802.1Q-2018";
44             }
45             leaf priority {
46                 type priority-type;
47                 description
48                     "Priority associated with the pcp.";
49                 reference
50                     "12.6.2.7 of IEEE Std 802.1Q-2018
51                     6.9.3 of IEEE Std 802.1Q-2018";
52             }
53             leaf drop-eligible {
54                 type boolean;
55                 description
56                     "Drop eligible value for pcp";
57                 reference
58                     "12.6.2.7 of IEEE Std 802.1Q-2018
59                     6.9.3 of IEEE Std 802.1Q-2018";
60             }
61         }
62     }
63 }
64 grouping pcp-encoding-table-grouping {
65     description
66         "The Priority Code Point encoding table encodes the priority and
67         drop-eligible parameters in the PCP field of the VLAN tag.";
68     reference
69         "12.6.2.9 of IEEE Std 802.1Q-2018
70         6.9.3 of IEEE Std 802.1Q-2018";
71     list pcp-encoding-map {
72         key "pcp";
```

```
1      description
2      "This map associated the priority and drop-eligible parameters
3      to the priority used to encode the PCP of the VLAN based upon
4      the priority code point selection type.";
5      leaf pcp {
6          type pcp-selection-type;
7          description
8          "The priority code point selection type.";
9          reference
10         "12.6.2.7 of IEEE Std 802.1Q-2018
11         6.9.3 of IEEE Std 802.1Q-2018";
12     }
13     list priority-map {
14         key "priority dei";
15         description
16         "This map associated the priority and drop-eligible
17         parameters to the priority code point field of the VLAN tag.";
18         leaf priority {
19             type priority-type;
20             description
21             "Priority associated with the pcp.";
22             reference
23             "12.6.2.7 of IEEE Std 802.1Q-2018
24             6.9.3 of IEEE Std 802.1Q-2018";
25         }
26         leaf dei {
27             type boolean;
28             description
29             "The drop eligible value.";
30             reference
31             "12.6.2 of IEEE Std 802.1Q-2018
32             8.6.6 of IEEE Std 802.1Q-2018";
33         }
34         leaf priority-code-point {
35             type priority-type;
36             description
37             "PCP value for priority when DEI value";
38             reference
39             "12.6.2.9 of IEEE Std 802.1Q-2018
40             6.9.3 of IEEE Std 802.1Q-2018";
41         }
42     }
43 }
44
45 grouping service-access-priority-table-grouping {
46     description
47     "The Service Access Priority Table associates a received
48     priority with a service access priority.";
49     reference
50     "12.6.2.17 of IEEE Std 802.1Q-2018
51     6.13.1 of IEEE Std 802.1Q-2018";
52     leaf priority0 {
53         type priority-type;
54         default "0";
55         description
56         "Service access priority value for priority 0";
57         reference
58         "12.6.2.17 of IEEE Std 802.1Q-2018
59         6.13.1 of IEEE Std 802.1Q-2018";
60     }
61     leaf priority1 {
62         type priority-type;
63         default "1";
64         description
65         "Service access priority value for priority 1";
66         reference
67         "12.6.2.17 of IEEE Std 802.1Q-2018
68         6.13.1 of IEEE Std 802.1Q-2018";
69     }
70     leaf priority2 {
71         type priority-type;
72         default "2";
```

```
1         description
2         "Service access priority value for priority 2";
3         reference
4         "12.6.2.17 of IEEE Std 802.1Q-2018
5         6.13.1 of IEEE Std 802.1Q-2018";
6     }
7     leaf priority3 {
8         type priority-type;
9         default "3";
10        description
11        "Service access priority value for priority 3";
12        reference
13        "12.6.2.17 of IEEE Std 802.1Q-2018
14        6.13.1 of IEEE Std 802.1Q-2018";
15    }
16    leaf priority4 {
17        type priority-type;
18        default "4";
19        description
20        "Service access priority value for priority 4";
21        reference
22        "12.6.2.17 of IEEE Std 802.1Q-2018
23        6.13.1 of IEEE Std 802.1Q-2018";
24    }
25    leaf priority5 {
26        type priority-type;
27        default "5";
28        description
29        "Service access priority value for priority 5";
30        reference
31        "12.6.2.17 of IEEE Std 802.1Q-2018
32        6.13.1 of IEEE Std 802.1Q-2018";
33    }
34    leaf priority6 {
35        type priority-type;
36        default "6";
37        description
38        "Service access priority value for priority 6";
39        reference
40        "12.6.2.17 of IEEE Std 802.1Q-2018
41        6.13.1 of IEEE Std 802.1Q-2018";
42    }
43    leaf priority7 {
44        type priority-type;
45        default "7";
46        description
47        "Service access priority value for priority 7";
48        reference
49        "12.6.2.17 of IEEE Std 802.1Q-2018
50        6.13.1 of IEEE Std 802.1Q-2018";
51    }
52    }
53    grouping traffic-class-table-grouping {
54        description
55        "The Traffic Class Table models the operations that can be
56        performed on, or inquire about, the current contents of the
57        Traffic Class Table (8.6.6) for a given Port.";
58        reference
59        "12.6.3 of IEEE Std 802.1Q-2018
60        8.6.6 of IEEE Std 802.1Q-2018";
61        list traffic-class-map {
62            key "priority";
63            description
64            "The priority index into the traffic class table.";
65            leaf priority {
66                type priority-type;
67                description
68                "The priority of the traffic class entry.";
69                reference
70                "8.6.6 of IEEE Std 802.1Q-2018";
71            }
72        }
73        list available-traffic-class {
```

```
1         key "num-traffic-class";
2         description
3             "The traffic class index associated with a given priority
4             within the traffic class table.";
5         reference
6             "8.6.6 of IEEE Std 802.1Q-2018";
7         leaf num-traffic-class {
8             type uint8 {
9                 range "1..8";
10            }
11            description
12                "The available number of traffic classes.";
13            reference
14                "8.6.6 of IEEE Std 802.1Q-2018";
15        }
16        leaf traffic-class {
17            type traffic-class-type;
18            description
19                "The traffic class index associated with a given traffic
20                class entry.";
21            reference
22                "8.6.6 of IEEE Std 802.1Q-2018";
23        }
24    }
25 }
26
27 grouping port-map-grouping {
28     description
29         "A set of control indicators, one for each Port. A Port Map,
30         containing a control element for each outbound Port";
31     reference
32         "8.8.1 of IEEE Std 802.1Q-2018
33         8.8.2 of IEEE Std 802.1Q-2018";
34     list port-map {
35         key "port-ref";
36         description
37             "The list of entries composing the port map.";
38         leaf port-ref {
39             type port-number-type;
40             description
41                 "The interface port reference associated with this map.";
42             reference
43                 "8.8.1 of IEEE Std 802.1Q-2018";
44         }
45     }
46     choice map-type {
47         description
48             "Type of port map";
49         container static-filtering-entries {
50             description
51                 "Static filtering entries attributes.";
52             leaf control-element {
53                 type enumeration {
54                     enum forward {
55                         description
56                             "Forwarded, independently of any dynamic filtering
57                             information held by the FDB.";
58                     }
59                     enum filter {
60                         description
61                             "Filtered, independently of any dynamic filtering
62                             information.";
63                     }
64                     enum forward-filter {
65                         description
66                             "Forwarded or filtered on the basis of dynamic
67                             filtering information, or on the basis of the
68                             default Group filtering behavior for the outbound
69                             Port (8.8.6) if no dynamic filtering information is
70                             present specifically for the MAC address.";
71                     }
72                 }
73             }
74         }
75     }
76     description
```

```
1         "containing a control element for each outbound Port,  
2         specifying that a frame with a destination MAC address,  
3         and in the case of VLAN Bridge components, VID that  
4         meets this specification.";  
5         reference  
6         "8.8.1 of IEEE Std 802.1Q-2018";  
7     }  
8     leaf connection-identifier {  
9         type port-number-type;  
10        description  
11        "A Port MAP may contain a connection identifier (8.8.12)  
12        for each outbound port. The connection identifier may be  
13        associated with the Bridge Port value maintained in a  
14        Dynamic Filtering Entry of the FDB for Bridge Ports.";  
15        reference  
16        "8.8.1 of IEEE Std 802.1Q-2018  
17        8.8.12 of IEEE Std 802.1Q-2018";  
18    }  
19 }  
20 container static-vlan-registration-entries {  
21     description  
22     "Static VLAN registration entries.";  
23     leaf registrar-admin-control {  
24         type enumeration {  
25             enum fixed-new-ignored {  
26                 description  
27                 "Registration Fixed (New ignored).";  
28             }  
29             enum fixed-new-propagated {  
30                 description  
31                 "Registration Fixed (New propagated).";  
32             }  
33             enum forbidden {  
34                 description  
35                 "Registration Forbidden.";  
36             }  
37             enum normal {  
38                 description  
39                 "Normal Registration.";  
40             }  
41         }  
42     }  
43     description  
44     "The Registrar Administrative Control values for MVRP  
45     and MIRP for the VID.";  
46     reference  
47     "8.8.2 of IEEE Std 802.1Q-2018";  
48 }  
49 leaf vlan-transmitted {  
50     type enumeration {  
51         enum tagged {  
52             description  
53             "VLAN-tagged";  
54         }  
55         enum untagged {  
56             description  
57             "VLAN-untagged";  
58         }  
59     }  
60     description  
61     "Whether frames are to be VLAN-tagged or untagged when  
62     transmitted.";  
63     reference  
64     "8.8.2 of IEEE Std 802.1Q-2018";  
65 }  
66 }  
67 container mac-address-registration-entries {  
68     description  
69     "MAC address registration entries attributes.";  
70     leaf control-element {  
71         type enumeration {  
72             enum registered {  
73                 description
```

```
1         "Forwarded, independently of any dynamic filtering
2         information held by the FDB.";
3     }
4     enum not-registered {
5         description
6         "Filtered, independently of any dynamic filtering
7         information.";
8     }
9     description
10    "Containing a control element for each outbound Port,
11    specifying that a frame with a destination MAC address,
12    and in the case of VLAN Bridge components, VID that
13    meets this specification.";
14    reference
15    "8.8.4 of IEEE Std 802.1Q-2018";
16 }
17 container dynamic-vlan-registration-entries {
18     description
19     "Dynamic VLAN registration entries attributes.";
20     leaf control-element {
21         type enumeration {
22             enum registered {
23                 description
24                 "Forwarded, independently of any dynamic filtering
25                 information held by the FDB.";
26             }
27         }
28     }
29     description
30     "Containing a control element for each outbound Port,
31     specifying that a frame with a destination MAC address,
32     and in the case of VLAN Bridge components, VID that
33     meets this specification.";
34     reference
35     "8.8.5 of IEEE Std 802.1Q-2018";
36 }
37 container dynamic-reservation-entries {
38     description
39     "Dynamic reservation entries attributes.";
40     leaf control-element {
41         type enumeration {
42             enum forward {
43                 description
44                 "Forwarded, independently of any dynamic filtering
45                 information held by the FDB.";
46             }
47             enum filter {
48                 description
49                 "Filtered, independently of any dynamic filtering
50                 information.";
51             }
52         }
53     }
54     description
55     "Containing a control element for each outbound Port,
56     specifying that a frame with a destination MAC address,
57     and in the case of VLAN Bridge components, VID that
58     meets this specification.";
59     reference
60     "8.8.7 of IEEE Std 802.1Q-2018";
61 }
62 container dynamic-filtering-entries {
63     description
64     "Dynamic filtering entries attributes.";
65     leaf control-element {
66         type enumeration {
67             enum forward {
68                 description
69                 "Forwarded, independently of any dynamic filtering
70                 information held by the FDB.";
```

```
1         }
2     }
3     description
4         "Containing a control element for each outbound Port,
5         specifying that a frame with a destination MAC address,
6         and in the case of VLAN Bridge components, VID that
7         meets this specification.";
8     reference
9         "8.8.3 of IEEE Std 802.1Q-2018";
10 }
11 }
12 }
13 }
14 }
15 }
16 }
17 }
18 }
19 }
20 }
21 }
22 }
23 }
24 }
25 }
26 }
27 }
28 }
29 }
30 }
31 }
32 }
33 }
34 }
35 }
36 }
37 }
38 }
39 }
40 }
41 }
42 }
43 }
44 }
45 }
46 }
47 }
48 }
49 }
50 }
51 }
52 }
53 }
54 }
```

```
grouping bridge-port-statistics-grouping {
    description
        "Grouping of bridge port statistics.";
    reference
        "12.6.1.1.3 of IEEE Std 802.1Q-2018";
    leaf delay-exceeded-discards {
        type yang:counter64;
        description
            "The number of frames discarded by this port due to excessive
            transit delay through the Bridge. It is incremented by both
            transparent and source route Bridges.";
        reference
            "12.6.1.1.3 of IEEE Std 802.1Q-2018
            8.6.6 of IEEE Std 802.1Q-2018";
    }
    leaf mtu-exceeded-discards {
        type yang:counter64;
        description
            "The number of frames discarded by this port due to an
            excessive size. It is incremented by both transparent and
            source route Bridges.";
        reference
            "12.6.1.1.3, item g) of IEEE Std 802.1Q-2018";
    }
    leaf frame-rx {
        type yang:counter64;
        description
            "The number of frames that have been received by this port
            from its segment. Note that a frame received on the interface
            corresponding to this port is only counted by this object if
            and only if it is for a protocol being processed by the local
            bridging function, including Bridge management frames.";
        reference
            "12.6.1.1.3 of IEEE Std 802.1Q-2018";
    }
    leaf octets-rx {
        type yang:counter64;
        description
            "The total number of octets in all valid frames received
            (including BPDUs, frames addressed to the Bridge as an end
            station, and frames that were submitted to the Forwarding
            Process).";
        reference
            "12.6.1.1.3 of IEEE Std 802.1Q-2018";
    }
    leaf frame-tx {
        type yang:counter64;
        description
            "The number of frames that have been transmitted by this port
            to its segment. Note that a frame transmitted on the interface
            corresponding to this port is only counted by this object if
            and only if it is for a protocol being processed by the local
            bridging function, including Bridge management frames.";
    }
    leaf octets-tx {
        type yang:counter64;
        description
            "The total number of octets that have been transmitted by this
```



```
1         port to its segment.";  
2     }  
3     leaf discard-inbound {  
4         type yang:counter64;  
5         description  
6             "Count of received valid frames that were discarded (i.e.,  
7             filtered) by the Forwarding Process.";  
8         reference  
9             "12.6.1.1.3 of IEEE Std 802.1Q-2018";  
10    }  
11    leaf forward-outbound {  
12        type yang:counter64;  
13        description  
14            "The number of frames forwarded to the associated MAC Entity  
15            (8.5).";  
16        reference  
17            "12.6.1.1.3 of IEEE Std 802.1Q-2018";  
18    }  
19    leaf discard-lack-of-buffers {  
20        type yang:counter64;  
21        description  
22            "The count of frames that were to be transmitted through the  
23            associated Port but were discarded due to lack of buffers.";  
24        reference  
25            "12.6.1.1.3 of IEEE Std 802.1Q-2018";  
26    }  
27    leaf discard-transit-delay-exceeded {  
28        type yang:counter64;  
29        description  
30            "The number of frames discarded by this port due to excessive  
31            transit delay through the Bridge. It is incremented by both  
32            transparent and source route Bridges.";  
33        reference  
34            "12.6.1.1.3 of IEEE Std 802.1Q-2018";  
35    }  
36 }  
37 }  
38 }
```

48.7.3 Definitions for the ieee802-dot1q-bridge YANG module

```
40 module ieee802-dot1q-bridge {  
41     namespace urn:ieee:std:802.1Q:yang:ieee802-dot1q-bridge;  
42     prefix dot1q;  
43     import ieee802-types {  
44         prefix ieee;  
45     }  
46     import ietf-yang-types {  
47         prefix yang;  
48     }  
49     import ietf-interfaces {  
50         prefix if;  
51     }  
52     import iana-if-type {  
53         prefix ianaif;  
54     }  
55     import ieee802-dot1q-types {  
56         prefix dot1qtypes;  
57     }  
58     organization
```

```
1      "IEEE 802.1 Working Group";
2  contact
3      "WG-URL: http://www.ieee802.org/1/
4      WG-EMail: stds-802-1-L@ieee.org
5
6      Contact: IEEE 802.1 Working Group Chair
7      Postal: C/O IEEE 802.1 Working Group
8              IEEE Standards Association
9              445 Hoes Lane
10             P.O. Box 1331
11             Piscataway
12             NJ 08854
13             USA
14
15     E-mail: STDS-802-1-L@IEEE.ORG";
16 description
17     "This YANG module describes the bridge configuration model for the
18     following IEEE 802.1Q Bridges:
19     1) Two Port MAC Relays
20     2) Customer VLAN Bridges
21     3) Provider Bridges.";
22 revision 2018-03-07 {
23     description
24         "Published as part of IEEE Std 802.1Q-2018.
25         Initial version.";
26     reference
27         "IEEE Std 802.1Q-2018, Bridges and Bridged Networks.";
28 }
29
30 feature ingress-filtering {
31     description
32         "Each Port may support an Enable Ingress Filtering parameter. A
33         frame received on a Port that is not in the member set (8.8.10)
34         associated with the frames VID shall be discarded if this
35         parameter is set. The default value for this parameter is reset,
36         i.e., Disable Ingress Filtering, for all Ports. Any Port that
37         supports setting this parameter shall also support resetting it.
38         The parameter may be configured by the management operations
39         defined in Clause 12.";
40     reference
41         "8.6.2 of IEEE Std 802.1Q-2018";
42 }
43 feature extended-filtering-services {
44     description
45         "Extended Filtering Services support the filtering behavior
46         required for regions of a network in which potential recipients
47         of multicast frames exist, and where both the potential
48         recipients of frames and the Bridges are able to support dynamic
49         configuration of filtering information for group MAC addresses.
50         In order to integrate this extended filtering behavior with the
51         needs of regions of the network that support only Basic
52         Filtering Services, Bridges that support Extended Filtering
53         Services can be statically and dynamically configured to modify
54         their filtering behavior on a per-group MAC address basis, and
55         also on the basis of the overall filtering service provided by
56         each outbound Port with regard to multicast frames. The latter
57         capability permits configuration of the Ports default forwarding
58         or filtering behavior with regard to group MAC addresses for
59         which no specific static or dynamic filtering information has
60         been configured.";
61     reference
62         "8.8.4 of IEEE Std 802.1Q-2018
63         Clause 10 of IEEE Std 802.1Q-2018";
64 }
65 feature port-and-protocol-based-vlan {
66     description
67         "A VLAN-aware bridge component implementation in conformance to
68         the provisions of this standard for Port-and-Protocol-based VLAN
69         classification (5.4.1) shall 1) Support one or more of the
70         following Protocol Classifications and Protocol Template
71         formats: Ethernet, RFC_1042, SNAP_8021H, SNAP_Other, or
72         LLC_Other (6.12); and may 2) Support configuration of the
```

```
1     contents of the Protocol Group Database.";  
2     reference  
3     "5.4.1.2 of IEEE Std 802.1Q-2018";  
4 }  
5 feature flow-filtering {  
6     description  
7     "Flow filtering support enables Bridges to distinguish frames  
8     belonging to different client flows and to use this information  
9     in the forwarding process. Information related to client flows  
10    may be used at the boundary of an SPT Domain to generate a flow  
11    hash value. The flow hash, carried in an F-TAG, serves to  
12    distinguish frames belonging to different flows and can be used  
13    in the forwarding process to distribute frames over equal cost  
14    paths. This provides for finer granularity load spreading while  
15    maintaining frame order for each client flow.";  
16    reference  
17    "44.2 of IEEE Std 802.1Q-2018";  
18 }  
19 feature simple-bridge-port {  
20     description  
21     "A simple bridge port allows underlying (MAC) layers to share  
22     the same Interface as the Bridge Port.";  
23 }  
24 feature flexible-bridge-port {  
25     description  
26     "A flexible bridge port supports an Interface that is a Bridge  
27     Port to be a separate Interface from the underlying (MAC) layer.";  
28 }  
29 identity type-of-bridge {  
30     description  
31     "Represents the configured Bridge type.";  
32 }  
33 identity customer-vlan-bridge {  
34     base type-of-bridge;  
35     description  
36     "Base identity for a Customer VLAN Bridge.";  
37 }  
38 identity provider-bridge {  
39     base type-of-bridge;  
40     description  
41     "Base identity for a Provider Bridge (PB).";  
42 }  
43 identity provider-edge-bridge {  
44     base type-of-bridge;  
45     description  
46     "Base identity for a Provider Edge Bridge (PEB).";  
47 }  
48 identity two-port-mac-relay-bridge {  
49     base type-of-bridge;  
50     description  
51     "Base identity for a Two Port MAC Relay (TPMR).";  
52 }  
53 identity type-of-component {  
54     description  
55     "Represents the type of Component.";  
56 }  
57 identity c-vlan-component {  
58     base type-of-component;  
59     description  
60     "Base identity for a C-VLAN component.";  
61 }  
62 identity s-vlan-component {  
63     base type-of-component;  
64     description  
65     "Base identity for a S-VLAN component.";  
66 }  
67 identity d-bridge-component {  
68     base type-of-component;  
69     description  
70     "Base identity for a VLAN unaware component.";  
71 }
```

```
1     identity edge-relay-component {
2         base type-of-component;
3         description
4             "Base identity for an EVB station ER component.";
5     }
6     identity type-of-port {
7         description
8             "Represents the type of Bridge port.";
9     }
10    identity c-vlan-bridge-port {
11        base type-of-port;
12        description
13            "Indicates the port can be a C-TAG aware port of an enterprise
14            VLAN aware Bridge.";
15    }
16    identity provider-network-port {
17        base type-of-port;
18        description
19            "Indicates the port can be an S-TAG aware port of a Provider
20            Bridge or Backbone Edge Bridge used for connections within a PBN
21            (Provider Bridged Network) or PBBN (Provider Backbone Bridged
22            Network).";
23    }
24    identity customer-network-port {
25        base type-of-port;
26        description
27            "Indicates the port can be an S-TAG aware port of a Provider
28            Bridge or Backbone Edge Bridge used for connections to the
29            exterior of a PBN (Provider Bridged Network) or PBBN (Provider
30            Backbone Bridged Network).";
31    }
32    identity customer-edge-port {
33        base type-of-port;
34        description
35            "Indicates the port can be a C-TAG aware port of a Provider
36            Bridge used for connections to the exterior of a PBN (Provider
37            Bridged Network) or PBBN (Provider Backbone Bridged Network).";
38    }
39    identity d-bridge-port {
40        base type-of-port;
41        description
42            "Indicates the port can be a VLAN-unaware member of an 802.1Q
43            Bridge.";
44    }
45    identity remote-customer-access-port {
46        base type-of-port;
47        description
48            "Indicates the port can be an S-TAG aware port of a Provider
49            Bridge capable of providing Remote Customer Service Interfaces.";
50    }
51    identity bridge-interface {
52        description
53            "Generic interface property that represents any interface that
54            can be associated with an IEEE 802.1Q compliant Bridge
55            component. Any new Interface types would derive from this
56            identity to automatically pick up Bridge related configuration
57            or operational data.";
58    }
59    container bridges {
60        description
61            "Contains the Bridge(s) configuration information.";
62        list bridge {
63            key "name";
64            unique "address";
65            description
66                "Provides configuration data in support of the Bridge
67                Configuration resources. There is a single bridge data node
68                per Bridge.";
69            leaf name {
70                type dot1qtypes:name-type;
71                description

```

```
1         "A text string associated with the Bridge, of locally
2         determined significance.";
3         reference
4         "12.4 of IEEE Std 802.1Q-2018";
5     }
6     leaf address {
7         type ieee:mac-address;
8         mandatory true;
9         description
10        "The MAC address for the Bridge from which the Bridge
11        Identifiers used by the STP, RSTP, and MSTP are derived.";
12        reference
13        "12.4 of IEEE Std 802.1Q-2018";
14    }
15    leaf bridge-type {
16        type identityref {
17            base type-of-bridge;
18        }
19        mandatory true;
20        description
21        "The type of Bridge.";
22    }
23    leaf ports {
24        type uint16 {
25            range "1..4095";
26        }
27        config false;
28        description
29        "The number of Bridge Ports (MAC Entities)";
30        reference
31        "12.4 of IEEE Std 802.1Q-2018";
32    }
33    leaf up-time {
34        type yang:zero-based-counter32;
35        units "seconds";
36        config false;
37        description
38        "The count in seconds of the time elapsed since the Bridge
39        was last reset or initialized.";
40        reference
41        "12.4 of IEEE Std 802.1Q-2018";
42    }
43    leaf components {
44        type uint32;
45        config false;
46        description
47        "The number of components associated with the Bridge.";
48    }
49    list component {
50        key "name";
51        description
52        "The set of components associated with a given Bridge. For
53        example, - A TPMR is associated with a single VLAN
54        unaware component. - A Customer VLAN Bridge is associated
55        with a single VLAN aware component. - A Provider Bridge is
56        associated with a single S-VLAN component and zero or more
57        C-VLAN components.";
58        reference
59        "12.3 of IEEE Std 802.1Q-2018";
60        leaf name {
61            type string;
62            description
63            "The name of the Component.";
64        }
65        leaf id {
66            type uint32;
67            description
68            "Unique identifier for a particular Bridge component
69            within the system.";
70            reference
71            "12.3, item 1) of IEEE Std 802.1Q-2018";
72        }
73    }
```

```
1         leaf type {
2             type identityref {
3                 base type-of-component;
4             }
5             mandatory true;
6             description
7                 "The type of component used to classify a particular
8                 Bridge component within a Bridge system comprising
9                 multiple components.";
10            reference
11                "12.3, item m) of IEEE Std 802.1Q-2018";
12        }
13        leaf address {
14            type ieee:mac-address;
15            description
16                "Unique EUI-48 Universally Administered MAC address
17                assigned to a Bridge component.";
18            reference
19                "13.24 of IEEE Std 802.1Q-2018
20                8.13.8 of IEEE Std 802.1Q-2018";
21        }
22        leaf traffic-class-enabled {
23            type boolean;
24            default "true";
25            description
26                "Indication of Traffic Classes enablement associated with
27                the Bridge Component. A value of True indicates that
28                Traffic Classes are enabled on this Bridge Component. A
29                value of False indicates that the Bridge Component
30                operates with a single priority level for all traffic.";
31            reference
32                "12.4.1.5.1 of IEEE Std 802.1Q-2018";
33        }
34        leaf ports {
35            type uint16 {
36                range "1..4095";
37            }
38            config false;
39            description
40                "The number of Bridge Ports associated with the Bridge
41                Component.";
42            reference
43                "12.4.1.1.3, item c) of IEEE Std 802.1Q-2018";
44        }
45        leaf-list bridge-port {
46            type if:interface-ref;
47            config false;
48            description
49                "List of bridge-port references.";
50        }
51        container capabilities {
52            config false;
53            description
54                "Array of Boolean values of the feature capabilities
55                associated with a given Bridge Component.";
56            reference
57                "12.10.1.1.3, item b) of IEEE Std 802.1Q-2018
58                12.4.1.5.2 of IEEE Std 802.1Q-2018";
59            leaf extended-filtering {
60                type boolean;
61                default "false";
62                description
63                    "Can perform filtering on individual multicast addresses
64                    controlled by MMRP.";
65                reference
66                    "12.4.1.5.2 of IEEE Std 802.1Q-2018";
67            }
68            leaf traffic-classes {
69                type boolean;
70                default "false";
71                description
72                    "Can map priority to multiple traffic classes.";
73            }
74        }
75    }
```

```
1         reference
2           "12.4.1.5.2 of IEEE Std 802.1Q-2018";
3     }
4     leaf static-entry-individual-port {
5         type boolean;
6         default "false";
7         description
8           "Static entries per port.";
9         reference
10          "12.4.1.5.2 of IEEE Std 802.1Q-2018";
11     }
12     leaf ivl-capable {
13         type boolean;
14         default "true";
15         description
16           "Independent VLAN Learning (IVL).";
17         reference
18           "12.4.1.5.2 of IEEE Std 802.1Q-2018";
19     }
20     leaf svl-capable {
21         type boolean;
22         default "false";
23         description
24           "Shared VLAN Learning (SVL).";
25         reference
26           "12.4.1.5.2 of IEEE Std 802.1Q-2018";
27     }
28     leaf hybrid-capable {
29         type boolean;
30         default "false";
31         description
32           "Both IVL and SVL simultaneously.";
33         reference
34           "12.4.1.5.2 of IEEE Std 802.1Q-2018";
35     }
36     leaf configurable-pvid-tagging {
37         type boolean;
38         default "false";
39         description
40           "Whether the implementation supports the ability to
41           override the default PVID setting and its egress status
42           (VLAN-tagged or Untagged) on each port.";
43         reference
44           "12.4.1.5.2 of IEEE Std 802.1Q-2018";
45     }
46     leaf local-vlan-capable {
47         type boolean;
48         default "false";
49         description
50           "Can support multiple local Bridges, outside the scope
51           of 802.1Q defined VLANs.";
52         reference
53           "12.4.1.5.2 of IEEE Std 802.1Q-2018";
54     }
55 }
56 container filtering-database {
57     when "../bridge-type != 'two-port-mac-relay-bridge'" {
58         description
59           "Applies to non TPMRs.";
60     }
61     description
62       "Contains filtering information used by the Forwarding
63       Process in deciding through which Ports of the Bridge
64       frames should be forwarded.";
65     reference
66       "12.7 of IEEE Std 802.1Q-2018";
67     leaf aging-time {
68         type uint32 {
69             range "10..10000000";
70         }
71         units "seconds";
72         default "300";
73     }
74 }
```

```
1         description
2             "The timeout period in seconds for aging out
3             dynamically-learned forwarding information.";
4         reference
5             "12.7 of IEEE Std 802.1Q-2018
6             8.8.3 of IEEE Std 802.1Q-2018";
7     }
8     leaf size {
9         type yang:gauge32;
10        config false;
11        description
12            "The maximum number of entries that can be held in the
13            FDB.";
14        reference
15            "12.7 of IEEE Std 802.1Q-2018";
16    }
17    leaf static-entries {
18        type yang:gauge32;
19        config false;
20        description
21            "The number of Static Filtering entries currently in the
22            FDB.";
23        reference
24            "12.7 of IEEE Std 802.1Q-2018
25            8.8.1 of IEEE Std 802.1Q-2018";
26    }
27    leaf dynamic-entries {
28        type yang:gauge32;
29        config false;
30        description
31            "The number of Dynamic Filtering entries currently in
32            the FDB.";
33        reference
34            "12.7 of IEEE Std 802.1Q-2018
35            8.8.3 of IEEE Std 802.1Q-2018";
36    }
37    leaf static-vlan-registration-entries {
38        type yang:gauge32;
39        config false;
40        description
41            "The number of Static VLAN Registration entries
42            currently in the FDB.";
43        reference
44            "12.7 of IEEE Std 802.1Q-2018
45            8.8.2 of IEEE Std 802.1Q-2018";
46    }
47    leaf dynamic-vlan-registration-entries {
48        type yang:gauge32;
49        config false;
50        description
51            "The number of Dynamic VLAN Registration entries
52            currently in the FDB.";
53        reference
54            "12.7 of IEEE Std 802.1Q-2018
55            8.8.5 of IEEE Std 802.1Q-2018";
56    }
57    leaf mac-address-registration-entries {
58        if-feature "extended-filtering-services";
59        type yang:gauge32;
60        config false;
61        description
62            "The number of MAC Address Registration entries
63            currently in the FDB.";
64        reference
65            "12.7 of IEEE Std 802.1Q-2018
66            8.8.4 of IEEE Std 802.1Q-2018";
67    }
68    list filtering-entry {
69        key "database-id vids address";
70        description
71            "Information for the entries associated with the
72            Permanent Database.";
```



```
1         leaf database-id {
2             type uint32;
3             description
4                 "The identity of this Filtering Database.";
5             reference
6                 "12.7.7 of IEEE Std 802.1Q-2018";
7         }
8         leaf address {
9             type ieee:mac-address;
10            description
11                "A MAC address (unicast, multicast, broadcast) for
12                which the device has forwarding and/or filtering
13                information.";
14            reference
15                "12.7.7 of IEEE Std 802.1Q-2018";
16        }
17        leaf vids {
18            type dot1qtypes:vid-range-type;
19            description
20                "The set of VLAN identifiers to which this entry
21                applies.";
22            reference
23                "12.7.7 of IEEE Std 802.1Q-2018";
24        }
25        leaf entry-type {
26            type enumeration {
27                enum static {
28                    description
29                        "Static entry type";
30                }
31                enum dynamic {
32                    description
33                        "Dynamic/learnt entry type";
34                }
35            }
36            description
37                "The type of filtering entry. Whether static or
38                dynamic. Static entries can be created, deleted, and
39                retrieved. However, dynamic entries can only be
40                deleted or retrieved by the management entity.
41                Consequently, a Bridge is not required to accept a
42                command that can alter the dynamic entries except
43                delete a dynamic entry.";
44            reference
45                "12.7.7 of IEEE Std 802.1Q-2018";
46        }
47        uses dot1qtypes:port-map-grouping;
48        leaf status {
49            type enumeration {
50                enum other {
51                    description
52                        "None of the following. This may include the case
53                        where some other object is being used to determine
54                        if and how frames addressed to the value of the
55                        corresponding instance of 'address' are being
56                        forwarded.";
57                }
58                enum invalid {
59                    description
60                        "This entry is no longer valid (e.g., it was
61                        learned but has since aged out), but has not yet
62                        been flushed from the table.";
63                }
64                enum learned {
65                    description
66                        "The value of the corresponding instance of the
67                        port node was learned and is being used.";
68                }
69                enum self {
70                    description
71                        "The value of the corresponding instance of the
72                        address node representing one of the devices
```

```
1         address.";
2     }
3     enum mgmt {
4         description
5             "The value of the corresponding instance of
6             address node that is also the value of an existing
7             instance.";
8     }
9     config false;
10    description
11        "The status of this entry.";
12    }
13    }
14    list vlan-registration-entry {
15        key "database-id vids";
16        description
17            "The VLAN Registration Entries models the operations
18            that can be performed on a single VLAN Registration
19            Entry in the FDB. The set of VLAN Registration Entries
20            within the FDB changes under management control and also
21            as a result of MVRP exchanges";
22        reference
23            "12.7.5 of IEEE Std 802.1Q-2018";
24        leaf database-id {
25            type uint32;
26            description
27                "The identity of this Filtering Database.";
28            reference
29                "12.7.7 of IEEE Std 802.1Q-2018";
30        }
31        leaf vids {
32            type dot1qtypes:vid-range-type;
33            description
34                "The set of VLAN identifiers to which this entry
35                applies.";
36            reference
37                "12.7.7 of IEEE Std 802.1Q-2018";
38        }
39        leaf entry-type {
40            type enumeration {
41                enum static {
42                    description
43                        "Static entry type";
44                }
45                enum dynamic {
46                    description
47                        "Dynamic/learnt entry type";
48                }
49            }
50            description
51                "The type of filtering entry. Whether static or
52                dynamic. Static entries can be created, deleted, and
53                retrieved. However, dynamic entries can only be
54                deleted or retrieved by the management entity.
55                Consequently, a Bridge is not required to accept a
56                command that can alter the dynamic entries except
57                delete a dynamic entry.";
58            reference
59                "12.7.7 of IEEE Std 802.1Q-2018";
60        }
61        uses dot1qtypes:port-map-grouping;
62    }
63    }
64    container permanent-database {
65        description
66            "The Permanent Database container models the operations
67            that can be performed on, or affect, the Permanent
68            Database. There is a single Permanent Database per FDB.";
69        leaf size {
70            type yang:gauge32;
71            config false;
72        }
73    }
74    }
```

```
1         description
2             "The maximum number of entries that can be held in the
3             FDB.";
4         reference
5             "12.7.6 of IEEE Std 802.1Q-2018";
6     }
7     leaf static-entries {
8         type yang:gauge32;
9         config false;
10        description
11            "The number of Static Filtering entries currently in the
12            FDB.";
13        reference
14            "12.7.6 of IEEE Std 802.1Q-2018";
15    }
16    leaf static-vlan-registration-entries {
17        type yang:gauge32;
18        config false;
19        description
20            "The number of Static VLAN Registration entries
21            currently in the FDB.";
22        reference
23            "12.7.6 of IEEE Std 802.1Q-2018";
24    }
25    list filtering-entry {
26        key "database-id vids address";
27        description
28            "Information for the entries associated with the
29            Permanent Database.";
30        leaf database-id {
31            type uint32;
32            description
33                "The identity of this Filtering Database.";
34            reference
35                "12.7.7 of IEEE Std 802.1Q-2018";
36        }
37        leaf address {
38            type ieee:mac-address;
39            description
40                "A MAC address (unicast, multicast, broadcast) for
41                which the device has forwarding and/or filtering
42                information.";
43            reference
44                "12.7.7 of IEEE Std 802.1Q-2018";
45        }
46        leaf vids {
47            type dot1qt-types:vid-range-type;
48            description
49                "The set of VLAN identifiers to which this entry
50                applies.";
51            reference
52                "12.7.7 of IEEE Std 802.1Q-2018";
53        }
54        leaf status {
55            type enumeration {
56                enum other {
57                    description
58                        "None of the following. This may include the case
59                        where some other object is being used to determine
60                        if and how frames addressed to the value of the
61                        corresponding instance of 'address' are being
62                        forwarded.";
63                }
64                enum invalid {
65                    description
66                        "This entry is no longer valid (e.g., it was
67                        learned but has since aged out), but has not yet
68                        been flushed from the table.";
69                }
70                enum learned {
71                    description
72                        "The value of the corresponding instance of the
```

```
1         port node was learned and is being used.";
2     }
3     enum self {
4         description
5             "The value of the corresponding instance of the
6             address node representing one of the devices
7             address.";
8     }
9     enum mgmt {
10        description
11            "The value of the corresponding instance of
12            address node that is also the value of an existing
13            instance.";
14    }
15    }
16    config false;
17    description
18        "The status of this entry.";
19    }
20    uses dot1qtypes:port-map-grouping;
21    }
22    }
23    container bridge-vlan {
24        when "../bridge-type != 'two-port-mac-relay-bridge'" {
25            description
26                "Applies to non TPMRs.";
27        }
28        description
29            "The Bridge VLAN container models configuration
30            information that modify, or inquire about, the overall
31            configuration of the Bridges VLAN resources. There is a
32            single Bridge VLAN Configuration managed object per
33            Bridge.";
34        reference
35            "12.10 of IEEE Std 802.1Q-2018";
36        leaf version {
37            type uint16;
38            config false;
39            description
40                "The version number supported.";
41            reference
42                "12.10.1.3 of IEEE Std 802.1Q-2018";
43        }
44        leaf max-vids {
45            type uint16;
46            config false;
47            description
48                "The maximum number of VIDs supported.";
49            reference
50                "12.10.1.3 of IEEE Std 802.1Q-2018";
51        }
52        leaf override-default-pvid {
53            type boolean;
54            default "false";
55            config false;
56            description
57                "Indicates if the default PVID can be overridden, and
58                its egress status (VLAN-tagged or untagged) on each
59                port.";
60            reference
61                "12.10.1.3 of IEEE Std 802.1Q-2018";
62        }
63        leaf protocol-template {
64            if-feature "port-and-protocol-based-vlan";
65            type dot1qtypes:protocol-frame-format-type;
66            config false;
67            description
68                "The data-link encapsulation format or the
69                detagged_frame_type in a Protocol Template";
70            reference
71                "12.10.1.7 of IEEE Std 802.1Q-2018";
72        }
73    }
74    }
```

```
1         leaf max-msti {
2             type uint16;
3             config false;
4             description
5                 "The maximum number of MSTIs supported within an MST
6                 region (i.e., the number of spanning tree instances that
7                 can be supported in addition to the CIST), for MST
8                 Bridges. For SST Bridges, this parameter may be either
9                 omitted or reported as 0.";
10            reference
11                "12.10.1.7 of IEEE Std 802.1Q-2018";
12        }
13    list vlan {
14        key "vid";
15        description
16            "List of VLAN related configuration nodes associated
17            with the Bridge.";
18        reference
19            "12.10.2 of IEEE Std 802.1Q-2018";
20        leaf vid {
21            type dot1qtypes:vlan-index-type;
22            description
23                "The VLAN identifier to which this entry applies.";
24            reference
25                "12.10.2 of IEEE Std 802.1Q-2018";
26        }
27        leaf name {
28            type dot1qtypes:name-type;
29            description
30                "A text string of up to 32 characters of locally
31                determined significance.";
32            reference
33                "12.10.2 of IEEE Std 802.1Q-2018";
34        }
35        leaf-list untagged-ports {
36            type if:interface-ref;
37            config false;
38            description
39                "The set of ports in the untagged set for this VID.";
40            reference
41                "12.10.2.1.3 of IEEE Std 802.1Q-2018
42                8.8.2 of IEEE Std 802.1Q-2018";
43        }
44        leaf-list egress-ports {
45            type if:interface-ref;
46            config false;
47            description
48                "The set of egress ports in the member set for this
49                VID.";
50            reference
51                "12.10.2.1.3 of IEEE Std 802.1Q-2018
52                8.8.10 of IEEE Std 802.1Q-2018";
53        }
54    }
55    list protocol-group-database {
56        if-feature "port-and-protocol-based-vlan";
57        key "db-index";
58        description
59            "List of the protocol group database entries.";
60        reference
61            "12.10.1.7 of IEEE Std 802.1Q-2018
62            6.12.3 of IEEE Std 802.1Q-2018";
63        leaf db-index {
64            type uint16;
65            description
66                "The protocol group database index.";
67        }
68        leaf frame-format-type {
69            type dot1qtypes:protocol-frame-format-type;
70            description
71                "The data-link encapsulation format or the
72                detagged_frame_type in a Protocol Template";
```

```
1         reference
2         "12.10.1.7 of IEEE Std 802.1Q-2018";
3     }
4     choice frame-format {
5         description
6         "The identification of the protocol above the
7         data-link layer in a Protocol Template. Depending on
8         the frame type, the octet string will have one of the
9         following values: - For ethernet, rfc1042 and
10        snap8021H, this is the 16-bit (2-octet) IEEE 802
11        Clause 9.3 EtherType field. - For snapOther, this is
12        the 40-bit (5-octet) PID. - For llcOther, this is the
13        2-octet IEEE 802.2 Link Service Access Point (LSAP)
14        pair: first octet for Destination Service Access Point
15        (DSAP) and second octet for Source Service Access
16        Point (SSAP).";
17        reference
18        "12.10.1.7 of IEEE Std 802.1Q-2018";
19        case ethernet-rfc1042-snap8021H {
20            when
21            "frame-format-type = 'Ethernet' or "+
22            "frame-format-type = 'rfc1042' or "+
23            "frame-format-type = 'snap8021H'" {
24                description
25                "Applies to Ethernet, RFC 1042, SNAP 8021H frame
26                formats.";
27            }
28            description
29            "Identifier used if Ethenet, RFC1042, or SNAP 8021H.";
30            leaf ethertype {
31                type dot1qtypes:ethertype-type;
32                description
33                "Format containing the 16-bit IEEE 802 EtherType
34                field.";
35                reference
36                "9.3 of IEEE Std 802-2014";
37            }
38        }
39        case snap-other {
40            when "frame-format-type = 'snapOther'" {
41                description
42                "Applies to Snap Other frame formats.";
43            }
44            description
45            "Identifier used if SNAP other.";
46            leaf protocol-id {
47                type string {
48                    pattern "[0-9a-fA-F]{2}-[0-9a-fA-F]{2}]{4}";
49                }
50                description
51                "Format containing the 40-bit protocol identifier
52                (PID). The canonical representation uses uppercase
53                characters.";
54                reference
55                "12.10.1.7.1 of IEEE Std 802.1Q-2018";
56            }
57        }
58        case llc-other {
59            when "frame-format-type = 'llcOther'" {
60                description
61                "Applies to LLC Other frame formats";
62            }
63            description
64            "Identifier used if LLC other.";
65            container dsap-ssap-pairs {
66                description
67                "A pair of ISO/IEC 8802-2 DSAP and SSAP address
68                field values, for matching frame formats of
69                LLC Other.";
70                leaf llc-address {
71                    type string {
72                        pattern "[0-9a-fA-F]{2}-[0-9a-fA-F]{2}";
```

```
1         }
2         description
3         "A pair of ISO/IEC 8802-2 DSAP and SSAP address
4         field values, for matching frame formats of
5         LLC_Other. The canonical representation uses
6         uppercase characters.";
7         reference
8         "12.10.1.7.1 of IEEE Std 802.1Q-2018";
9     }
10    }
11    leaf group-id {
12        type uint32;
13        description
14        "Designates a group of protocols in the Protocol Group
15        Database.";
16        reference
17        "6.12.2 of IEEE Std 802.1Q-2018";
18    }
19    list vid-to-fid-allocation {
20        key "vids";
21        description
22        "This list allows inquiries about VID to FID
23        allocations.";
24        leaf vids {
25            type dot1qttype:vid-range-type;
26            description
27            "Range of VLAN identifiers.";
28            reference
29            "12.10.3 of IEEE Std 802.1Q-2018";
30        }
31        leaf fid {
32            type uint32;
33            config false;
34            description
35            "The Filtering Database used by a set of VIDs.";
36            reference
37            "12.10.3 of IEEE Std 802.1Q-2018";
38        }
39        leaf allocation-type {
40            type enumeration {
41                enum undefined {
42                    description
43                    "No allocation defined.";
44                }
45                enum fixed {
46                    description
47                    "A fixed allocation to FID is defined.";
48                }
49                enum dynamic {
50                    description
51                    "A dynamic allocation to FID is defined.";
52                }
53            }
54            config false;
55            description
56            "The type of allocation used";
57            reference
58            "12.10.3 of IEEE Std 802.1Q-2018";
59        }
60    }
61    list fid-to-vid-allocation {
62        key "fid";
63        description
64        "The FID to VID allocations managed object models
65        operations that inquire about FID to VID allocations.";
66        leaf fid {
67            type uint32;
68            description
69            "The Filtering Database used by a set of VIDs.";
70        }
71    }
72 }
```

```
1         reference
2         "12.10.3 of IEEE Std 802.1Q-2018";
3     }
4     leaf allocation-type {
5         type enumeration {
6             enum undefined {
7                 description
8                 "No allocation defined.";
9             }
10            enum fixed {
11                description
12                "A fixed allocation to FID is defined.";
13            }
14            enum dynamic {
15                description
16                "A dynamic allocation to FID is defined.";
17            }
18        }
19        config false;
20        description
21        "The type of allocation used";
22        reference
23        "12.10.3 of IEEE Std 802.1Q-2018";
24    }
25    leaf-list vid {
26        type dot1qtypes:vlan-index-type;
27        config false;
28        description
29        "The VLAN identifier to which this entry applies.";
30        reference
31        "12.7.7 of IEEE Std 802.1Q-2018";
32    }
33    list vid-to-fid {
34        key "vid";
35        description
36        "Fixed allocation of a VID to an FID. The underlying
37        system will ensure that subsequent commands that make
38        changes to the VID to FID mapping can override previous
39        associations.";
40        reference
41        "12.10.3.4 of IEEE Std 802.1Q-2018
42        12.10.3.5 of IEEE Std 802.1Q-2018";
43        leaf vid {
44            type dot1qtypes:vlan-index-type;
45            description
46            "A list of VLAN identifier associated with a given
47            database identifier (i.e., FID).";
48            reference
49            "12.7.7 of IEEE Std 802.1Q-2018";
50        }
51        leaf fid {
52            type uint32;
53            description
54            "The Filtering Database used by this VLAN";
55            reference
56            "12.10.3 of IEEE Std 802.1Q-2018";
57        }
58    }
59    }
60    container bridge-mst {
61        when "../bridge-type != 'two-port-mac-relay-bridge'" {
62            description
63            "Applies to non TPMRs.";
64        }
65        description
66        "The Bridge MST container models configuration information
67        that modify, or inquire about, the overall configuration
68        of the Bridges MST resources.";
69        reference
70        "12.12 of IEEE Std 802.1Q-2018";
71        leaf-list mstid {
```



```
1         type dot1qtypes:mstid-type;
2         description
3             "The list of MSTID values that are currently supported
4             by the Bridge";
5     }
6     list fid-to-mstid {
7         key "fid";
8         description
9             "The FID to MSTID allocation table.";
10        reference
11            "12.12.2 of IEEE Std 802.1Q-2018";
12        leaf fid {
13            type uint32;
14            description
15                "The Filtering Database identifier.";
16            reference
17                "12.12.2 of IEEE Std 802.1Q-2018";
18        }
19        leaf mstid {
20            type dot1qtypes:mstid-type;
21            description
22                "The MSTID to which the FID is to be allocated.";
23            reference
24                "12.12.2 of IEEE Std 802.1Q-2018";
25        }
26    }
27    list fid-to-mstid-allocation {
28        key "fids";
29        description
30            "The FID to MSTID allocation table";
31        leaf fids {
32            type dot1qtypes:vid-range-type;
33            description
34                "Range of FIDs.";
35            reference
36                "12.12.2 of IEEE Std 802.1Q-2018";
37        }
38        leaf mstid {
39            type dot1qtypes:mstid-type;
40            description
41                "The MSTID to which the FID is allocated.";
42            reference
43                "12.12.2 of IEEE Std 802.1Q-2018";
44        }
45    }
46 }
47 }
48 }
49 }
50 }
51 }
52 }
53 }
54 }
```

```
55 augment "/if:interfaces/if:interface" {
56     when
57         "if:type = 'ianaif:bridge' or if:type = "+
58         "'ianaif:ethernetCsmacd' or if:type = 'ianaif:ieee8023adLag'"+
59         "or if:type = 'ianaif:ilan'" {
60         description
61             "Applies when a Bridge interface.";
62     }
63     description
64         "Augment the interface model with the Bridge Port";
65     container bridge-port {
66         description
67             "Bridge Port is an extension of the IETF Interfaces model
68             (RFC7223).";
69         leaf component-name {
70             type string;
71             description
72                 "Used to reference configured Component node.";
73         }
74         leaf port-type {
75             type identityref {
76                 base type-of-port;
77             }
78         }
79     }
80 }
```

```
1         description
2         "The port type. Indicates the capabilities of this port.";
3         reference
4         "12.4.2.1 of IEEE Std 802.1Q-2018";
5     }
6     leaf pvid {
7         when "../component-name != 'd-bridge-component'" {
8             description
9             "Applies to non TPMRs";
10        }
11        type dot1qtypes:vlan-index-type;
12        default "1";
13        description
14        "The primary (default) VID assigned to a specific Bridge
15        Port.";
16        reference
17        "12.10.1 of IEEE Std 802.1Q-2018
18        5.4, item m) of IEEE Std 802.1Q-2018";
19    }
20    leaf default-priority {
21        type dot1qtypes:priority-type;
22        default "0";
23        description
24        "The default priority assigned to a specific Bridge Port.";
25        reference
26        "12.6.2 of IEEE Std 802.1Q-2018";
27    }
28    container priority-regeneration {
29        description
30        "The Priority Regeneration Table parameters associated with
31        a specific Bridge Port. A list of Regenerated User
32        Priorities for each received priority on each port of a
33        Bridge. The regenerated priority value may be used to index
34        the Traffic Class Table for each input port. This only has
35        effect on media that support native priority. The default
36        values for Regenerated User Priorities are the same as the
37        User Priorities";
38        reference
39        "12.6.2 of IEEE Std 802.1Q-2018
40        6.9.4 of IEEE Std 802.1Q-2018";
41        uses dot1qtypes:priority-regeneration-table-grouping;
42    }
43    leaf pcp-selection {
44        type dot1qtypes:pcp-selection-type;
45        default "8POD";
46        description
47        "The Priority Code Point selection assigned to a specific
48        Bridge Port. This object identifies the rows in the PCP
49        encoding and decoding tables that are used to remark frames
50        on this port if this remarking is enabled";
51        reference
52        "12.6.2 of IEEE Std 802.1Q-2018
53        6.9.3 of IEEE Std 802.1Q-2018";
54    }
55    container pcp-decoding-table {
56        description
57        "The Priority Code Point Decoding Table parameters
58        associated with a specific Bridge Port.";
59        uses dot1qtypes:pcp-decoding-table-grouping;
60    }
61    container pcp-encoding-table {
62        description
63        "The Priority Code Point Encoding Table parameters
64        associated with a specific Bridge Port.";
65        uses dot1qtypes:pcp-encoding-table-grouping;
66    }
67    leaf use-dei {
68        type boolean;
69        default "false";
70        description
71        "The Drop Eligible Indicator. If it is set to True, then the
72        drop_eligible parameter is encoded in the DEI of transmitted
```

```
1         frames, and the drop_eligible parameter shall be true(1) for
2         a received frame if the DEI is set in the VLAN tag or the
3         Priority Code Point Decoding Table indicates drop_eligible
4         True for the received PCP value. If this parameter is False,
5         the DEI shall be transmitted as zero and ignored on receipt.";
6     reference
7         "12.6.2 of IEEE Std 802.1Q-2018
8         6.9.3 of IEEE Std 802.1Q-2018";
9 }
10 leaf drop-encoding {
11     type boolean;
12     default "false";
13     description
14         "The Drop Encoding parameter. If a Bridge supports encoding
15         or decoding of drop_eligible from the PCP field of a VLAN
16         tag (6.7.3) on any of its Ports, then it shall implement a
17         Boolean parameter Require Drop Encoding on each of its Ports
18         with default value False. If Require Drop Encoding is True
19         and the Bridge Port cannot encode particular priorities with
20         drop_eligible, then frames queued with those priorities and
21         drop_eligible True shall be discarded and not transmitted.";
22     reference
23         "12.6.2 of IEEE Std 802.1Q-2018
24         8.6.6 of IEEE Std 802.1Q-2018";
25 }
26 leaf service-access-priority-selection {
27     type boolean;
28     default "false";
29     description
30         "The Service Access Priority selection. Indication of
31         whether the Service Access Priority Selection function is
32         supported on the Customer Bridge Port to request priority
33         handling of the frame from a Port-based service interface.";
34     reference
35         "12.6.2 of IEEE Std 802.1Q-2018
36         6.13 of IEEE Std 802.1Q-2018";
37 }
38 container service-access-priority {
39     description
40         "The Service Access Priority table parameters. A table that
41         contains information about the Service Access Priority
42         Selection function for a Provider Bridge. The use of this
43         table enables a mechanism for a Customer Bridge attached to
44         a Provider Bridged Network to request priority handling of
45         frames.";
46     reference
47         "12.6.2 of IEEE Std 802.1Q-2018
48         6.13.1 of IEEE Std 802.1Q-2018";
49     uses dot1qtypes:service-access-priority-table-grouping;
50 }
51 container traffic-class {
52     description
53         "The Traffic Class table parameters. A table mapping
54         evaluated priority to Traffic Class, for forwarding by the
55         Bridge";
56     reference
57         "12.6.3 of IEEE Std 802.1Q-2018
58         8.6.6 of IEEE Std 802.1Q-2018";
59     uses dot1qtypes:traffic-class-table-grouping;
60 }
61 leaf acceptable-frame {
62     when "../component-name != 'd-bridge-component'" {
63         description
64             "Applies to non TPMRs";
65     }
66     type enumeration {
67         enum admit-only-VLAN-tagged-frames {
68             description
69                 "Admit only VLAN-tagged frames.";
70         }
71         enum admit-only-untagged-and-priority-tagged {
72             description
73                 "Admit only untagged and priority-tagged frames.";
74         }
75     }
76 }
```

```
1         "Admit only untagged and priority-tagged frames.";
2     }
3     enum admit-all-frames {
4         description
5             "Admit all frames.";
6     }
7     default "admit-all-frames";
8     description
9         "To configure the Acceptable Frame Types parameter
10        associated with one or more Ports";
11    reference
12        "12.10.1.3 of IEEE Std 802.1Q-2018
13        6.9 of IEEE Std 802.1Q-2018";
14    }
15    leaf enable-ingress-filtering {
16        when "../component-name != 'd-bridge-component'" {
17            description
18                "Applies to non TPMRs";
19        }
20        type boolean;
21        default "false";
22        description
23            "To enable the Ingress Filtering feature associated with one
24            or more Ports.";
25        reference
26            "12.10.1.4 of IEEE Std 802.1Q-2018
27            8.6.2 of IEEE Std 802.1Q-2018";
28    }
29    leaf enable-restricted-vlan-registration {
30        when "../component-name != 'd-bridge-component'" {
31            description
32                "Applies to non TPMRs";
33        }
34        type boolean;
35        default "false";
36        description
37            "To enable the Restricted VLAN Registration associated with
38            one or more Ports.";
39        reference
40            "11.2.3.2.3 of IEEE Std 802.1Q-2018
41            12.10.1.6 of IEEE Std 802.1Q-2018";
42    }
43    leaf enable-vid-translation-table {
44        when "../component-name != 'd-bridge-component'" {
45            description
46                "Applies to non TPMRs";
47        }
48        type boolean;
49        default "false";
50        description
51            "To enable VID Translation table associated with a Bridge
52            Port. This is not applicable to Bridge Ports that do no
53            support a VID Translation Table.";
54        reference
55            "12.10.1.8 of IEEE Std 802.1Q-2018
56            6.9 of IEEE Std 802.1Q-2018";
57    }
58    leaf enable-egress-vid-translation-table {
59        when "../component-name != 'd-bridge-component'" {
60            description
61                "Applies to non TPMRs";
62        }
63        type boolean;
64        default "false";
65        description
66            "To enable Egress VID Translation table associated with a
67            Bridge Port. This is not applicable to Ports that do not
68            support an Egress VID Translation table.";
69        reference
70            "12.10.1.9 of IEEE Std 802.1Q-2018
71            6.9 of IEEE Std 802.1Q-2018";
72    }
73    }
74    }
75    }
76    }
77    }
78    }
79    }
80    }
81    }
82    }
83    }
84    }
85    }
86    }
87    }
88    }
89    }
90    }
91    }
92    }
93    }
94    }
95    }
96    }
97    }
98    }
99    }
100   }
```

```
1      }
2      list protocol-group-vid-set {
3          when "../component-name != 'd-bridge-component'" {
4              description
5                  "Applies to non TPMRs";
6          }
7          if-feature "port-and-protocol-based-vlan";
8          key "group-id";
9          description
10             "The list of VID values associated with the Protocol Group
11             Identifier for this port.";
12         reference
13             "12.10.1.1.3 of IEEE Std 802.1Q-2018";
14         leaf group-id {
15             type uint32;
16             description
17                 "The protocol group identifier";
18             reference
19                 "12.10.1.7 of IEEE Std 802.1Q-2018";
20         }
21     }
22     leaf-list vid {
23         type dot1qttype:vlanid;
24         description
25             "The VLAN identifier to which this entry applies.";
26         reference
27             "12.10.2 of IEEE Std 802.1Q-2018";
28     }
29 }
30 leaf admin-point-to-point {
31     type enumeration {
32         enum force-true {
33             value 1;
34             description
35                 "Indicates that this port should always be treated as if
36                 it is connected to a point-to-point link.";
37         }
38         enum force-false {
39             value 2;
40             description
41                 "Indicates that this port should be treated as having a
42                 shared media connection.";
43         }
44         enum auto {
45             value 3;
46             description
47                 "Indicates that this port is considered to have a
48                 point-to-point link if it is an Aggregator and all of
49                 its members are aggregatable, or if the MAC entity is
50                 configured for full duplex operation, either through
51                 auto-negotiation or by management means.";
52         }
53     }
54     description
55         "For a port running spanning tree, this object represents
56         the administrative point-to-point status of the LAN segment
57         attached to this port, using the enumeration values of IEEE
58         Std 802.1AC. A value of forceTrue(1) indicates that this
59         port should always be treated as if it is connected to a
60         point-to-point link. A value of forceFalse(2) indicates that
61         this port should be treated as having a shared media
62         connection. A value of auto(3) indicates that this port is
63         considered to have a point-to-point link if it is an
64         Aggregator and all of its members are aggregatable, or if
65         the MAC entity is configured for full duplex operation,
66         either through auto-negotiation or by management means.
67         Manipulating this object changes the underlying
68         adminPointToPointMAC.";
69     reference
70         "12.4.2 of IEEE Std 802.1Q-2018
71         6.8.2 of IEEE Std 802.1Q-2018";
72 }
73 leaf protocol-based-vlan-classification {
```

```
1      when "../component-name != 'd-bridge-component'" {
2          description
3              "Applies to non TPMRs";
4      }
5      if-feature "port-and-protocol-based-vlan";
6      type boolean;
7      config false;
8      description
9          "A boolean indication indicating if Port-and-Protocol-based
10         VLAN classification is supported on a given Port.";
11     reference
12         "5.4.1.2 of IEEE Std 802.1Q-2018";
13 }
14 leaf max-vid-set-entries {
15     when "../component-name != 'd-bridge-component'" {
16         description
17             "Applies to non TPMRs";
18     }
19     if-feature "port-and-protocol-based-vlan";
20     type uint16;
21     config false;
22     description
23         "The maximum number of entries supported in the VID set on a
24         given Port.";
25     reference
26         "12.10.1.1.3 of IEEE Std 802.1Q-2018";
27 }
28 leaf port-number {
29     type dot1qtypes:port-number-type;
30     config false;
31     description
32         "An integer that uniquely identifies a Bridge Port.";
33     reference
34         "12.3, item i) of IEEE Std 802.1Q-2018
35         17.3.2.2 of IEEE Std 802.1Q-2018";
36 }
37 leaf address {
38     type ieee:mac-address;
39     config false;
40     description
41         "The specific MAC address of the individual MAC Entity
42         associated with the Port.";
43     reference
44         "12.4.2 of IEEE Std 802.1Q-2018
45         12.4.2.1.1.3, item a) of IEEE Std 802.1Q-2018";
46 }
47 leaf capabilities {
48     type bits {
49         bit tagging {
50             position "0";
51             description
52                 "Supports 802.1Q VLAN tagging of frames and MVRP.";
53         }
54         bit configurable-acceptable-frame-type {
55             position "1";
56             description
57                 "Allows modified values of acceptable frame types";
58         }
59         bit ingress-filtering {
60             position "2";
61             description
62                 "Supports the discarding of any frame received on a Port
63                 whose VLAN classification does not include that Port in
64                 its member set.";
65         }
66     }
67 }
68 config false;
69 description
70     "The feature capabilities associated with port. Indicates
71     the parts of IEEE 802.1Q that are optional on a per-port
72     basis, that are implemented by this device, and that are
73     manageable.";
```

```
1         reference
2         "12.10.1.1.3, item c) of IEEE Std 802.1Q-2018
3         12.4.2 of IEEE Std 802.1Q-2018";
4     }
5     leaf type-capabilities {
6         type bits {
7             bit customer-vlan-port {
8                 position "0";
9                 description
10                "Indicates the port can be a C-TAG aware port of an
11                enterprise VLAN aware Bridge";
12            }
13            bit provider-network-port {
14                position "1";
15                description
16                "Indicates the port can be an S-TAG aware port of a
17                Provider Bridge or Backbone Edge Bridge used for
18                connections within a PBN or PBBN.";
19            }
20            bit customer-network-port {
21                position "2";
22                description
23                "Indicates the port can be an S-TAG aware port of a
24                Provider Bridge or Backbone Edge Bridge used for
25                connections to the exterior of a PBN or PBBN.";
26            }
27            bit customer-edge-port {
28                position "3";
29                description
30                "Indicates the port can be a C-TAG aware port of a
31                Provider Bridge used for connections to the exterior of
32                a PBN or PBBN.";
33            }
34            bit customer-backbone-port {
35                position "4";
36                description
37                "Indicates the port can be a I-TAG aware port of a
38                Backbone Edge Bridge's B-component.";
39            }
40            bit virtual-instance-port {
41                position "5";
42                description
43                "Indicates the port can be a virtual S-TAG aware port
44                within a Backbone Edge Bridge's I-component which is
45                responsible for handling S-tagged traffic for a specific
46                backbone service instance.";
47            }
48            bit d-bridge-port {
49                position "6";
50                description
51                "Indicates the port can be a VLAN-unaware member of an
52                802.1Q Bridge.";
53            }
54            bit remote-customer-access-port {
55                position "7";
56                description
57                "Indicates the port can be an S-TAG aware port of a
58                Provider Bridge capable of providing Remote Customer
59                Service Interfaces.";
60            }
61            bit station-facing-bridge-port {
62                position "8";
63                description
64                "Indicates the station-facing Bridge Port in a EVB
65                Bridge.";
66            }
67            bit uplink-access-port {
68                position "9";
69                description
70                "Indicates the uplink access port in an EVB Bridge or
71                EVB station.";
72            }
73        }
74    }
```

```
1         bit uplink-relay-port {
2             position "10";
3             description
4                 "Indicates the uplink relay port in an EVB station.";
5         }
6         config false;
7         description
8             "The type of feature capabilities supported with port.
9             Indicates the capabilities of this port.";
10        reference
11            "12.4.2 of IEEE Std 802.1Q-2018";
12    }
13    leaf external {
14        type boolean;
15        config false;
16        description
17            "A boolean indicating whether the port is external. A value
18            of True means the port is external. A value of False means
19            the port is internal.";
20        reference
21            "12.4.2 of IEEE Std 802.1Q-2018";
22    }
23    leaf oper-point-to-point {
24        type boolean;
25        config false;
26        description
27            "For a port running spanning tree, this object represents
28            the operational point-to-point status of the LAN segment
29            attached to this port. It indicates whether a port is
30            considered to have a point-to-point connection.
31
32            If admin-point-to-point is set to auto(2), then the value of
33            oper-point-to-point is determined in accordance with the
34            specific procedures defined for the MAC entity concerned, as
35            defined in IEEE Std 802.1AC.
36
37            The value is determined dynamically; that is, it is
38            re-evaluated whenever the value of admin-point-to-point
39            changes, and whenever the specific procedures defined for
40            the MAC entity evaluate a change in its point-to-point
41            status.";
42        reference
43            "IEEE Std 802.1AC
44            12.4.2 of IEEE Std 802.1Q-2018";
45    }
46    container statistics {
47        config false;
48        description
49            "Container of operational state node information associated
50            with the bridge port.";
51        uses dot1qtypes:bridge-port-statistics-grouping;
52        leaf discard-on-ingress-filtering {
53            when "../component-name != 'd-bridge-component'" {
54                description
55                    "Applies to non TPMRs";
56            }
57            if-feature "ingress-filtering";
58            type yang:counter64;
59            description
60                "The number of frames that were discarded as a result of
61                Ingress Filtering being enabled.
62
63                Discontinuities in the value of this counter can occur at
64                re-initialization of the management system, and at other
65                times as indicated by the value of 'discontinuity-time'.";
66            reference
67                "12.6.1.1.3 of IEEE Std 802.1Q-2018";
68        }
69    }
70    list vid-translations {
71        when "../component-name != 'd-bridge-component'" {
```



```
1         description
2             "Applies to non TPMRs";
3     }
4     key "local-vid";
5     description
6         "To configure the VID Translation Table (6.9) associated
7         with a Port. This object is not applicable to Ports that do
8         not support a VID Translation Table. The default
9         configuration of the table has the value of the Relay VID
10        equal to the value of the Local VID. If no local VID is
11        configured, then it is assumed that the relay VID is the
12        same value as the local VID.
13
14        If the port supports an Egress VID translation table, the
15        VID Translation Configuration object configures the Local
16        VID to Relay VID mapping on ingress only. If an Egress VID
17        translation is not supported, the VID Translation
18        Configuration object defines a single bidirectional mapping.
19        In this case, the Bridge should not allow multiple keys
20        ('local-vid') mapped to the same 'relay-vid' value.";
21    leaf local-vid {
22        type dot1qttype:vlanid;
23        description
24            "The Local VID after translation received at the ISS or
25            EISS.";
26        reference
27            "12.10.1.8 of IEEE Std 802.1Q-2018
28            6.9 of IEEE Std 802.1Q-2018";
29    }
30    leaf relay-vid {
31        type dot1qttype:vlanid;
32        description
33            "The Relay VID received before translation received at ISS
34            or EISS.";
35        reference
36            "12.10.1.8 of IEEE Std 802.1Q-2018
37            6.9 of IEEE Std 802.1Q-2018";
38    }
39    list egress-vid-translations {
40        when "../component-name != 'd-bridge-component'" {
41            description
42                "Applies to non TPMRs";
43        }
44        key "relay-vid";
45        description
46            "To configure the Egress VID Translation Table (6.9)
47            associated with a Port. This object is not applicable to
48            Ports that do not support an Egress VID Translation Table.
49            The default configuration of the table has the value of the
50            Local VID equal to the value of the Relay VID. If no Relay
51            VID is configured, then it is assumed that the local VID is
52            the same value as the relay VID.";
53        leaf relay-vid {
54            type dot1qttype:vlanid;
55            description
56                "The Relay VID received before translation received at ISS
57                or EISS.";
58            reference
59                "12.10.1.9 of IEEE Std 802.1Q-2018
60                6.9 of IEEE Std 802.1Q-2018";
61        }
62        leaf local-vid {
63            type dot1qttype:vlanid;
64            description
65                "The Local VID after translation received at the ISS or
66                EISS.";
67            reference
68                "12.10.1.9 of IEEE Std 802.1Q-2018
69                6.9 of IEEE Std 802.1Q-2018";
70        }
71    }
72 }
```

```
1     }  
2   }  
3 }
```

48.7.4 Definitions for the ieee802-dot1q-tpmr YANG module

```
6 module ieee802-dot1q-tpmr {  
7   namespace urn:ieee:std:802.1Q:yang:ieee802-dot1q-tpmr;  
8   prefix dot1q-tpmr;  
9   import ieee802-dot1q-bridge {  
10    prefix dot1q;  
11  }  
12  import ietf-yang-types {  
13   prefix yang;  
14  }  
15  import ietf-interfaces {  
16   prefix if;  
17  }  
18  organization  
19    "IEEE 802.1 Working Group";  
20  contact  
21    "WG-URL: http://www.ieee802.org/1/  
22     WG-EMail: stds-802-1-L@ieee.org  
23  
24     Contact: IEEE 802.1 Working Group Chair  
25     Postal: C/O IEEE 802.1 Working Group  
26             IEEE Standards Association  
27             445 Hoes Lane  
28             P.O. Box 1331  
29             Piscataway  
30             NJ 08854  
31             USA  
32  
33     E-mail: STDS-802-1-L@IEEE.ORG";  
34  description  
35    "This YANG module describes the bridge configuration model for the  
36    Two Port MAC Relays.";  
37  revision 2018-03-07 {  
38    description  
39      "Published as part of IEEE Std 802.1Q-2018.  
40      Initial version.";  
41    reference  
42      "IEEE Std 802.1Q-2018, Bridges and Bridged Networks.";  
43  }  
44  augment "/if:interfaces/if:interface/dot1q:bridge-port" {  
45    when "dot1q:port-type = 'dot1q:d-bridge-port'" {  
46      description  
47        "Applies to TPMRs ports";  
48    }  
49    description  
50      "Augment Interface model with TPMR port configuration  
51      specific nodes.";  
52    leaf managed-address {  
53      type boolean;  
54      default "true";  
55      description  
56        "A Boolean value, which is TRUE if the MAC address is the  
57        management address for the TPMR, and is otherwise FALSE.  
58  
59        The TPMR management entity may make use of one or both Ports  
60        of a TPMR to transmit and receive management frames. However,  
61        the MAC address used by the TPMR management entity as the  
62        source MAC address in transmitted management frames (the  
63        management MAC address) is the individual MAC address  
64        associated with one of the Ports of the TPMR";  
65      reference  
66        "12.19.1.1.1.3 of IEEE Std 802.1Q-2018";  
67    }  
68    container mac-status-propagation {  
69      description
```

```
1      "MAC status propagation configuration node parameters.";
2  leaf link-notify {
3      type boolean;
4      default "true";
5      description
6          "The current value (Boolean) of LinkNotify (23.5.1) being
7          used by the MSP state machines.";
8      reference
9          "12.19.4.1.1.3 of IEEE Std 802.1Q-2018
10         12.19.4.1.2.2 of IEEE Std 802.1Q-2018";
11 }
12 leaf link-notify-wait {
13     type yang:timeticks {
14         range "20..100";
15     }
16     default "40";
17     description
18         "The current value, in centiseconds, of LinkNotifyWait
19         (23.5.2) being used by the MSP state machines.";
20     reference
21         "12.19.4.1.1.3 of IEEE Std 802.1Q-2018
22         12.19.4.1.2.2 of IEEE Std 802.1Q-2018";
23 }
24 leaf link-notify-retry {
25     type yang:timeticks {
26         range "10..100";
27     }
28     default "100";
29     description
30         "The current value, in centiseconds, of LinkNotifyRetry
31         (23.5.3) being used by the MSP state machines.";
32     reference
33         "12.19.4.1.1.3 of IEEE Std 802.1Q-2018
34         12.19.4.1.2.2 of IEEE Std 802.1Q-2018";
35 }
36 leaf mac-notify {
37     type boolean;
38     default "true";
39     description
40         "The current value (Boolean) of MACNotify (23.5.4) being
41         used by the MSP state machines.";
42     reference
43         "12.19.4.1.1.3 of IEEE Std 802.1Q-2018
44         12.19.4.1.2.2 of IEEE Std 802.1Q-2018";
45 }
46 leaf mac-notify-time {
47     type yang:timeticks {
48         range "1..50";
49     }
50     default "20";
51     description
52         "The current value, in centiseconds, of MACNotifyTime
53         (23.5.5) being used by the MSP state machines.";
54     reference
55         "12.19.4.1.1.3 of IEEE Std 802.1Q-2018
56         12.19.4.1.2.2 of IEEE Std 802.1Q-2018";
57 }
58 leaf mac-recover-time {
59     type yang:timeticks {
60         range "2..50";
61     }
62     default "10";
63     description
64         "The current value, in centiseconds, of MACRecoverTime
65         (23.5.6) being used by the MSP state machines.";
66     reference
67         "12.19.4.1.1.3 of IEEE Std 802.1Q-2018
68         12.19.4.1.2.2 of IEEE Std 802.1Q-2018";
69 }
70 }
71 }
72 augment
```

```
1      "/if:interfaces/if:interface/dot1q:bridge-port/dot1q:statistics" {
2      when "../dot1q:port-type = 'dot1q:d-bridge-port'" {
3          description
4              "Applies to TPMRs ports";
5      }
6      description
7          "Augment Interface model with TPMR port operational state
8          specific nodes.";
9      leaf acks-tx {
10         type yang:counter64;
11         config false;
12         description
13             "The number of acks transmitted (23.6.15) by the Ports
14             Transmit Process as a consequence of txAck being set.
15
16             Discontinuities in the value of this counter can occur at
17             re-initialization of the management system, and at other times
18             as indicated by the value of 'discontinuity-time.';
19         reference
20             "12.19.4.1.3.3 of IEEE Std 802.1Q-2018";
21     }
22     leaf add-notificatons-tx {
23         type yang:counter64;
24         config false;
25         description
26             "The number of adds transmitted (23.6.16) by the Ports
27             Transmit Process as a consequence of txAdd being set.
28
29             Discontinuities in the value of this counter can occur at
30             re-initialization of the management system, and at other times
31             as indicated by the value of 'discontinuity-time.';
32         reference
33             "12.19.4.1.3.3 of IEEE Std 802.1Q-2018";
34     }
35     leaf loss-notification-tx {
36         type yang:counter64;
37         config false;
38         description
39             "The number of losses transmitted (23.6.18) by the Ports
40             Transmit Process as a consequence of txLoss being set.
41
42             Discontinuities in the value of this counter can occur at
43             re-initialization of the management system, and at other times
44             as indicated by the value of 'discontinuity-time.';
45         reference
46             "12.19.4.1.3.3 of IEEE Std 802.1Q-2018";
47     }
48     leaf loss-confirmation-tx {
49         type yang:counter64;
50         config false;
51         description
52             "The number of loss confirms transmitted (23.6.19) by the
53             Ports Transmit Process as a consequence of txLossConfirm being
54             set.
55
56             Discontinuities in the value of this counter can occur at
57             re-initialization of the management system, and at other times
58             as indicated by the value of 'discontinuity-time.';
59         reference
60             "12.19.4.1.3.3 of IEEE Std 802.1Q-2018";
61     }
62     leaf acks-rx {
63         type yang:counter64;
64         config false;
65         description
66             "The number of acks received (23.6.10) by the Ports Transmit
67             Process.
68
69             Discontinuities in the value of this counter can occur at
70             re-initialization of the management system, and at other times
71             as indicated by the value of 'discontinuity-time.';
72         reference
```

```
1         "12.19.4.1.3.3 of IEEE Std 802.1Q-2018";
2     }
3     leaf add-notificatons-rx {
4         type yang:counter64;
5         config false;
6         description
7             "The number of adds received (23.6.11) by the Ports Receive
8             Process.
9             Discontinuities in the value of this counter can occur at
10            re-initialization of the management system, and at other times
11            as indicated by the value of 'discontinuity-time.'";
12            reference
13                "12.19.4.1.3.3 of IEEE Std 802.1Q-2018";
14        }
15    leaf loss-notification-rx {
16        type yang:counter64;
17        config false;
18        description
19            "The number of losses received (23.6.13) by the Ports Receive
20            Process.
21            Discontinuities in the value of this counter can occur at
22            re-initialization of the management system, and at other times
23            as indicated by the value of 'discontinuity-time.'";
24            reference
25                "12.19.4.1.3.3 of IEEE Std 802.1Q-2018";
26        }
27    leaf loss-confirmation-rx {
28        type yang:counter64;
29        config false;
30        description
31            "The number of loss confirms received (23.6.14) by the Ports
32            Receive Process.
33            Discontinuities in the value of this counter can occur at
34            re-initialization of the management system, and at other times
35            as indicated by the value of 'discontinuity-time.'";
36            reference
37                "12.19.4.1.3.3 of IEEE Std 802.1Q-2018";
38        }
39    leaf add-events {
40        type yang:counter64;
41        config false;
42        description
43            "The number of transitions to STM:ADD directly from STM:DOWN
44            or STM:LOSS (23.8).
45            Discontinuities in the value of this counter can occur at
46            re-initialization of the management system, and at other times
47            as indicated by the value of 'discontinuity-time.'";
48            reference
49                "12.19.4.1.3.3 of IEEE Std 802.1Q-2018";
50        }
51    leaf loss-events {
52        type yang:counter64;
53        config false;
54        description
55            "The number of transitions to STM:LOSS directly from STM:UP or
56            STM:ADD (23.8).
57            Discontinuities in the value of this counter can occur at
58            re-initialization of the management system, and at other times
59            as indicated by the value of 'discontinuity-time.'";
60            reference
61                "12.19.4.1.3.3 of IEEE Std 802.1Q-2018";
62        }
63    leaf mac-status-notifications {
64        type yang:counter64;
65        config false;
66        description
67            "The number of transitions to SNM:MAC_NOTIFICATION (23.9).
```

```
1         Discontinuities in the value of this counter can occur at
2         re-initialization of the management system, and at other times
3         as indicated by the value of 'discontinuity-time.'";
4     reference
5         "12.19.4.1.3.3 of IEEE Std 802.1Q-2018";
6     }
7 }
```

48.7.5 Definitions for the ieee802-dot1q-vlan-bridge YANG module

```
10 module ieee802-dot1q-vlan-bridge {
11     namespace urn:ieee:std:802.1Q:yang:ieee802-dot1q-vlan-bridge;
12     prefix dot1q-vlan-bridge;
13     organization
14         "IEEE 802.1 Working Group";
15     contact
16         "WG-URL: http://www.ieee802.org/1/
17         WG-EMail: stds-802-1-L@ieee.org
18
19         Contact: IEEE 802.1 Working Group Chair
20         Postal: C/O IEEE 802.1 Working Group
21                 IEEE Standards Association
22                 445 Hoes Lane
23                 P.O. Box 1331
24                 Piscataway
25                 NJ 08854
26                 USA
27
28         E-mail: STDS-802-1-L@IEEE.ORG";
29     description
30         "This YANG module describes the bridge configuration model for
31         Customer VLAN Bridges.";
32     revision 2018-03-07 {
33         description
34             "Published as part of IEEE Std 802.1Q-2018.
35             Initial version.";
36         reference
37             "IEEE Std 802.1Q-2018, Bridges and Bridged Networks.";
38     }
39 }
```

48.7.6 Definitions for the ieee802-dot1q-pb YANG module

```
40 module ieee802-dot1q-pb {
41     namespace urn:ieee:std:802.1Q:yang:ieee802-dot1q-pb;
42     prefix dot1q-pb;
43     import ieee802-dot1q-bridge {
44         prefix dot1q;
45     }
46     import ieee802-dot1q-types {
47         prefix dot1qtypes;
48     }
49     import ietf-interfaces {
50         prefix if;
51     }
52     organization
53         "IEEE 802.1 Working Group";
54     contact
55         "WG-URL: http://www.ieee802.org/1/
56         WG-EMail: stds-802-1-L@ieee.org
57
58         Contact: IEEE 802.1 Working Group Chair
59         Postal: C/O IEEE 802.1 Working Group
60                 IEEE Standards Association
61                 445 Hoes Lane
62                 P.O. Box 1331
63                 Piscataway
64                 NJ 08854
```

```
1           USA
2
3       E-mail: STDS-802-1-L@IEEE.ORG";
4   description
5       "This YANG module describes the bridge configuration model for
6       Provider Bridges.";
7   revision 2018-03-07 {
8       description
9           "Published as part of IEEE Std 802.1Q-2018.
10          Initial version.";
11      reference
12          "IEEE Std 802.1Q-2018, Bridges and Bridged Networks.";
13  }
14
15  augment "/if:interfaces/if:interface/dot1q:bridge-port" {
16      description
17          "Augment the interface model with 802.1Q Bridge Port
18          configuration specific nodes.";
19      leaf svid {
20          type dot1qtypes:vlanid;
21          description
22              "Service VLAN identifier.";
23          reference
24              "12.13.2.1 of IEEE Std 802.1Q-2018";
25      }
26      list cvid-registration {
27          when
28              "../dot1q:component-name = 'dot1q:c-vlan-component' and "+
29              "../dot1q:port-type = 'dot1q:customer-edge-port'" {
30              description
31                  "Applies when the component associated with this interface
32                  is a C-VLAN component and the port-type is a customer edge
33                  port.";
34          }
35          key "cvid";
36          description
37              "The C-VID Registration Table, provides a mapping between a
38              C-VID and the service instance represented by an S-VID
39              selected for that C-VLAN. This table provides the equivalent
40              functionality of
41              1) Configuring the PVID of the internal CNP on the S-VLAN
42              component
43              2) Adding the corresponding PEP on the C-VLAN component to
44              the member set of the C-VLAN
45              3) Adding the PEP and/or CEP to the untagged set of the
46              C-VLAN (if it is desired that frames forwarded to that
47              port are transmitted untagged for this C-VLAN).";
48          leaf cvid {
49              type dot1qtypes:vlanid;
50              description
51                  "Customer VLAN identifiers associated with this bridge port.";
52              reference
53                  "12.13.2.1 of IEEE Std 802.1Q-2018";
54          }
55          leaf svid {
56              type dot1qtypes:vlanid;
57              description
58                  "Service VLAN identifier.";
59              reference
60                  "12.13.2.1 of IEEE Std 802.1Q-2018";
61          }
62          leaf untagged-pep {
63              type boolean;
64              default "true";
65              description
66                  "A boolean indicating frames for this C-VLAN should be
67                  forwarded untagged through the Provider Edge Port.";
68              reference
69                  "12.13.2.1 of IEEE Std 802.1Q-2018";
70          }
71          leaf untagged-cep {
72              type boolean;
```

```
1         default "true";
2         description
3             "A boolean indicating frames for this C-VLAN should be
4             forwarded untagged through the Customer Edge Port.";
5         reference
6             "12.13.2.1 of IEEE Std 802.1Q-2018";
7     }
8 }
9 list service-priority-regeneration {
10     when
11         "../dot1q:component-name = 'dot1q:c-vlan-component' and "+
12         "../dot1q:port-type = 'dot1q:customer-edge-port'" {
13         description
14             "Applies when the component associated with this interface
15             is a C-VLAN component and the port-type is a customer edge
16             port.";
17     }
18     key "svid";
19     description
20         "The Service Priority Regeneration Table, which provides the
21         Priority Regeneration Table (12.6.2) for each internal CNP
22         connected to the C-VLAN component associated with the CEP.";
23     leaf svid {
24         type dot1qtypes:vlanid;
25         description
26             "Service VLAN identifier.";
27         reference
28             "12.13.2.6 of IEEE Std 802.1Q-2018";
29     }
30     container priority-regeneration {
31         description
32             "Contains Service Priority Regeneration table nodal
33             information.";
34         reference
35             "12.13.2.6 of IEEE Std 802.1Q-2018";
36         uses dot1qtypes:priority-regeneration-table-grouping;
37     }
38 }
39 list rcap-internal-interface {
40     when
41         "../dot1q:component-name = 'dot1q:s-vlan-component' and "+
42         "../dot1q:port-type = 'dot1q:remote-customer-access-port'" {
43         description
44             "Applies when the component associated with this interface
45             is a C-VLAN component and the port-type is a customer edge
46             port.";
47     }
48     key "external-svid";
49     description
50         "Designating an external port as an RCAP automatically creates
51         a Port-mapping S-VLAN component associated with that port.
52         This Port-mapping S-VLAN component includes one internal PNP.";
53     leaf external-svid {
54         type dot1qtypes:vlanid;
55         description
56             "External Service VLAN identifier.";
57         reference
58             "12.13.3.2 of IEEE Std 802.1Q-2018";
59     }
60     leaf internal-port-number {
61         type dot1qtypes:port-number-type;
62         description
63             "The number of the RCAP.";
64         reference
65             "12.13.3.2 of IEEE Std 802.1Q-2018";
66     }
67     leaf internal-svid {
68         type dot1qtypes:vlanid;
69         description
70             "Internal Service VLAN Identifier (not applicable for a
71             C-tagged RCSI).";
72         reference
```



```
1         "12.13.3.2 of IEEE Std 802.1Q-2018";
2     }
3     leaf internal-interface-type {
4         type enumeration {
5             enum port-based-rcsi {
6                 description
7                     "Port-based RCSI";
8             }
9             enum c-tagged-rcsi {
10                description
11                    "C-tagged RCSI";
12            }
13            enum pnp {
14                description
15                    "Provider Network Port";
16            }
17            enum discard {
18                description
19                    "Discard (external S-VID is not associated with an
20                    internal port).";
21            }
22        }
23    }
24 }
```

48.7.7 Definitions for the ieee802-dot1q-stream-filters-gates YANG module

```
25
26
27 module ieee802-dot1q-stream-filters-gates {
28     yang-version "1.1";
29     namespace urn:ieee:std:802.1Q:yang:ieee802-dot1q-stream-filters-gates;
30     prefix sfsg;
31     import ieee802-dot1q-bridge {
32         prefix dot1q;
33     }
34     organization
35         "IEEE 802.1 Working Group";
36     contact
37         "WG-URL: https://1.ieee802.org/
38         WG-EMail: stds-802-1@ieee.org
39         Contact: IEEE 802.1 Working Group Chair
40         Postal: C/O IEEE 802.1 Working Group
41                 IEEE Standards Association
42                 445 Hoes Lane
43                 P.O. Box 1331
44                 Piscataway
45                 NJ 08855-1331
46                 USA
47
48         E-mail: STDS-802-1-L@LISTSERV.IEEE.ORG";
49     description
50         "This module provides management of 802.1Q bridge components that
51         support Stream Filters and Stream Gates. NOTICE: This YANG module is
52         part of an unapproved IEEE Standards Draft and is subject to change.";
53     revision 2019-04-13 {
54         description
55             "Initial revision from IEEE P802.1Qcr.";
56         reference
```

```
1         "IEEE Std 802.1Qcr";
2     }
3     feature closed-gate-state {
4         description
5         "The bridge component supports gate state closed.";
6         reference
7         "IEEE Std 802.1Qcr";
8     }
9
10    /* Types and groupings */
11    typedef ipv-type {
12        type enumeration {
13            enum zero {
14                value 0;
15                description
16                "Priority 0";
17            }
18            enum one {
19                value 1;
20                description
21                "Priority 1";
22            }
23            enum two {
24                value 2;
25                description
26                "Priority 2";
27            }
28            enum three {
29                value 3;
30                description
31                "Priority 3";
32            }
33            enum four {
34                value 4;
35                description
36                "Priority 4";
37            }
38            enum five {
39                value 5;
40                description
41                "Priority 5";
42            }
43            enum six {
44                value 6;
45                description
46                "Priority 6";
47            }
48            enum seven {
49                value 7;
50                description
51                "Priority 7";
52            }
53            enum wildcard {
54                description
```

```
1         "No Priority";
2     }
3 }
4 description
5     "An IPV can be either of the following:
6         1) The null value. For a frame that passes through the gate, the
7            priority value associated with the frame is used to determine
8            the frame's traffic class, using the Traffic Class Table as
9            specified in 8.6.6.
10        2) An internal priority value. For a frame that passes through the
11           gate, the IPV is used, in place of the priority value
12           associated with the frame, to determine the frame's traffic
13           class, using the Traffic Class Table as specified in 8.6.6.";
14 reference
15     "8.6.5.2 of IEEE Std 802.1Qcr";
16 }
17 typedef gate-state-value-type {
18     type enumeration {
19         enum open {
20             description
21                 "Gate open";
22         }
23         enum closed {
24             description
25                 "Gate closed";
26         }
27     }
28     description
29         "The StreamGateStatesValue indicates the desired gate state, open or
30         closed, for the stream gate.";
31     reference
32         "12.31.3.2.1 of IEEE Std 802.1Qcr";
33 }
34 typedef stream-gate-ref {
35     type leafref {
36         path
37             '/dot1q:bridges'+
38             '/dot1q:bridge'+
39             '/dot1q:component'+
40             '/sfsg:stream-gates'+
41             '/sfsg:stream-gate-instance-table'+
42             '/sfsg:stream-gate-instance-id';
43     }
44     description
45         "This type is used to refer to a stream gate instance.";
46 }
47 augment "/dot1q:bridges/dot1q:bridge/dot1q:component" {
48     description
49         "Augments the Bridge component with Stream Filters and Stream Gates.";
50     container stream-filters {
51         description
52             "This container encapsulates all nodes related to Stream Filters.";
53         reference
54             "12.31.1 of IEEE Std 802.1Qcr
55             12.31.2 of IEEE Std 802.1Qcr
56             12.31.3 of IEEE Std 802.1Qcr";
57         list stream-filter-instance-table {
58             key "stream-filter-instance-id";
59             description
```

```
1      "There is one Stream Filter Instance Table per Bridge component.  
2      Each table row contains a set of parameters that defines a single  
3      Stream Filter (8.6.5.1), as detailed in Table 12-31. The table  
4      rows form an ordered list of filter instances, the order being  
5      determined by the StreamFilterInstance parameter. Tables can be  
6      created or removed dynamically in implementations that support  
7      dynamic configuration of Bridge components. Rows in the table can  
8      be created or removed dynamically in implementations that support  
9      dynamic configuration of stream filters. The value of the  
10     stream-handle-spec and priority-spec parameters associated with a  
11     received frame determine which stream filter is selected by the  
12     frame, and therefore what combination of filtering and policing  
13     actions is applied to the frame. If the stream-handle-spec and  
14     priority-spec parameters associated with a received frame match  
15     more than one stream filter, the stream filter that is selected  
16     is the one that appears earliest in the ordered list. If a  
17     received frame's stream-handle-spec and priority-spec does not  
18     match any of the stream filters in the table, the frame is  
19     processed as if Stream Filters and Stream Gates would not be  
20     supported.";  
21     reference  
22     "12.31.2 of IEEE Std 802.1Qcr";  
23     leaf stream-filter-instance-id {  
24         type uint32;  
25         mandatory true;  
26         description  
27         "An integer index value that determines the place of the stream  
28         filter in the ordered list of stream filter instances. The  
29         values are ordered according to their integer value; smaller  
30         values appear earlier in the ordered list.";  
31         reference  
32         "12.31.2.1 of IEEE Std 802.1Qcr";  
33     }  
34     choice stream-handle-spec {  
35         description  
36         "The stream_handle specification data type allows either of the  
37         following to be represented:  
38         a) A stream_handle value, represented as an integer.  
39         b) The wild card value, which matches any frame";  
40         reference  
41         "12.31.2.2 of IEEE Std 802.1Qcr";  
42  
43         /* NOTE: The mapping of the wildcard literal is  
44         *         other than in the MIB definition, where  
45         *         the wildcard value is mapped to -1.  
46         */  
47         case wildcard {  
48             leaf wildcard {  
49                 type empty;  
50                 description  
51                 "The stream handle specification represents a wild card  
52                 value.";  
53             }  
54         }  
55         case stream-handle {  
56             leaf stream-handle {  
57                 type uint32;  
58                 mandatory true;  
59                 description
```

```
1         "The stream handle specification refers to a stream_handle
2         value.";
3     }
4 }
5 }
6 leaf priority-spec {
7     type ipv-type;
8     mandatory true;
9     description
10        "The priority specification data type allows either of the
11        following to be represented:
12        a) A priority value, represented as an integer.
13        b) The wild card value, which matches any priority.";
14    reference
15        "12.31.2.3 of IEEE Std 802.1Qcr";
16 }
17 leaf stream-gate-ref {
18     type stream-gate-ref;
19     mandatory true;
20     description
21        "The StreamGateInstance parameter identifies the stream gate
22        (12.31.3) that is associated with the stream filter. The
23        relationship between stream filters and stream gates is many to
24        one; a given stream filter can be associated with only one
25        stream gate, but there can be multiple stream filters
26        associated with a given stream gate.";
27    reference
28        "12.31.2.4 of IEEE Std 802.1Qcr";
29 }
30 list filter-specification-list {
31     key "index";
32     description
33        "The filter specification list contains one or more filter
34        specifications that are assigned with this stream filter.";
35    reference
36        "12.31.2.5 of IEEE Std 802.1Qcr";
37    leaf index {
38        type uint8;
39        description
40            "The index of this filter specification.";
41    }
42    choice filter-specification {
43        description
44            "The filter specification type and its parameters.";
45        reference
46            "12.31.2.5 of IEEE Std 802.1Qcr
47            8.6.5.1 of IEEE Std 802.1Qcr
48            8.6.5.3 of IEEE Std 802.1Qcr";
49        case maximum-sdu-size {
50            description
51                "Maximum SDU size filter";
52            leaf maximum-sdu-size {
53                type uint32;
54                mandatory true;
55                description
56                    "The allowed maximum SDU size, in octets.";
57            }
58            leaf stream-blocked-due-to-oversize-frame-enabled {
59                type boolean;
```

```
1         default "false";
2         description
3             "A value of TRUE indicates that the
4             StreamBlockedDueToOversizeFrame function is enabled; a
5             value of FALSE indicates that the
6             StreamBlockedDueToOversizeFrame function is disabled. The
7             default value of StreamBlockedDueToOversizeFrameEnable is
8             FALSE.";
9         reference
10            "8.6.5.1 of IEEE Std 802.1Qcr
11            8.6.5.3.1 of IEEE Std 802.1Qcr";
12     }
13     leaf stream-blocked-due-to-oversize-frame {
14         type boolean;
15         default "false";
16         config false;
17         description
18             "If StreamBlockedDueToOversizeFrameEnable is TRUE, a
19             value of TRUE in StreamBlockedDueToOversizeFrame
20             indicates that all frames are to be dropped (i.e., the
21             behavior is identical as if the maximum SDU size would be
22             set to 0 octets). If StreamBlockedDueToOversizeFrame is
23             FALSE, it has no effect. The default value of
24             StreamBlockedDueToOversizeFrame is FALSE; if any frame is
25             discarded because it exceeds the maximum SDU size for the
26             stream, then StreamBlockedDueToOversizeFrame is set TRUE.";
27         reference
28            "8.6.5.1 of IEEE Std 802.1Qcr
29            8.6.5.3.1 of IEEE Std 802.1Qcr";
30     }
31 }
32 leaf max-stream-filter-instances {
33     type uint32;
34     config false;
35     description
36         "The maximum number of Stream Filter instances supported by this
37         Bridge component.";
38     reference
39         "12.31.1.1 of IEEE Std 802.1Qcr
40         8.6.5.1 of IEEE Std 802.1Qcr";
41 }
42 container stream-gates {
43     description
44         "This container encapsulates all nodes related to Stream Gates.";
45     list stream-gate-instance-table {
46         key "stream-gate-instance-id";
47         description
48             "There is one Stream Gate Instance Table per Bridge component.
49             Each table row contains a set of parameters that defines a single
50             Stream Gate (8.6.5.1.2), as detailed in Table 12-32. Tables can
51             be created or removed dynamically in implementations that support
52             dynamic configuration of Bridge components. Rows in the table can
53             be created or removed dynamically in implementations that support
54             dynamic configuration of stream gates.";
55         reference
```

```
1         "12.31.3 of IEEE Std 802.1Qcr";
2     leaf stream-gate-instance-id {
3         type uint32;
4         description
5             "An integer table index that allows the stream gate to be
6             referenced from Stream Filter Instance Table entries.";
7         reference
8             "12.31.2.4 of IEEE Std 802.1Qcr
9             8.6.5.1 of IEEE Std 802.1Qcr
10            8.6.5.2 of IEEE Std 802.1Qcr";
11    }
12    leaf gate-enable {
13        type boolean;
14        default "false";
15        description
16            "A Boolean variable that indicates whether the operation of the
17            state machines is enabled (TRUE) or disabled (FALSE). This
18            variable is set by management. The default value of this
19            variable is FALSE.";
20        reference
21            "8.6.9.4.14 of IEEE Std 802.1Q-2018";
22    }
23    leaf admin-gate-states {
24        type gate-state-value-type;
25        default "open";
26        description
27            "The administrative state associated with this gate, as set by
28            the management.";
29        reference
30            "12.31.3.2.1 of IEEE Std 802.1Qcr
31            8.6.10.4 of IEEE Std 802.1Qcr";
32    }
33    leaf admin-ipv {
34        type ipv-type;
35        default "wildcard";
36        description
37            "The administrative internal priority value specification.";
38        reference
39            "12.31.3.3 of IEEE Std 802.1Qcr
40            8.6.10.6 of IEEE Std 802.1Qcr
41            8.6.5.2 of IEEE Std 802.1Qcr";
42    }
43    }
44    leaf max-stream-gate-instances {
45        type uint32;
46        config false;
47        description
48            "The maximum number of Stream Gate instances supported by this
49            Bridge component.";
50        reference
51            "12.31.1.2 of IEEE Std 802.1Qcr
52            8.6.5.2 of IEEE Std 802.1Qcr";
53    }
54    }
55    }
56    }
57    }
58    }
59    }
60    }
61    }
62    }
63    }
64    }
65    }
66    }
67    }
68    }
69    }
70    }
71    }
72    }
73    }
74    }
75    }
76    }
77    }
78    }
79    }
80    }
81    }
82    }
83    }
84    }
85    }
86    }
87    }
88    }
89    }
90    }
91    }
92    }
93    }
94    }
95    }
96    }
97    }
98    }
99    }
100   }
```

<<Editor's Note: The discontinuous numbering of the previous and subsequent clause is intentional. A clause for PSFP appears between both (P802.1Qcw).>>

48.7.8 Definitions for the ieee802-dot1q-ats YANG module

```
1
2
3
4 module ieee802-dot1q-ats {
5     yang-version "1.1";
6     namespace urn:ieee:std:802.1Q:yang:ieee802-dot1q-ats;
7     prefix ats;
8     import ietf-yang-types {
9         prefix yang;
10    }
11    import ietf-interfaces {
12        prefix if;
13    }
14    import ieee802-dot1q-types {
15        prefix dot1qtypes;
16    }
17    import ieee802-dot1q-bridge {
18        prefix dot1q;
19    }
20    import ieee802-dot1q-stream-filters-gates {
21        prefix sfsg;
22    }
23    organization
24        "IEEE 802.1 Working Group";
25    contact
26        "WG-URL: https://1.ieee802.org/
27        WG-EMail: stds-802-1@ieee.org
28        Contact: IEEE 802.1 Working Group Chair
29        Postal: C/O IEEE 802.1 Working Group
30                IEEE Standards Association
31                445 Hoes Lane
32                P.O. Box 1331
33                Piscataway
34                NJ 08855-1331
35                USA
36
37        E-mail: STDS-802-1-L@LISTSERV.IEEE.ORG";
38    description
39        "This module provides management of 802.1Q bridge components that
40        support Asynchronous Traffic Shaping (ATS). NOTICE: This YANG module is
41        part of an unapproved IEEE Standards Draft and is subject to change.";
42    revision 2019-04-13 {
43        description
44            "Initial revision from IEEE P802.1Qcr.";
45        reference
46            "IEEE Std 802.1Qcr";
47    }
48    typedef scheduler-ref-type {
49        type leafref {
50            path
51                '/dot1q:bridges'+
52                '/dot1q:bridge'+
53                '/dot1q:component'+
54                '/ats:schedulers'+
55                '/ats:scheduler-instance-table'+
56                '/ats:scheduler-instance-id';
57        }
58        description
59            "This type is used to refer to an ATS scheduler instance.";
```



```
1     }
2     typedef scheduler-group-ref-type {
3         type leafref {
4             path
5                 '/dot1q:bridges'+
6                 '/dot1q:bridge'+
7                 '/dot1q:component'+
8                 '/ats:scheduler-groups'+
9                 '/ats:scheduler-group-instance-table'+
10                '/ats:scheduler-group-instance-id';
11        }
12        description
13            "This type is used to refer to an ATS scheduler group instance.";
14    }
15    augment
16        "/dot1q:bridges"+
17        "/dot1q:bridge"+
18        "/dot1q:component"+
19        "/sfsg:stream-filters"+
20        "/sfsg:stream-filter-instance-table"+
21        "/sfsg:filter-specification-list"+
22        "/sfsg:filter-specification" {
23        description
24            "Augments the Bridge component Stream Filter specification type by a
25            ATS scheduler filter specification type.";
26        case scheduler-ref {
27            leaf scheduler-ref {
28                type ats:scheduler-ref-type;
29                mandatory true;
30                description
31                    "A reference to the ATS scheduler associated with this filter.";
32            }
33        }
34    }
35    augment "/if:interfaces/if:interface/dot1q:bridge-port" {
36        description
37            "Augments Bridge Ports by ATS Per-Port Parameters";
38        container ats-port-parameters {
39            description
40                "This container comprises all ATS Per-Port Parameters.";
41            leaf discarded-frames-count {
42                type yang:counter64;
43                config false;
44                description
45                    "A counter of frames discarded by ATS scheduler instances
46                    associated with the Bridge Port";
47                reference
48                    "12.31.7.3 of IEEE Std 802.1Qcr";
49            }
50        }
51    }
52    augment "/dot1q:bridges/dot1q:bridge/dot1q:component" {
53        description
54            "Augments the Bridge component by
55            a) ATS Schedulers
56            b) ATS Scheduler Groups";
57        container schedulers {
58            description
59                "This container comprises all ATS scheduler instance related nodes.";
```

```
1      list scheduler-instance-table {
2          key "scheduler-instance-id";
3          description
4              "Each table row in the Scheduler Instance Table comprises a set
5              of parameters that defines a single ATS scheduler instance, as
6              detailed in 8.6.5.2.3.";
7          reference
8              "12.31.5 of IEEE Std 802.1Qcr";
9          leaf scheduler-instance-id {
10             type uint32;
11             mandatory true;
12             description
13                 "An integer table index that allows the scheduler instance to
14                 be referenced from Stream Filter Instance Table entries.";
15             reference
16                 "12.31.5.1 of IEEE Std 802.1Qcr
17                 8.6.5.3.3 of IEEE Std 802.1Qcr";
18         }
19         leaf committed-information-rate {
20             type uint64;
21             mandatory true;
22             description
23                 "The committed information rate parameter of the scheduler
24                 instance, in bits per second.";
25             reference
26                 "12.31.5.3 of IEEE Std 802.1Qcr
27                 8.6.5.3.3 of IEEE Std 802.1Qcr";
28         }
29         leaf committed-burst-size {
30             type uint32;
31             mandatory true;
32             description
33                 "The committed burst size parameter of the scheduler instance,
34                 in bits.";
35             reference
36                 "12.31.5.2 of IEEE Std 802.1Qcr
37                 8.6.5.3.3 of IEEE Std 802.1Qcr";
38         }
39         leaf scheduler-group-ref {
40             type ats:scheduler-group-ref-type;
41             mandatory true;
42             description
43                 "The SchedulerGroupInstanceID parameter identifies the
44                 scheduler group (12.32.5) that is associated with the scheduler
45                 instance. Multiple scheduler instances can be associated to one
46                 scheduler group, as detailed in 8.6.5.2.3.";
47             reference
48                 "12.31.6 of IEEE Std 802.1Qcr";
49         }
50     }
51     leaf max-scheduler-instances {
52         type uint32;
53         config false;
54         description
55             "The maximum number of scheduler instances supported by this
56             Bridge component.";
57         reference
58             "12.31.1.5 of IEEE Std 802.1Qcr
59             8.6.5.3.3 of IEEE Std 802.1Qcr";
```

```
1     }
2   }
3   container scheduler-groups {
4     description
5     "This container comprises all ATS scheduler group related nodes.";
6     list scheduler-group-instance-table {
7       key "scheduler-group-instance-id";
8       description
9       "Each table row in the Scheduler Group Instance Table comprises a
10      set of parameters that defines a single ATS scheduler group
11      instance (8.6.5.3.3).";
12      reference
13      "12.31.6 of IEEE Std 802.1Qcr
14      8.6.5.3.3 of IEEE Std 802.1Qcr";
15      leaf scheduler-group-instance-id {
16        type uint32;
17        description
18        "An integer table index that allows the scheduler group
19        instance to be referenced from Scheduler Instance Table
20        entries.";
21        reference
22        "12.31.6.1 of IEEE Std 802.1Qcr
23        8.6.5.3.3 of IEEE Std 802.1Qcr";
24      }
25      leaf max-residence-time {
26        type uint32;
27        mandatory true;
28        description
29        "The maximum residence time parameter of the scheduler group,
30        in nanoseconds.";
31        reference
32        "8.6.11.2.13 of IEEE Std 802.1Qcr
33        8.6.5.3.3 of IEEE Std 802.1Qcr";
34      }
35    }
36    leaf max-scheduler-group-instances {
37      type uint32;
38      config false;
39      description
40      "The maximum number of scheduler group instances supported by
41      this Bridge component.";
42      reference
43      "12.31.1.6 of IEEE Std 802.1Qcr
44      8.6.5.3.3 of IEEE Std 802.1Qcr";
45    }
46  }
47  container scheduler-timing-characteristics {
48    description
49    "This container comprises all ATS scheduler timing
50    characteristics related nodes.";
51    list scheduler-timing-characteristics-table {
52      key "reception-port transmission-port";
53      config false;
54      description
55      "Each row in this table comprises the timing characteristics of
56      a reception Port transmission Port pair, as detailed in Table
57      12-36.";
58      reference
59      "12.31.8 of IEEE Std 802.1Qcr
60      8.6.11 of IEEE Std 802.1Qcr";
```

```
1      leaf reception-port {
2          type dot1qtypes:port-number-type;
3          config false;
4          mandatory true;
5          description
6              "A reference to the associated reception Port.";
7          reference
8              "12.31.8.1 of IEEE Std 802.1Qcr";
9      }
10     leaf transmission-port {
11         type dot1qtypes:port-number-type;
12         config false;
13         mandatory true;
14         description
15             "A reference to the associated transmission Port.";
16         reference
17             "12.31.8.2 of IEEE Std 802.1Qcr";
18     }
19     leaf clock-offset-variation-max {
20         type uint32;
21         config false;
22         mandatory true;
23         description
24             "The maximum clock offset variation associated with the
25             reception Port transmission Port pair, in nanoseconds.";
26         reference
27             "12.31.8.3 of IEEE Std 802.1Qcr";
28     }
29     leaf clock-rate-deviation-max {
30         type uint32;
31         config false;
32         mandatory true;
33         description
34             "The maximum clock rate deviation associated with the
35             reception Port transmission Port pair, in ppm.";
36         reference
37             "12.31.8.4 of IEEE Std 802.1Qcr";
38     }
39     leaf arrival-recognition-delay-max {
40         type uint32;
41         config false;
42         mandatory true;
43         description
44             "The maximum arrival time recognition delay associated with
45             the reception Port transmission Port pair, in nanoseconds.";
46         reference
47             "12.31.8.5 of IEEE Std 802.1Qcr";
48     }
49     leaf processing-delay-min {
50         type uint32;
51         config false;
52         mandatory true;
53         description
54             "The minimum processing delay associated with the reception
55             Port transmission Port pair, in nanoseconds.";
56         reference
57             "12.31.8.6 of IEEE Std 802.1Qcr";
58     }
59     leaf processing-delay-max {
```

```
1         type uint32;  
2         config false;  
3         mandatory true;  
4         description  
5             "The maximum processing delay associated with the reception  
6             Port transmission Port pair, in nanoseconds.";  
7         reference  
8             "12.31.8.7 of IEEE Std 802.1Qcr";  
9     }  
10    }  
11    }  
12    }  
13 }
```

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54