

## Introduction

The initial version of this proposal was 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.

The second version of this proposal has been crafted jointly by the editor's of P802.1Qcr (Johannes Specht), P802.1Qcw (Marina Gutiérrez), and P802.1Qcx (Marc Holness). This version shows a complete overview of clause arrangement throughout clauses 48.4, 48.5, 48.6 and 48.7, which covers content under construction in all these amendment projects plus enhancement of missing contents required to cover IEEE Std 802.1Qcp-2018. The contents in this version give a complete picture of how these clauses would look like in the base document. The subsequent Table shows a suggested assignment of contents in this proposal to the aforementioned three projects and the maintenance group, which can decide to assign execution to one of the three projects.

**Table 0—Clause and project overview**

Clauses	Preliminary Editor Assignment (just for coordination between MH, MG, and JS)	Suggested Project
48.4 (introduction)	JS [done]	Maintainance
48.4.1 (Generic Bridge Model)	MH [done]	Maintainance
48.4.2 (TPMR Model)	MH [done]	Maintainance
48.4.3 (Customer VLAN Bridge Model)	MH [done]	Maintainance
48.4.4 (Provider Backbone Bridge Model)	MH [done]	Maintainance
48.4.5 (Stream Filters & Gates Model)	JS [done]	P802.1Qcr
48.4.6 (PSFP Model)	MG [done]	P802.1Qcw
48.4.7 (ATS Model)	JS [done]	P802.1Qcr
48.4.8 (Scheduled Traffic Model)	MG	P802.1Qcw
48.4.9 (Preemption Model)	MG	P802.1Qcw
48.4.10 (CFM Model)	MH	P802.1Qcx
48.5 (introduction)	JS	Maintainance
48.6 (introduction)	JS [done]	Maintainance

**Table 0—Clause and project overview**

Clauses	Preliminary Editor Assignment (just for coordination between MH,MG, and JS)	Suggested Project
48.7 (introduction)	JS [done]	Maintainance
48.5.1, 48.6.1, 48.7.1 ( <b>ieee802-types YANG module</b> )	MH ["clever" statements saying we don't do data tree and Managed Object table here]	Maintainance (IEEE Std 802.1Qcp-2018)
48.5.2, 48.6.2, 48.7.2 ( <b>ieee802-dot1q-types YANG module</b> )	MH ["clever" statements saying we don't do data tree and Managed Object-table here]	Maintainance (IEEE Std 802.1Qcp-2018)
48.5.3, 48.6.3, 48.7.3 ( <b>ieee802-dot1q-bridge YANG module</b> )	MH [done]	Maintainance (IEEE Std 802.1Qcp-2018)
48.5.4, 48.6.4, 48.7.4 ( <b>ieee802-dot1q-tpmr YANG module</b> )	MH [Managed Object table??]	Maintainance (IEEE Std 802.1Qcp-2018)
48.5.5, 48.6.5, 48.7.5 ( <b>ieee802-dot1q-vlan-bridge YANG module</b> )	MH [done]	Maintainance (IEEE Std 802.1Qcp-2018)
48.5.6, 48.6.6, 48.7.6 ( <b>ieee802-dot1q-pb YANG module</b> )	MH [Managed Object table??]	Maintainance (IEEE Std 802.1Qcp-2018)
48.5.7, 48.6.7, 48.7.7 ( <b>ieee802-dot1q-stream-filters-gates YANG module</b> )	JS [done]	P802.1Qcr
48.5.8, 48.6.8, 48.7.8 ( <b>ieee802-dot1q-psfp YANG module</b> )	MG	P802.1Qcw
48.5.9, 48.6.9, 48.7.9 ( <b>ieee802-dot1q-ats YANG module</b> )	JS [done]	P802.1Qcr
48.5.10, 48.6.10, 48.7.10 ( <b>ieee802-dot1q-sched YANG module</b> )	MG	P802.1Qcw
48.5.11, 48.6.11, 48.7.11 ( <b>ieee802-dot1q-preemption YANG module</b> )	MG	P802.1Qcw
48.5.12, 48.6.12, 48.7.12 ( <b>ieee802-dot1q-cfm-types YANG module</b> )	MH	P802.1Qcx
48.5.13, 48.6.13, 48.7.13 ( <b>ieee802-dot1q-cfm YANG module</b> )	MH	P802.1Qcx
48.5.14, 48.6.14, 48.7.14 ( <b>ieee802-dot1q-cfm-bridge YANG module</b> )	MH	P802.1Qcx
48.5.15, 48.6.15, 48.7.15 ( <b>ieee802-dot1q-cfm-alarms YANG module</b> )	MH	P802.1Qcx

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) Subclauses related to the ieee802-dot1q-vlan-bridge YANG module may be remove due because the module is empty. Subsequent clauses at the same level will be re-numbered accordingly.
- d) 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.
- e) 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.

<<Editor's Note: The table will be broken into rows and distributed to projects accordingly.>>

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 VLAN Bridge YANG model, Provider Bridge model).

**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-psfp	48.5.8, 48.6.8, 48.7.8	PSFP-specific extensions to the ieee802-dot1q-stream-filters-gates and ieee802-dot1q-bridge modules.
ieee802-dot1q-ats	48.5.9, 48.6.9, 48.7.9	ATS-specific extensions to the ieee802-dot1q-stream-filters-gates and ieee802-dot1q-bridge modules.
ieee802-dot1q-sched	48.5.10, 48.6.10, 48.7.10	N/A
ieee802-dot1q-preemption	48.5.11, 48.6.11, 48.7.11	N/A
ieee802-dot1q-cfm-types	48.5.12, 48.6.12, 48.7.12	N/A
ieee802-dot1q-cfm	48.5.13, 48.6.13, 48.7.13	N/A
ieee802-dot1q-cfm-bridge	48.5.14, 48.6.14, 48.7.14	N/A
ieee802-dot1q-cfm-alarms	48.5.15, 48.6.15, 48.7.15	N/A

1 A system implementing the generic Bridge model implements the YANG modules as described in Table 48-  
2 2.

3  
4  
5 **Table 48-2—YANG module dependencies for the generic Bridge model**

6  
7  
8  
9  
10  
11  
12  
13  
14

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

15  
16 **48.4.2 Two-Port MAC Relay model**

17  
18 A system implementing the TPMR YANG model (48.3.2.1) implements the YANG modules as described in  
19 Table 48-3.

20  
21  
22 **Table 48-3—YANG module dependencies for the Two-Port MAC Relay model**

23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33

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

34  
35 **48.4.3 Customer VLAN Bridge model**

36  
37 A system implementing the Customer VLAN Bridge YANG model (48.3.2.2) implements the YANG  
38 modules as described in Table 48-4.

39  
40  
41 **Table 48-4—YANG module dependencies for the Customer VLAN Bridge model**

42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54

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

#### 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>	—
<i>ieee802-dot1q-types</i>	—
<i>ieee802-dot1q-bridge</i>	—
<i>ieee802-dot1q-pb</i>	—

#### 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 in Table 48-6.

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

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

#### 48.4.6 Per-Stream Filtering and Policing (PSFP) model

A system implementing the PSFP model (48.3.3) implements the YANG modules as described in Table 48-7.

**Table 48-7—YANG module dependencies for the PSFP model**

YANG module	Notes
<i>ieee802-types</i>	—
<i>ieee802-dot1q-types</i>	—
<i>ieee802-dot1q-bridge</i>	—
<i>ieee802-dot1q-stream-filters-gates</i>	—
<i>ieee802-dot1q-psfp</i>	—

#### 48.4.7 Asynchronous Traffic Shaping (ATS) model

A system implementing the ATS model (48.3.4) implements the YANG modules as described in Table 48-8.

**Table 48-8—YANG module dependencies for the ATS model**

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

#### 48.4.8 Scheduled Traffic model

#### 48.4.9 Frame Preemption model

#### 48.4.10 CFM model

### 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

1 Access Control (MAC) Bridges, Two-Port MAC Relays (TPMRs), Customer Virtual Local Area Network  
2 (VLAN) Bridges, and Provider Bridges (as specified by this standard) with the capabilities currently  
3 specified in 12.4 to 12.8, 12.10, 12.13, and 12.19 of this standard.

4  
5 In support of this standard, the YANG data model extends the IETF Interface Management YANG model (as  
6 specified in IETF RFC 8343).

7  
8  
9 The Bridge Port YANG node augments the Interface Management YANG model. The specific Bridge (e.g.,  
10 TPMR, Customer VLAN, Provider Bridge) YANG models are augmentations from the Bridge YANG  
11 model. A system implementing these YANG models shall implement the *ieee802-dot1q-bridge*,  
12 *ieee802-types*, and *ieee802-dot1q-types* YANG models.

### 13 14 **48.5.1 Relationship of the ieee802-types YANG module**

15  
16 N/A

17  
18  
19 <<Editor's Note: The treatment of ieee802-types YANG module needs discussion (e.g., scope?!), given that it  
20 contains IEEE802-wide types. However, These are simple and we may omit descriptions for these in this  
21 clause.>>

### 22 23 **48.5.2 Relationship of the ieee802-dot1q-types YANG module**

24  
25 N/A

26  
27  
28 <<Editor's Note: Contents of ieee802-dot1q-types are simple, we may omit descriptions for these in this  
29 clause. Alternatively (which may even be better), we may craft an explicit text that can be copied into all new  
30 level 3 subclauses of this clause (48.5) in case the relationship table is not provided (i.e., stating the criteria,  
31 etc.)>>

### 32 33 **48.5.3 Relationship of the ieee802-dot1q-bridge YANG module**

34  
35 <<I've made some minor tweaks, however, the base content is copied from multiple tables in IEEE 802.1Qcp-  
36 2018. The tweaks are as follows:  
37 - read/write attributes ("r-w" and "r") removed (already visible at other places and thus less prone to break,  
38 MIB tables don't show these either)  
39 - slight adjustments to the bold cell contents in the left column  
40 - Indentations simplified>>

41  
42 **Table 48-9—Cross-reference table of the ieee802-dot1q-bridge YANG module**

43 44 <b>Bridge management information</b>	45 <b>YANG node(s)</b>
46 <b>* Bridge</b>	<b>ieee802-dot1q-bridge:bridges:bridge</b>
47 name (12.4)	name — KEY
48 address (12.4)	address
49 type	bridge-type
50 ports (12.4)	ports
51 upTime (12.4)	up-time
52 components (12.3)	components



**Table 48-9—Cross-reference table of the ieee802-dot1q-bridge YANG module (continued)**

Bridge management information	YANG node(s)
<b>* Bridge Component</b>	<b>ieee802-dot1q-bridge:bridges:bridge:component</b>
—	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
<b>Bridge Component Capabilities</b>	<b>ieee802-dot1q-bridge:bridges:bridge:component:capabilities</b>
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
<b>Filtering Database</b>	<b>ieee802-dot1q-bridge:bridges:bridge:component:filtering-database</b>
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
<b>Filtering Entries</b>	<b>ieee802-dot1q-bridge:bridges:bridge:component:filtering-database:filtering-entries</b>
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

**Table 48-9—Cross-reference table of the ieee802-dot1q-bridge YANG module (continued)**

Bridge management information	YANG node(s)
<b>VLAN Registration Entries</b>	<b>ieee802-dot1q-bridge:bridges:bridge:component:filtering-database:vlan-registration-entries</b>
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
<b>Permanent Database</b>	<b>ieee802-dot1q-bridge:bridges:bridge:component:permanent-database</b>
size (12.7.6)	size
staticEntries (12.7.6)	static-entries
staticVlanRegistrationEntries (12.7.6)	static-vlan-registration-entries
<b>Permanent Filtering Entries</b>	<b>ieee802-dot1q-bridge:bridges:bridge:component:permanent-database:filtering-entries</b>
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
<b>Bridge VLAN</b>	<b>ieee802-dot1q-bridge:bridges:bridge:component:bridge-vlan</b>
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
<b>Bridge VLAN ID Entries</b>	<b>ieee802-dot1q-bridge:bridges:bridge:component:bridge-vlan:vlan-id</b>
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
<b>Protocol Group Database</b>	<b>ieee802-dot1q-bridge:bridges:bridge:component:bridge-vlan:protocol-group-database</b>
frameFormatType (12.10.1.7)	frame-format-type
protocolGroupId (6.12.2)	protocol-group-id
<b>VID to FID</b>	<b>ieee802-dot1q-bridge:bridges:bridge:component:bridge-vlan:vid-to-fid</b>
vid (12.10.3.4)	vid
fid (12.10.3.4)	fid

**Table 48-9—Cross-reference table of the ieee802-dot1q-bridge YANG module (continued)**

Bridge management information	YANG node(s)
<b>VID to FID Allocations</b>	<b>ieee802-dot1q-bridge:bridges:bridge:component:bridge-vlan:vid-to-fid-allocation</b>
vid (12.10.3.2)	vid — KEY
fid (12.10.3).2	fid
allocationType (12.10.3.2)	allocation-type
<b>FID to VID Allocations</b>	<b>ieee802-dot1q-bridge:bridges:bridge:component:bridge-vlan:fid-to-vid-allocation</b>
fid (12.10.3.3)	fid — KEY
* vid (12.10.3)	* vid
* allocationType (12.10.3)	* allocation-type
<b>Bridge MST</b>	<b>ieee802-dot1q-bridge:bridges:bridge:component:bridge-mst</b>
* MSTID (12.12.1)	<b>ieee802-dot1q-bridge:bridges:bridge:component:bridge-mst:mstid</b>
<b>FID to MSTID</b>	<b>ieee802-dot1q-bridge:bridges:bridge:component:bridge-mst:fid-to-mstid</b>
fid (12.12.2)	fid — KEY
mstid (12.12.2)	mstid
<b>FID to MSTID Allocation</b>	<b>ieee802-dot1q-bridge:bridges:bridge:component:bridge-mst:fid-to-mstid-allocation</b>
fids (12.12.2)	fids — KEY
mstid (12.12.2)	mstid

**Table 48-9—Cross-reference table of the ieee802-dot1q-bridge YANG module (continued)**

Bridge management information	YANG node(s)
<b>Bridge Port</b>	<b>ietf-interfaces:interfaces:interface:ieee802-dot1q-bridge:bridge-port</b>
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
<u>mediaDependentOverhead (12.4.2)</u>	<u>media-dependent-overhead</u>

**Table 48-9—Cross-reference table of the ieee802-dot1q-bridge YANG module (continued)**

Bridge management information	YANG node(s)
<b>Bridge Port Statistics</b>	<b>ietf-interfaces:interfaces:interface:ieee802-dot1q-bridge:bridge-port:statistics</b>
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)
<b>Stream Filters</b>	<b>ieee802-dot1q-bridge:bridges:bridge:component:stream-filters</b>
MaxStreamFilterInstances (12.31.1.1)	max-stream-filter-instances
<b>Stream Filter Instance Table (Table 12-32)</b>	<b>ieee802-dot1q-bridge:bridges:bridge:component:stream-filters:stream-filter-instance-table</b>
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 12-32)	stream-gate-ref
<b>FilterSpecificationList (12.31.2.5)</b>	<b>ieee802-dot1q-bridge:bridges:bridge:component:stream-filters:stream-filter-instance-table:filter-specification-list</b>
—	index—KEY
<b>Maximum SDU Size Filters</b>	<b>ieee802-dot1q-bridge:bridges:bridge:component:stream-filters:stream-filter-instance-table:filter-specification-list:maximum-sdu-size</b>
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
<b>Stream Gates</b>	<b>ieee802-dot1q-bridge:bridges:bridge:component:stream-gates</b>
MaxStreamGateInstances (12.31.1.2)	max-stream-gate-instances
<b>Stream Gate Instance Table (Table 12-33)</b>	<b>ieee802-dot1q-bridge:bridges:bridge:component:stream-gates:stream-gate-instance-table</b>
StreamGateInstance (12.31.2.4)	stream-gate-instance-id—KEY
StreamGateEnabled (Table 12-33)	stream-handle-spec
PrioritySpec (Table 12-33,12.31.2.3)	priority-spec
AdminGateStates (Table 12-33)	admin-gate-states
AdminIPV (12.31.2.5)	admin-ipv

1           **48.5.8 Relationship of the ieee802-dot1q-psfp YANG module**

2  
3           N/A

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

**48.5.9 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)
<b>ATS Schedulers</b>	<b>ieee802-dot1q-bridge:bridges:bridge:component:schedulers</b>
MaxSchedulerInstances (12.31.1.5)	max-scheduler-instances
<b>Scheduler Instance Table (Table 12-35)</b>	<b>ieee802-dot1q-bridge:bridges:bridge:component:schedulers:scheduler-instance-table</b>
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
<b>ATS Scheduler Groups</b>	<b>ieee802-dot1q-bridge:bridges:bridge:component:scheduler-groups</b>
MaxSchedulerGroupInstances (12.31.1.6)	max-scheduler-group-instances
<b>Scheduler Group Instance Table (Table 12-36)</b>	<b>ieee802-dot1q-bridge:bridges:bridge:component:scheduler-groups:scheduler-group-instance-table</b>
SchedulerGroupInstanceID (12.31.6.1)	scheduler-group-instance-id—KEY
MaxResidenceTime (12.31.6.2)	max-residence-time
<b>ATS Scheduler Timing Characteristics Table (Table 12-38)</b>	<b>ieee802-dot1q-bridge:bridges:bridge:component:scheduler-timing-characteristics:scheduler-timing-characteristics-table</b>
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



1 **48.5.10 Relationship of the ieee802-dot1q-sched YANG module**

2  
3 N/A

4  
5 **48.5.11 Relationship of the ieee802-dot1q-preemption YANG module**

6  
7 N/A

8  
9 **48.5.12 Relationship of the ieee802-dot1q-cfm-types YANG module**

10  
11 N/A

12  
13 **48.5.13 Relationship of the ieee802-dot1q-cfm YANG module**

14  
15 N/A

16  
17 **48.5.14 Relationship of the ieee802-dot1q-cfm-bridge YANG module**

18  
19 N/A

20  
21 **48.5.15 Relationship of the ieee802-dot1q-cfm-alarms YANG module**

22  
23 N/A

24  
25 **48.6 YANG data scheme tree definitions**

26  
27 A simplified graphical representation of the data model is used in this document. The meaning of the  
28 symbols in these diagrams is as follows:  
29

- 30 — Brackets “[” and “]” enclose list keys.  
31 — Abbreviations before data node names: “rw” means configuration (read-write), and “ro” means state  
32 data (read-only).  
33 — Symbols after data node names: “?” means an optional node, “!” means a presence container, and “\*”  
34 denotes a list and leaf-list.  
35 — Parentheses enclose choice and case nodes, and case nodes are also marked with a colon (“:”).  
36

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

38  
39 **48.6.1 Data scheme definition for the ieee802-types YANG module**

40  
41 N/A

42  
43 **48.6.2 Data scheme definition for the ieee802-dot1q-types YANG module**

44  
45 N/A

46  
47 **48.6.3 Data scheme definition for the ieee802-dot1q-bridge YANG module**

48  
49  
50 module: ieee802-dot1q-bridge  
51     +--rw bridges  
52         +--rw bridge\* [name]             dot1qtypes:name-type  
53             +--rw address                 ieee:mac-address  
54             +--rw bridge-type             identityref  
              +--ro ports?                 uint16

```
1      +--ro up-time?          yang:zero-based-counter32
2      +--ro components?      uint32
3      +--rw component* [name]
4          +--rw name          string
5          +--rw id?           uint32
6          +--rw type          identityref
7          +--rw address?      ieee:mac-address
8          +--rw traffic-class-enabled? boolean
9          +--ro ports?        uint16
10         +--ro bridge-port*  if:interface-ref
11         +--ro capabilities
12             | +--ro extended-filtering?    boolean
13             | +--ro traffic-classes?      boolean
14             | +--ro static-entry-individual-port? boolean
15             | +--ro ivl-capable?         boolean
16             | +--ro svl-capable?        boolean
17             | +--ro hybrid-capable?     boolean
18             | +--ro configurable-pvid-tagging? boolean
19             | +--ro local-vlan-capable?  boolean
20         +--rw filtering-database
21             | +--rw aging-time?          uint32
22             | +--ro size?                yang:gauge32
23             | +--ro static-entries?      yang:gauge32
24             | +--ro dynamic-entries?     yang:gauge32
25             | +--ro static-vlan-registration-entries? yang:gauge32
26             | +--ro dynamic-vlan-registration-entries? yang:gauge32
27             | +--ro mac-address-registration-entries? yang:gauge32 {extended-
28 filtering-services}?
29             | +--rw filtering-entry* [database-id vids address]
30             | | +--rw database-id      uint32
31             | | +--rw address          ieee:mac-address
32             | | +--rw vids             dot1qtypes:vid-range-type
33             | | +--rw entry-type?      enumeration
34             | | +--rw port-map* [port-ref]
35             | | | +--rw port-ref          port-number-type
36             | | | +--rw (map-type)?
37             | | | +--:(static-filtering-entries)
38             | | | | +--rw static-filtering-entries
39             | | | | | +--rw control-element? enumeration
40             | | | | | +--rw connection-identifier? port-number-type
41             | | | +--:(static-vlan-registration-entries)
42             | | | | +--rw static-vlan-registration-entries
43             | | | | | +--rw registrar-admin-control? enumeration
44             | | | | | +--rw vlan-transmitted? enumeration
45             | | | +--:(mac-address-registration-entries)
46             | | | | +--rw mac-address-registration-entries
47             | | | | | +--rw control-element? enumeration
48             | | | +--:(dynamic-vlan-registration-entries)
49             | | | | +--rw dynamic-vlan-registration-entries
50             | | | | | +--rw control-element? enumeration
51             | | | +--:(dynamic-reservation-entries)
52             | | | | +--rw dynamic-reservation-entries
53             | | | | | +--rw control-element? enumeration
54             | | | +--:(dynamic-filtering-entries)
55             | | | | +--rw dynamic-filtering-entries
56             | | | | | +--rw control-element? enumeration
57             | | +--ro status?          enumeration
58             +--rw vlan-registration-entry* [database-id vids]
59             | +--rw database-id      uint32
60             | +--rw vids             dot1qtypes:vid-range-type
61             | +--rw entry-type?      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 permanent-database
11     | +---ro size?                               yang:gauge32
12     | +---ro static-entries?                     yang:gauge32
13     | +---ro static-vlan-registration-entries?  yang:gauge32
14     | +---rw filtering-entry* [database-id vids address]
15     |     +---rw database-id      uint32
16     |     +---rw address          ieee:mac-address
17     |     +---rw vids             dot1qtypes:vid-range-type
18     |     +---ro status?          enumeration
19     |     +---rw port-map* [port-ref]
20     |         +---rw port-ref          port-number-type
21     |         +---rw (map-type)?
22     |             +---:(static-filtering-entries)
23     |             | +---rw static-filtering-entries
24     |             |           +---rw control-element?  enumeration
25     |             |           +---rw connection-identifier? port-number-type
26     |             +---:(static-vlan-registration-entries)
27     |             | +---rw static-vlan-registration-entries
28     |             |           +---rw registrar-admin-control? enumeration
29     |             |           +---rw vlan-transmitted?  enumeration
30     |             +---:(mac-address-registration-entries)
31     |             | +---rw mac-address-registration-entries
32     |             |           +---rw control-element?  enumeration
33     |             +---:(dynamic-vlan-registration-entries)
34     |             | +---rw dynamic-vlan-registration-entries
35     |             |           +---rw control-element?  enumeration
36     |             +---:(dynamic-reservation-entries)
37     |             | +---rw dynamic-reservation-entries
38     |             |           +---rw control-element?  enumeration
39     |             +---:(dynamic-filtering-entries)
40     |             | +---rw dynamic-filtering-entries
41     |             |           +---rw control-element?  enumeration
42     +---rw bridge-vlan
43     | +---ro version?                          uint16
44     | +---ro max-vids?                          uint16
45     | +---ro override-default-pvid?            boolean
46     | +---ro protocol-template?                dot1qtypes:protocol-frame-format-
47     type {port-and-protocol-based-vlan}?
48     | +---ro max-msti?                          uint16
49     | +---rw vlan* [vid]
50     | | +---rw vid                              dot1qtypes:vlan-index-type
51     | | +---rw name?                            dot1qtypes:name-type
52     | | +---ro untagged-ports*                  if:interface-ref
53     | | +---ro egress-ports*                   if:interface-ref
54     | +---rw protocol-group-database* [db-index] {port-and-protocol-
55     based-vlan}?
56     | | +---rw db-index                          uint16
57     | | +---rw frame-format-type?                dot1qtypes:protocol-frame-format-
58     type
59     | | +---rw (frame-format)?
60     | | | +---:(ethernet-rfc1042-snap8021H)
61     | | | | +---rw ethertype?                    dot1qtypes:ethertype-type
62     | | | +---:(snap-other)
63     | | | | +---rw protocol-id?                  string
64     | | | +---:(llc-other)
65     | | | | +---rw dsap-ssap-pairs
66     | | | |           +---rw llc-address?        string
67     | | +---rw group-id?                          uint32
68     +---rw vid-to-fid-allocation* [vids]
69     | +---rw vids                                dot1qtypes:vid-range-type

```

```
1         | | +--ro fid?                uint32
2         | | +--ro allocation-type?  enumeration
3         | | +--rw fid-to-vid-allocation* [fid]
4         | | +--rw fid                uint32
5         | | +--ro allocation-type?  enumeration
6         | | +--ro vid*               dot1qtypes:vlan-index-type
7         | | +--rw vid-to-fid* [vid]
8         | | +--rw vid                dot1qtypes:vlan-index-type
9         | | +--rw fid?               uint32
10        +--rw bridge-mst
11        +--rw mstid*                 dot1qtypes:mstid-type
12        +--rw fid-to-mstid* [fid]
13        | +--rw fid                uint32
14        | +--rw mstid?              dot1qtypes:mstid-type
15        +--rw fid-to-mstid-allocation* [fids]
16        +--rw fids                  dot1qtypes:vid-range-type
17        +--rw mstid?                dot1qtypes:mstid-type
18    augment /if:interfaces/if:interface:
19    +--rw bridge-port
20    +--rw component-name?            string
21    +--rw port-type?                 identityref
22    +--rw pvid?                       dot1qtypes:vlan-index-type
23    +--rw default-priority?           dot1qtypes:priority-type
24    +--rw priority-regeneration
25    | +--rw priority0?                priority-type
26    | +--rw priority1?                priority-type
27    | +--rw priority2?                priority-type
28    | +--rw priority3?                priority-type
29    | +--rw priority4?                priority-type
30    | +--rw priority5?                priority-type
31    | +--rw priority6?                priority-type
32    | +--rw priority7?                priority-type
33    +--rw pcp-selection?              dot1qtypes:pcp-selection-type
34    +--rw pcp-decoding-table
35    | +--rw pcp-decoding-map* [pcp]
36    | | +--rw pcp                    pcp-selection-type
37    | | +--rw priority-map* [priority-code-point]
38    | | +--rw priority-code-point    priority-type
39    | | +--rw priority?               priority-type
40    | | +--rw drop-eligible?          boolean
41    +--rw pcp-encoding-table
42    | +--rw pcp-encoding-map* [pcp]
43    | | +--rw pcp                    pcp-selection-type
44    | | +--rw priority-map* [priority dei]
45    | | +--rw priority                priority-type
46    | | +--rw dei                     boolean
47    | | +--rw priority-code-point?    priority-type
48    +--rw use-dei?                    boolean
49    +--rw drop-encoding?                boolean
50    +--rw service-access-priority-selection? boolean
51    +--rw service-access-priority
52    | +--rw priority0?                priority-type
53    | +--rw priority1?                priority-type
54    | +--rw priority2?                priority-type
55    | +--rw priority3?                priority-type
56    | +--rw priority4?                priority-type
57    | +--rw priority5?                priority-type
58    | +--rw priority6?                priority-type
59    | +--rw priority7?                priority-type
60    +--rw traffic-class
61    | +--rw traffic-class-map* [priority]
62    | | +--rw priority                priority-type
63    | | +--rw available-traffic-class* [num-traffic-class]
64    | | +--rw num-traffic-class        uint8
65    | | +--rw traffic-class?           traffic-class-type
66    +--rw acceptable-frame?            enumeration
67    +--rw enable-ingress-filtering?     boolean
68    +--rw enable-restricted-vlan-registration? boolean
69    +--rw enable-vid-translation-table? boolean
70    +--rw enable-egress-vid-translation-table? boolean
71    +--rw protocol-group-vid-set* [group-id] {port-and-protocol-based-vlan}?
72    | +--rw group-id                uint32
```

```
1      | +--rw vid*          dot1qtypes:vlanid
2      +--rw admin-point-to-point?      enumeration
3      +--ro protocol-based-vlan-classification?  boolean {port-and-protocol-
4      based-vlan}?
5      +--ro max-vid-set-entries?        uint16 {port-and-protocol-
6      based-vlan}?
7      +--ro port-number?                dot1qtypes:port-number-type
8      +--ro address?                    ieee:mac-address
9      +--ro capabilities?                bits
10     +--ro type-capabilities?           bits
11     +--ro external?                    boolean
12     +--ro oper-point-to-point?         boolean
13     +--ro media-dependent-overhead?    uint8
14     +--ro statistics
15     | +--ro delay-exceeded-discards?    yang:counter64
16     | +--ro mtu-exceeded-discards?      yang:counter64
17     | +--ro frame-rx?                   yang:counter64
18     | +--ro octets-rx?                   yang:counter64
19     | +--ro frame-tx?                   yang:counter64
20     | +--ro octets-tx?                   yang:counter64
21     | +--ro discard-inbound?            yang:counter64
22     | +--ro forward-outbound?           yang:counter64
23     | +--ro discard-lack-of-buffers?     yang:counter64
24     | +--ro discard-transit-delay-exceeded? yang:counter64
25     | +--ro discard-on-error?           yang:counter64
26     | +--ro discard-on-ingress-filtering? yang:counter64 {ingress-
27     filtering}?
28     +--rw vid-translations* [local-vid]
29     | +--rw local-vid      dot1qtypes:vlanid
30     | +--rw relay-vid?    dot1qtypes:vlanid
31     +--rw egress-vid-translations* [relay-vid]
32     +--rw relay-vid      dot1qtypes:vlanid
33     +--rw local-vid?    dot1qtypes:vlanid
```

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

```
27
28
29 module: ieee802-dot1q-tpmr
30   augment /if:interfaces/if:interface/dot1q:bridge-port:
31     +--rw managed-address?      boolean
32     +--rw mac-status-propagation
33     +--rw link-notify?          boolean
34     +--rw link-notify-wait?     yang:timeticks
35     +--rw link-notify-retry?    yang:timeticks
36     +--rw mac-notify?           boolean
37     +--rw mac-notify-time?      yang:timeticks
38     +--rw mac-recover-time?     yang:timeticks
39   augment /if:interfaces/if:interface/dot1q:bridge-port/dot1q:statistics:
40     +--ro acks-tx?               yang:counter64
41     +--ro add-notificatons-tx?   yang:counter64
42     +--ro loss-notification-tx?  yang:counter64
43     +--ro loss-confirmation-tx?  yang:counter64
44     +--ro acks-rx?               yang:counter64
45     +--ro add-notificatons-rx?   yang:counter64
46     +--ro loss-notification-rx?  yang:counter64
47     +--ro loss-confirmation-rx?  yang:counter64
48     +--ro add-events?            yang:counter64
49     +--ro loss-events?           yang:counter64
50     +--ro mac-status-notifications? yang:counter64
```

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

51 N/A

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

```
52
53 module: ieee802-dot1q-pb
54   augment /if:interfaces/if:interface/dot1q:bridge-port:
55     +--rw svid?                  dot1qtypes:vlanid
```

```
1      +---rw cvid-registration* [cvld]
2      | +---rw cvld          dot1qtypes:vlanid
3      | +---rw svid?        dot1qtypes:vlanid
4      | +---rw untagged-pep? boolean
5      | +---rw untagged-cep? boolean
6      +---rw service-priority-regeneration* [svid]
7      | +---rw svid          dot1qtypes:vlanid
8      | +---rw priority-regeneration
9      |   +---rw priority0?  priority-type
10     |   +---rw priority1?  priority-type
11     |   +---rw priority2?  priority-type
12     |   +---rw priority3?  priority-type
13     |   +---rw priority4?  priority-type
14     |   +---rw priority5?  priority-type
15     |   +---rw priority6?  priority-type
16     |   +---rw priority7?  priority-type
17     +---rw rcap-internal-interface* [external-svid]
18     | +---rw external-svid  dot1qtypes:vlanid
19     | +---rw internal-port-number? dot1qtypes:port-number-type
20     | +---rw internal-svid?  dot1qtypes:vlanid
21     | +---rw internal-interface-type? enumeration
```

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

```
19 module: ieee802-dot1q-stream-filters-gates
20   augment /dot1q:bridges/dot1q:bridge/dot1q:component:
21     +---rw stream-filters
22     | +---rw stream-filter-instance-table* [stream-filter-instance-id]
23     | | +---rw stream-filter-instance-id  uint32
24     | | +---rw (stream-handle-spec)?
25     | | | +---:(wildcard)
26     | | | | +---rw wildcard?             empty
27     | | | +---:(stream-handle)
28     | | |   +---rw stream-handle         uint32
29     | | +---rw priority-spec             ipv-type
30     | | +---rw stream-gate-ref           stream-gate-ref
31     | | +---rw filter-specification-list* [index]
32     | |   +---rw index                   uint8
33     | |   +---rw (filter-specification)?
34     | |     +---:(maximum-sdu-size)
35     | |     +---rw maximum-sdu-size      uint32
36     | |     +---rw stream-blocked-due-to-oversize-frame-enabled? boolean
37     | |     +---ro stream-blocked-due-to-oversize-frame?      boolean
38     | +---ro max-stream-filter-instances? uint32
39     +---rw stream-gates
40     | +---rw stream-gate-instance-table* [stream-gate-instance-id]
41     | | +---rw stream-gate-instance-id  uint32
42     | | +---rw gate-enable?             boolean
43     | | +---rw admin-gate-states?       gate-state-value-type
44     | | +---rw admin-ipv?               ipv-type
45     | +---ro max-stream-gate-instances? uint32
```

#### 48.6.8 Data scheme definition for the ieee802-dot1q-psfp YANG module

#### 48.6.9 Data scheme definition for the ieee802-dot1q-ats YANG module

```
50 module: ieee802-dot1q-ats
51   augment /dot1q:bridges/dot1q:bridge/dot1q:component
52     /sfsg:stream-filters/sfsg:stream-filter-instance-table
53     /sfsg:filter-specification-list/sfsg:filter-specification:
54     +---:(scheduler-ref)
```

```
1      +---rw scheduler-ref      ats:scheduler-ref-type
2  augment /if:interfaces/if:interface/dot1q:bridge-port:
3      +---rw ats-port-parameters
4      +---ro discarded-frames-count?  yang:counter64
5  augment /dot1q:bridges/dot1q:bridge/dot1q:component:
6      +---rw schedulers
7      | +---rw scheduler-instance-table* [scheduler-instance-id]
8      | | +---rw scheduler-instance-id      uint32
9      | | +---rw committed-information-rate  uint64
10     | | +---rw committed-burst-size       uint32
11     | | +---rw scheduler-group-ref        ats:scheduler-group-ref-type
12     | +---ro max-scheduler-instances?    uint32
13     +---rw scheduler-groups
14     | +---rw scheduler-group-instance-table* [scheduler-group-instance-id]
15     | +---rw scheduler-group-instance-id  uint32
16     | +---rw max-residence-time          uint32
17     +---ro max-scheduler-group-instances? uint32
18     +---rw scheduler-timing-characteristics
19     | +---ro scheduler-timing-characteristics-table* [reception-port
20     | transmission-port]
21     | +---rw reception-port              dot1qtypes:port-number-type
22     | +---ro transmission-port           dot1qtypes:port-number-type
23     | +---ro clock-offset-variation-max  uint32
24     | +---ro clock-rate-deviation-max    uint32
25     | +---ro arrival-recognition-delay-max uint32
26     | +---ro processing-delay-min        uint32
27     | +---ro processing-delay-max        uint32
```

#### 48.6.10 Data scheme definition for the ieee802-dot1q-sched YANG module

N/A

#### 48.6.11 Data scheme definition for the ieee802-dot1q-preemption YANG module

N/A

#### 48.6.12 Data scheme definition for the ieee802-dot1q-cfm-types YANG module

N/A

#### 48.6.13 Data scheme definition for the ieee802-dot1q-cfm YANG module

N/A

#### 48.6.14 Data scheme definition for the ieee802-dot1q-cfm-bridge YANG module

N/A

#### 48.6.15 Data scheme definition for the ieee802-dot1q-cfm-alarms YANG module

N/A

### 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

```
1 module ieee802-types {
2   namespace urn:ieee:std:802.1Q:yang:ieee802-types;
3   prefix ieee;
4   organization
5     "IEEE 802.1 Working Group";
6   contact
7     "WG-URL: http://www.ieee802.org/1/
8     WG-EMail: stds-802-1-L@ieee.org
9
10    Contact: IEEE 802.1 Working Group Chair
11    Postal: C/O IEEE 802.1 Working Group
12           IEEE Standards Association
13           445 Hoes Lane
14           P.O. Box 1331
15           Piscataway
16           NJ 08854
17           USA
18
19    E-mail: STDS-802-1-L@IEEE.ORG";
20  description
21    "This module contains a collection of generally useful derived
22    data types for IEEE YANG models.";
23  revision 2018-03-07 {
24    description
25      "Published as part of IEEE Std 802.1Q-2018.
26      Initial version.";
27    reference
28      "IEEE Std 802.1Q-2018, Bridges and Bridged Networks.";
29  }
30  typedef mac-address {
31    type string {
32      pattern "[0-9a-fA-F]{2}(-[0-9a-fA-F]{2}){5}";
33    }
34    description
35      "The mac-address type represents a MAC address in the canonical
36      format and hexadecimal format specified by IEEE Std 802. The
37      hexadecimal representation uses uppercase characters.";
38    reference
39      "3.1 of IEEE Std 802-2014
40      8.1 of IEEE Std 802-2014";
41  }
42 }
```

### 48.7.2 Definitions for the ieee802-dot1q-types YANG module

```
1 module ieee802-dot1q-types {
2   namespace urn:ieee:std:802.1Q:yang:ieee802-dot1q-types;
3   prefix dot1q-types;
4   import ietf-yang-types {
5     prefix yang;
6   }
7   organization
8     "IEEE 802.1 Working Group";
9   contact
10     "WG-URL: http://www.ieee802.org/1/
11     WG-EMail: stds-802-1-L@ieee.org
12
13    Contact: IEEE 802.1 Working Group Chair
14    Postal: C/O IEEE 802.1 Working Group
15           IEEE Standards Association
16           445 Hoes Lane
17           P.O. Box 1331
18           Piscataway
19           NJ 08854
20           USA
21
22    E-mail: STDS-802-1-L@IEEE.ORG";
23 }
```



```
1      description
2      "Common types used within dot1q-bridge modules.";
3      revision 2018-03-07 {
4          description
5              "Published as part of IEEE Std 802.1Q-2018.
6              Initial version.";
7          reference
8              "IEEE Std 802.1Q-2018, Bridges and Bridged Networks.";
9      }
10     identity dot1q-vlan-type {
11         description
12             "Base identity from which all 802.1Q VLAN tag types are derived
13             from.";
14     }
15     identity c-vlan {
16         base dot1q-vlan-type;
17         description
18             "An 802.1Q Customer VLAN, using the 81-00 EtherType";
19         reference
20             "5.5 of IEEE Std 802.1Q-2018";
21     }
22     identity s-vlan {
23         base dot1q-vlan-type;
24         description
25             "An 802.1Q Service VLAN, using the 88-A8 EtherType originally
26             introduced in 802.1ad, and incorporated into 802.1Q (2011)";
27         reference
28             "5.6 of IEEE Std 802.1Q-2018";
29     }
30     typedef name-type {
31         type string {
32             length "0..32";
33         }
34         description
35             "A text string of up to 32 characters, of locally determined
36             significance.";
37     }
38     typedef port-number-type {
39         type uint32 {
40             range "1..65535";
41         }
42         description
43             "The port number of the Bridge port for which this entry
44             contains Bridge management information.";
45     }
46     typedef priority-type {
47         type uint8 {
48             range "0..7";
49         }
50         description
51             "A range of priorities from 0 to 7 (inclusive). The Priority
52             Code Point (PCP) is a 3-bit field that refers to the class of
53             service associated with an 802.1Q VLAN tagged frame. The field
54             specifies a priority value between 0 and 7, these values can be
55             used by quality of service (QoS) to prioritize different classes
56             of traffic.";
57     }
58     typedef vid-range-type {
59         type string {
60             pattern
61                 "([1-9]" +
62                 "[0-9]{0,3}" +
63                 "(-[1-9][0-9]{0,3})?" +
64                 "(,[1-9][0-9]{0,3}(-[1-9][0-9]{0,3})?)*";
65         }
66         description
67             "A list of VLAN Ids, or non overlapping VLAN ranges, in
68             ascending order, between 1 and 4094.
69
70             This type is used to match an ordered list of VLAN Ids, or
71             contiguous ranges of VLAN Ids. Valid VLAN Ids must be in the
```

```
1         range 1 to 4094, and included in the list in non overlapping
2         ascending order.
3         For example: 1,10-100,250,500-1000";
4     }
5     typedef vlanid {
6         type uint16 {
7             range "1..4094";
8         }
9         description
10        "The vlanid type uniquely identifies a VLAN. This is the 12-bit
11        VLAN-ID used in the VLAN Tag header. The range is defined by the
12        referenced specification. This type is in the value set and its
13        semantics equivalent to the VlanId textual convention of the
14        SMiv2.";
15    }
16    typedef vlan-index-type {
17        type uint32 {
18            range "1..4094 | 4096..4294967295";
19        }
20        description
21        "A value used to index per-VLAN tables. Values of 0 and 4095 are
22        not permitted. The range of valid VLAN indices. If the value is
23        greater than 4095, then it represents a VLAN with scope local to
24        the particular agent, i.e., one without a global VLAN-ID
25        assigned to it. Such VLANs are outside the scope of IEEE 802.1Q,
26        but it is convenient to be able to manage them in the same way
27        using this YANG module.";
28        reference
29        "9.6 of IEEE Std 802.1Q-2018";
30    }
31    typedef mstid-type {
32        type uint32 {
33            range "1..4094";
34        }
35        description
36        "In an MSTP Bridge, an MSTID, i.e., a value used to identify a
37        spanning tree (or MST) instance";
38        reference
39        "13.8 of IEEE Std 802.1Q-2018";
40    }
41    typedef pcp-selection-type {
42        type enumeration {
43            enum 8POD {
44                description
45                "8 priorities, 0 drop eligible";
46            }
47            enum 7P1D {
48                description
49                "7 priorities, 1 drop eligible";
50            }
51            enum 6P2D {
52                description
53                "6 priorities, 2 drop eligible";
54            }
55            enum 5P3D {
56                description
57                "5 priorities, 3 drop eligible";
58            }
59        }
60        description
61        "Priority Code Point selection types.";
62        reference
63        "12.6.2.5.3 of IEEE Std 802.1Q-2018
64        6.9.3 of IEEE Std 802.1Q-2018";
65    }
66    typedef protocol-frame-format-type {
67        type enumeration {
68            enum Ethernet {
69                description
70                "Ethernet frame format";
71            }
72        }
73    }
```

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

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

```
1         incoming priority values on a per-Port basis, under management
2         control.";
3     reference
4     "6.9.4 of IEEE Std 802.1Q-2018";
5     leaf priority0 {
6         type priority-type;
7         default "0";
8         description
9         "Priority 0";
10        reference
11        "12.6.2.3 of IEEE Std 802.1Q-2018
12        6.9.4 of IEEE Std 802.1Q-2018";
13    }
14    leaf priority1 {
15        type priority-type;
16        default "1";
17        description
18        "Priority 1";
19        reference
20        "12.6.2.3 of IEEE Std 802.1Q-2018
21        6.9.4 of IEEE Std 802.1Q-2018";
22    }
23    leaf priority2 {
24        type priority-type;
25        default "2";
26        description
27        "Priority 2";
28        reference
29        "12.6.2.3 of IEEE Std 802.1Q-2018
30        6.9.4 of IEEE Std 802.1Q-2018";
31    }
32    leaf priority3 {
33        type priority-type;
34        default "3";
35        description
36        "Priority 3";
37        reference
38        "12.6.2.3 of IEEE Std 802.1Q-2018
39        6.9.4 of IEEE Std 802.1Q-2018";
40    }
41    leaf priority4 {
42        type priority-type;
43        default "4";
44        description
45        "Priority 4";
46        reference
47        "12.6.2.3 of IEEE Std 802.1Q-2018
48        6.9.4 of IEEE Std 802.1Q-2018";
49    }
50    leaf priority5 {
51        type priority-type;
52        default "5";
53        description
54        "Priority 5";
55        reference
56        "12.6.2.3 of IEEE Std 802.1Q-2018
57        6.9.4 of IEEE Std 802.1Q-2018";
58    }
59    leaf priority6 {
60        type priority-type;
61        default "6";
62        description
63        "Priority 6";
64        reference
65        "12.6.2.3 of IEEE Std 802.1Q-2018
66        6.9.4 of IEEE Std 802.1Q-2018";
67    }
68    leaf priority7 {
69        type priority-type;
70        default "7";
71        description
72        "Priority 7";
```

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

```
1      description
2      "The priority code point selection type.";
3      reference
4      "12.6.2.7 of IEEE Std 802.1Q-2018
5      6.9.3 of IEEE Std 802.1Q-2018";
6  }
7  list priority-map {
8      key "priority dei";
9      description
10     "This map associated the priority and drop-eligible
11     parameters to the priority code point field of the VLAN tag.";
12     leaf priority {
13         type priority-type;
14         description
15         "Priority associated with the pcp.";
16         reference
17         "12.6.2.7 of IEEE Std 802.1Q-2018
18         6.9.3 of IEEE Std 802.1Q-2018";
19     }
20     leaf dei {
21         type boolean;
22         description
23         "The drop eligible value.";
24         reference
25         "12.6.2 of IEEE Std 802.1Q-2018
26         8.6.6 of IEEE Std 802.1Q-2018";
27     }
28     leaf priority-code-point {
29         type priority-type;
30         description
31         "PCP value for priority when DEI value";
32         reference
33         "12.6.2.9 of IEEE Std 802.1Q-2018
34         6.9.3 of IEEE Std 802.1Q-2018";
35     }
36 }
37 }
38 }
39 grouping service-access-priority-table-grouping {
40     description
41     "The Service Access Priority Table associates a received
42     priority with a service access priority.";
43     reference
44     "12.6.2.17 of IEEE Std 802.1Q-2018
45     6.13.1 of IEEE Std 802.1Q-2018";
46     leaf priority0 {
47         type priority-type;
48         default "0";
49         description
50         "Service access priority value for priority 0";
51         reference
52         "12.6.2.17 of IEEE Std 802.1Q-2018
53         6.13.1 of IEEE Std 802.1Q-2018";
54     }
55     leaf priority1 {
56         type priority-type;
57         default "1";
58         description
59         "Service access priority value for priority 1";
60         reference
61         "12.6.2.17 of IEEE Std 802.1Q-2018
62         6.13.1 of IEEE Std 802.1Q-2018";
63     }
64     leaf priority2 {
65         type priority-type;
66         default "2";
67         description
68         "Service access priority value for priority 2";
69         reference
70         "12.6.2.17 of IEEE Std 802.1Q-2018
71         6.13.1 of IEEE Std 802.1Q-2018";
72     }
73 }
```

```
1     leaf priority3 {
2         type priority-type;
3         default "3";
4         description
5             "Service access priority value for priority 3";
6         reference
7             "12.6.2.17 of IEEE Std 802.1Q-2018
8             6.13.1 of IEEE Std 802.1Q-2018";
9     }
10    leaf priority4 {
11        type priority-type;
12        default "4";
13        description
14            "Service access priority value for priority 4";
15        reference
16            "12.6.2.17 of IEEE Std 802.1Q-2018
17            6.13.1 of IEEE Std 802.1Q-2018";
18    }
19    leaf priority5 {
20        type priority-type;
21        default "5";
22        description
23            "Service access priority value for priority 5";
24        reference
25            "12.6.2.17 of IEEE Std 802.1Q-2018
26            6.13.1 of IEEE Std 802.1Q-2018";
27    }
28    leaf priority6 {
29        type priority-type;
30        default "6";
31        description
32            "Service access priority value for priority 6";
33        reference
34            "12.6.2.17 of IEEE Std 802.1Q-2018
35            6.13.1 of IEEE Std 802.1Q-2018";
36    }
37    leaf priority7 {
38        type priority-type;
39        default "7";
40        description
41            "Service access priority value for priority 7";
42        reference
43            "12.6.2.17 of IEEE Std 802.1Q-2018
44            6.13.1 of IEEE Std 802.1Q-2018";
45    }
46    }
47    grouping traffic-class-table-grouping {
48        description
49            "The Traffic Class Table models the operations that can be
50            performed on, or inquire about, the current contents of the
51            Traffic Class Table (8.6.6) for a given Port.";
52        reference
53            "12.6.3 of IEEE Std 802.1Q-2018
54            8.6.6 of IEEE Std 802.1Q-2018";
55        list traffic-class-map {
56            key "priority";
57            description
58                "The priority index into the traffic class table.";
59            leaf priority {
60                type priority-type;
61                description
62                    "The priority of the traffic class entry.";
63                reference
64                    "8.6.6 of IEEE Std 802.1Q-2018";
65            }
66            list available-traffic-class {
67                key "num-traffic-class";
68                description
69                    "The traffic class index associated with a given priority
70                    within the traffic class table.";
71                reference
72                    "8.6.6 of IEEE Std 802.1Q-2018";
73            }
74        }
75    }
```



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

```
1         }
2     leaf connection-identifier {
3         type port-number-type;
4         description
5             "A Port MAP may contain a connection identifier (8.8.12)
6             for each outbound port. The connection identifier may be
7             associated with the Bridge Port value maintained in a
8             Dynamic Filtering Entry of the FDB for Bridge Ports.";
9         reference
10            "8.8.1 of IEEE Std 802.1Q-2018
11            8.8.12 of IEEE Std 802.1Q-2018";
12    }
13 }
14 container static-vlan-registration-entries {
15     description
16         "Static VLAN registration entries.";
17     leaf registrar-admin-control {
18         type enumeration {
19             enum fixed-new-ignored {
20                 description
21                     "Registration Fixed (New ignored).";
22             }
23             enum fixed-new-propagated {
24                 description
25                     "Registration Fixed (New propagated).";
26             }
27             enum forbidden {
28                 description
29                     "Registration Forbidden.";
30             }
31             enum normal {
32                 description
33                     "Normal Registration.";
34             }
35         }
36     }
37     description
38         "The Registrar Administrative Control values for MVRP
39         and MIRP for the VID.";
40     reference
41         "8.8.2 of IEEE Std 802.1Q-2018";
42 }
43 leaf vlan-transmitted {
44     type enumeration {
45         enum tagged {
46             description
47                 "VLAN-tagged";
48         }
49         enum untagged {
50             description
51                 "VLAN-untagged";
52         }
53     }
54     description
55         "Whether frames are to be VLAN-tagged or untagged when
56         transmitted.";
57     reference
58         "8.8.2 of IEEE Std 802.1Q-2018";
59 }
60 }
61 container mac-address-registration-entries {
62     description
63         "MAC address registration entries attributes.";
64     leaf control-element {
65         type enumeration {
66             enum registered {
67                 description
68                     "Forwarded, independently of any dynamic filtering
69                     information held by the FDB.";
70             }
71             enum not-registered {
72                 description
73                     "Filtered, independently of any dynamic filtering
```

```
1         information.";
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         "8.8.4 of IEEE Std 802.1Q-2018";
10 }
11 container dynamic-vlan-registration-entries {
12     description
13         "Dynamic VLAN registration entries attributes.";
14     leaf control-element {
15         type enumeration {
16             enum registered {
17                 description
18                     "Forwarded, independently of any dynamic filtering
19                     information held by the FDB.";
20             }
21         }
22     }
23     description
24         "containing a control element for each outbound Port,
25         specifying that a frame with a destination MAC address,
26         and in the case of VLAN Bridge components, VID that
27         meets this specification.";
28     reference
29         "8.8.5 of IEEE Std 802.1Q-2018";
30 }
31 container dynamic-reservation-entries {
32     description
33         "Dynamic reservation entries attributes.";
34     leaf control-element {
35         type enumeration {
36             enum forward {
37                 description
38                     "Forwarded, independently of any dynamic filtering
39                     information held by the FDB.";
40             }
41             enum filter {
42                 description
43                     "Filtered, independently of any dynamic filtering
44                     information.";
45             }
46         }
47     }
48     description
49         "Containing a control element for each outbound Port,
50         specifying that a frame with a destination MAC address,
51         and in the case of VLAN Bridge components, VID that
52         meets this specification.";
53     reference
54         "8.8.7 of IEEE Std 802.1Q-2018";
55 }
56 container dynamic-filtering-entries {
57     description
58         "Dynamic filtering entries attributes.";
59     leaf control-element {
60         type enumeration {
61             enum forward {
62                 description
63                     "Forwarded, independently of any dynamic filtering
64                     information held by the FDB.";
65             }
66         }
67     }
68     description
69         "Containing a control element for each outbound Port,
70         specifying that a frame with a destination MAC address,
71         and in the case of VLAN Bridge components, VID that
```

```
1         meets this specification.";
2         reference
3             "8.8.3 of IEEE Std 802.1Q-2018";
4     }
5 }
6 }
7 grouping bridge-port-statistics-grouping {
8     description
9         "Grouping of bridge port statistics.";
10    reference
11        "12.6.1.1.3 of IEEE Std 802.1Q-2018";
12    leaf delay-exceeded-discards {
13        type yang:counter64;
14        description
15            "The number of frames discarded by this port due to excessive
16            transit delay through the Bridge. It is incremented by both
17            transparent and source route Bridges.";
18        reference
19            "12.6.1.1.3 of IEEE Std 802.1Q-2018
20            8.6.6 of IEEE Std 802.1Q-2018";
21    }
22    leaf mtu-exceeded-discards {
23        type yang:counter64;
24        description
25            "The number of frames discarded by this port due to an
26            excessive size. It is incremented by both transparent and
27            source route Bridges.";
28        reference
29            "12.6.1.1.3, item g) of IEEE Std 802.1Q-2018";
30    }
31    leaf frame-rx {
32        type yang:counter64;
33        description
34            "The number of frames that have been received by this port
35            from its segment. Note that a frame received on the interface
36            corresponding to this port is only counted by this object if
37            and only if it is for a protocol being processed by the local
38            bridging function, including Bridge management frames.";
39        reference
40            "12.6.1.1.3 of IEEE Std 802.1Q-2018";
41    }
42    leaf octets-rx {
43        type yang:counter64;
44        description
45            "The total number of octets in all valid frames received
46            (including BPDUs, frames addressed to the Bridge as an end
47            station, and frames that were submitted to the Forwarding
48            Process).";
49        reference
50            "12.6.1.1.3 of IEEE Std 802.1Q-2018";
51    }
52    leaf frame-tx {
53        type yang:counter64;
54        description
55            "The number of frames that have been transmitted by this port
56            to its segment. Note that a frame transmitted on the interface
57            corresponding to this port is only counted by this object if
58            and only if it is for a protocol being processed by the local
59            bridging function, including Bridge management frames.";
60    }
61    leaf octets-tx {
62        type yang:counter64;
63        description
64            "The total number of octets that have been transmitted by this
65            port to its segment.";
66    }
67    leaf discard-inbound {
68        type yang:counter64;
69        description
70            "Count of received valid frames that were discarded (i.e.,
```

```
1         filtered) by the Forwarding Process.";
2         reference
3         "12.6.1.1.3 of IEEE Std 802.1Q-2018";
4     }
5     leaf forward-outbound {
6         type yang:counter64;
7         description
8         "The number of frames forwarded to the associated MAC Entity
9         (8.5).";
10        reference
11        "12.6.1.1.3 of IEEE Std 802.1Q-2018";
12    }
13    leaf discard-lack-of-buffers {
14        type yang:counter64;
15        description
16        "The count of frames that were to be transmitted through the
17        associated Port but were discarded due to lack of buffers.";
18        reference
19        "12.6.1.1.3 of IEEE Std 802.1Q-2018";
20    }
21    leaf discard-transit-delay-exceeded {
22        type yang:counter64;
23        description
24        "The number of frames discarded by this port due to excessive
25        transit delay through the Bridge. It is incremented by both
26        transparent and source route Bridges.";
27        reference
28        "12.6.1.1.3 of IEEE Std 802.1Q-2018";
29    }
30    leaf discard-on-error {
31        type yang:counter64;
32        description
33        "The number of frames that were to be forwarded on the
34        associated MAC but could not be transmitted (e.g., frame would
35        be too large, 6.5.8).";
36        reference
37        "12.6.1.1.3 of IEEE Std 802.1Q-2018";
38    }
39 }
40 }
```

### 48.7.3 Definitions for the ieee802-dot1q-bridge YANG module

```
36 module ieee802-dot1q-bridge {
37     namespace urn:ieee:std:802.1Q:yang:ieee802-dot1q-bridge;
38     prefix dot1q;
39     import ieee802-types {
40         prefix ieee;
41     }
42     import ietf-yang-types {
43         prefix yang;
44     }
45     import ietf-interfaces {
46         prefix if;
47     }
48     import iana-if-type {
49         prefix ianaif;
50     }
51     import ieee802-dot1q-types {
52         prefix dot1qtypes;
53     }
54     organization
55     "IEEE 802.1 Working Group";
56     contact
57     "WG-URL: http://www.ieee802.org/1/
58     WG-EMail: stds-802-1-L@ieee.org
59     Contact: IEEE 802.1 Working Group Chair
```

```
1         Postal: C/O IEEE 802.1 Working Group
2             IEEE Standards Association
3             445 Hoes Lane
4             P.O. Box 1331
5             Piscataway
6             NJ 08854
7             USA
8
9         E-mail: STDS-802-1-L@IEEE.ORG";
10        description
11        "This YANG module describes the bridge configuration model for the
12        following IEEE 802.1Q Bridges:
13        1) Two Port MAC Relays
14        2) Customer VLAN Bridges
15        3) Provider Bridges.";
16        revision 2018-03-07 {
17            description
18            "Published as part of IEEE Std 802.1Q-2018.
19            Initial version.";
20            reference
21            "IEEE Std 802.1Q-2018, Bridges and Bridged Networks.";
22        }
23
24        feature ingress-filtering {
25            description
26            "Each Port may support an Enable Ingress Filtering parameter. A
27            frame received on a Port that is not in the member set (8.8.10)
28            associated with the frames VID shall be discarded if this
29            parameter is set. The default value for this parameter is reset,
30            i.e., Disable Ingress Filtering, for all Ports. Any Port that
31            supports setting this parameter shall also support resetting it.
32            The parameter may be configured by the management operations
33            defined in Clause 12.";
34            reference
35            "8.6.2 of IEEE Std 802.1Q-2018";
36        }
37
38        feature extended-filtering-services {
39            description
40            "Extended Filtering Services support the filtering behavior
41            required for regions of a network in which potential recipients
42            of multicast frames exist, and where both the potential
43            recipients of frames and the Bridges are able to support dynamic
44            configuration of filtering information for group MAC addresses.
45            In order to integrate this extended filtering behavior with the
46            needs of regions of the network that support only Basic
47            Filtering Services, Bridges that support Extended Filtering
48            Services can be statically and dynamically configured to modify
49            their filtering behavior on a per-group MAC address basis, and
50            also on the basis of the overall filtering service provided by
51            each outbound Port with regard to multicast frames. The latter
52            capability permits configuration of the Ports default forwarding
53            or filtering behavior with regard to group MAC addresses for
54            which no specific static or dynamic filtering information has
55            been configured.";
56            reference
57            "8.8.4 of IEEE Std 802.1Q-2018
58            Clause 10 of IEEE Std 802.1Q-2018";
59        }
60
61        feature port-and-protocol-based-vlan {
62            description
63            "A VLAN-aware bridge component implementation in conformance to
64            the provisions of this standard for Port-and-Protocol-based VLAN
65            classification (5.4.1) shall 1) Support one or more of the
66            following Protocol Classifications and Protocol Template
67            formats: Ethernet, RFC_1042, SNAP_8021H, SNAP_Other, or
68            LLC_Other (6.12); and may 2) Support configuration of the
69            contents of the Protocol Group Database.";
70            reference
71            "5.4.1.2 of IEEE Std 802.1Q-2018";
72        }
73
74        feature flow-filtering {
75            description
```

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

```
1      description
2      "Represents the type of Bridge port.";
3  }
4  identity c-vlan-bridge-port {
5      base type-of-port;
6      description
7          "Indicates the port can be a C-TAG aware port of an enterprise
8          VLAN aware Bridge.";
9  }
10 identity provider-network-port {
11     base type-of-port;
12     description
13         "Indicates the port can be an S-TAG aware port of a Provider
14         Bridge or Backbone Edge Bridge used for connections within a PBN
15         (Provider Bridged Network) or PBBN (Provider Backbone Bridged
16         Network).";
17 }
18 identity customer-network-port {
19     base type-of-port;
20     description
21         "Indicates the port can be an S-TAG aware port of a Provider
22         Bridge or Backbone Edge Bridge used for connections to the
23         exterior of a PBN (Provider Bridged Network) or PBBN (Provider
24         Backbone Bridged Network).";
25 }
26 identity customer-edge-port {
27     base type-of-port;
28     description
29         "Indicates the port can be a C-TAG aware port of a Provider
30         Bridge used for connections to the exterior of a PBN (Provider
31         Bridged Network) or PBBN (Provider Backbone Bridged Network).";
32 }
33 identity d-bridge-port {
34     base type-of-port;
35     description
36         "Indicates the port can be a VLAN-unaware member of an 802.1Q
37         Bridge.";
38 }
39 identity remote-customer-access-port {
40     base type-of-port;
41     description
42         "Indicates the port can be an S-TAG aware port of a Provider
43         Bridge capable of providing Remote Customer Service Interfaces.";
44 }
45 identity bridge-interface {
46     description
47         "Generic interface property that represents any interface that
48         can be associated with an IEEE 802.1Q compliant Bridge
49         component. Any new Interface types would derive from this
50         identity to automatically pick up Bridge related configuration
51         or operational data.";
52 }
53
54 container bridges {
55     description
56         "Contains the Bridge(s) configuration information.";
57     list bridge {
58         key "name";
59         unique "address";
60         description
61             "Provides configuration data in support of the Bridge
62             Configuration resources. There is a single bridge data node
63             per Bridge.";
64         leaf name {
65             type dot1qtypes:name-type;
66             description
67                 "A text string associated with the Bridge, of locally
68                 determined significance.";
69             reference
70                 "12.4 of IEEE Std 802.1Q-2018";
71         }
72         leaf address {
```



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

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

```
1         description
2           "Static entries per port.";
3         reference
4           "12.4.1.5.2 of IEEE Std 802.1Q-2018";
5       }
6     leaf ivl-capable {
7       type boolean;
8       default "true";
9       description
10        "Independent VLAN Learning (IVL).";
11      reference
12        "12.4.1.5.2 of IEEE Std 802.1Q-2018";
13    }
14  leaf svl-capable {
15    type boolean;
16    default "false";
17    description
18    "Shared VLAN Learning (SVL).";
19    reference
20    "12.4.1.5.2 of IEEE Std 802.1Q-2018";
21  }
22  leaf hybrid-capable {
23    type boolean;
24    default "false";
25    description
26    "Both IVL and SVL simultaneously.";
27    reference
28    "12.4.1.5.2 of IEEE Std 802.1Q-2018";
29  }
30  leaf configurable-pvid-tagging {
31    type boolean;
32    default "false";
33    description
34    "Whether the implementation supports the ability to
35    override the default PVID setting and its egress status
36    (VLAN-tagged or Untagged) on each port.";
37    reference
38    "12.4.1.5.2 of IEEE Std 802.1Q-2018";
39  }
40  leaf local-vlan-capable {
41    type boolean;
42    default "false";
43    description
44    "Can support multiple local Bridges, outside the scope
45    of 802.1Q defined VLANs.";
46    reference
47    "12.4.1.5.2 of IEEE Std 802.1Q-2018";
48  }
49  }
50  container filtering-database {
51    when "../bridge-type != 'two-port-mac-relay-bridge'" {
52      description
53        "Applies to non TPMRs.";
54    }
55    description
56    "Contains filtering information used by the Forwarding
57    Process in deciding through which Ports of the Bridge
58    frames should be forwarded.";
59    reference
60    "12.7 of IEEE Std 802.1Q-2018";
61    leaf aging-time {
62      type uint32 {
63        range "10..10000000";
64      }
65      units "seconds";
66      default "300";
67      description
68      "The timeout period in seconds for aging out
69      dynamically-learned forwarding information.";
70      reference
71      "12.7 of IEEE Std 802.1Q-2018
72      8.8.3 of IEEE Std 802.1Q-2018";
```

```
1      }
2      leaf size {
3          type yang:gauge32;
4          config false;
5          description
6              "The maximum number of entries that can be held in the
7              FDB.";
8          reference
9              "12.7 of IEEE Std 802.1Q-2018";
10     }
11     leaf static-entries {
12         type yang:gauge32;
13         config false;
14         description
15             "The number of Static Filtering entries currently in the
16             FDB.";
17         reference
18             "12.7 of IEEE Std 802.1Q-2018
19             8.8.1 of IEEE Std 802.1Q-2018";
20     }
21     leaf dynamic-entries {
22         type yang:gauge32;
23         config false;
24         description
25             "The number of Dynamic Filtering entries currently in
26             the FDB.";
27         reference
28             "12.7 of IEEE Std 802.1Q-2018
29             8.8.3 of IEEE Std 802.1Q-2018";
30     }
31     leaf static-vlan-registration-entries {
32         type yang:gauge32;
33         config false;
34         description
35             "The number of Static VLAN Registration entries
36             currently in the FDB.";
37         reference
38             "12.7 of IEEE Std 802.1Q-2018
39             8.8.2 of IEEE Std 802.1Q-2018";
40     }
41     leaf dynamic-vlan-registration-entries {
42         type yang:gauge32;
43         config false;
44         description
45             "The number of Dynamic VLAN Registration entries
46             currently in the FDB.";
47         reference
48             "12.7 of IEEE Std 802.1Q-2018
49             8.8.5 of IEEE Std 802.1Q-2018";
50     }
51     leaf mac-address-registration-entries {
52         if-feature "extended-filtering-services";
53         type yang:gauge32;
54         config false;
55         description
56             "The number of MAC Address Registration entries
57             currently in the FDB.";
58         reference
59             "12.7 of IEEE Std 802.1Q-2018
60             8.8.4 of IEEE Std 802.1Q-2018";
61     }
62     list filtering-entry {
63         key "database-id vids address";
64         description
65             "Information for the entries associated with the
66             Permanent Database.";
67         leaf database-id {
68             type uint32;
69             description
70                 "The identity of this Filtering Database.";
71             reference
72                 "12.7.7 of IEEE Std 802.1Q-2018";
```

```
1         }
2     leaf address {
3         type ieee:mac-address;
4         description
5             "A MAC address (unicast, multicast, broadcast) for
6             which the device has forwarding and/or filtering
7             information.";
8         reference
9             "12.7.7 of IEEE Std 802.1Q-2018";
10    }
11    leaf vids {
12        type dot1qtotypes:vid-range-type;
13        description
14            "The set of VLAN identifiers to which this entry
15            applies.";
16        reference
17            "12.7.7 of IEEE Std 802.1Q-2018";
18    }
19    leaf entry-type {
20        type enumeration {
21            enum static {
22                description
23                    "Static entry type";
24            }
25            enum dynamic {
26                description
27                    "Dynamic/learnt entry type";
28            }
29        }
30        description
31            "The type of filtering entry. Whether static or
32            dynamic. Static entries can be created, deleted, and
33            retrieved. However, dynamic entries can only be
34            deleted or retrieved by the management entity.
35            Consequently, a Bridge is not required to accept a
36            command that can alter the dynamic entries except
37            delete a dynamic entry.";
38        reference
39            "12.7.7 of IEEE Std 802.1Q-2018";
40    }
41    uses dot1qtotypes:port-map-grouping;
42    leaf status {
43        type enumeration {
44            enum other {
45                description
46                    "None of the following. This may include the case
47                    where some other object is being used to determine
48                    if and how frames addressed to the value of the
49                    corresponding instance of 'address' are being
50                    forwarded.";
51            }
52            enum invalid {
53                description
54                    "This entry is no longer valid (e.g., it was
55                    learned but has since aged out), but has not yet
56                    been flushed from the table.";
57            }
58            enum learned {
59                description
60                    "The value of the corresponding instance of the
61                    port node was learned and is being used.";
62            }
63            enum self {
64                description
65                    "The value of the corresponding instance of the
66                    address node representing one of the devices
67                    address.";
68            }
69            enum mgmt {
70                description
71                    "The value of the corresponding instance of
72                    address node that is also the value of an existing
```

```
1         instance.";
2     }
3     config false;
4     description
5         "The status of this entry.";
6     }
7 list vlan-registration-entry {
8     key "database-id vids";
9     description
10        "The VLAN Registration Entries models the operations
11        that can be performed on a single VLAN Registration
12        Entry in the FDB. The set of VLAN Registration Entries
13        within the FDB changes under management control and also
14        as a result of MVRP exchanges";
15    reference
16        "12.7.5 of IEEE Std 802.1Q-2018";
17    leaf database-id {
18        type uint32;
19        description
20            "The identity of this Filtering Database.";
21        reference
22            "12.7.7 of IEEE Std 802.1Q-2018";
23    }
24    leaf vids {
25        type dot1qtypes:vid-range-type;
26        description
27            "The set of VLAN identifiers to which this entry
28            applies.";
29        reference
30            "12.7.7 of IEEE Std 802.1Q-2018";
31    }
32    leaf entry-type {
33        type enumeration {
34            enum static {
35                description
36                    "Static entry type";
37            }
38            enum dynamic {
39                description
40                    "Dynamic/learnt entry type";
41            }
42        }
43        description
44            "The type of filtering entry. Whether static or
45            dynamic. Static entries can be created, deleted, and
46            retrieved. However, dynamic entries can only be
47            deleted or retrieved by the management entity.
48            Consequently, a Bridge is not required to accept a
49            command that can alter the dynamic entries except
50            delete a dynamic entry.";
51        reference
52            "12.7.7 of IEEE Std 802.1Q-2018";
53    }
54    uses dot1qtypes:port-map-grouping;
55 }
56 container permanent-database {
57     description
58         "The Permanent Database container models the operations
59         that can be performed on, or affect, the Permanent
60         Database. There is a single Permanent Database per FDB.";
61     leaf size {
62         type yang:gauge32;
63         config false;
64         description
65             "The maximum number of entries that can be held in the
66             FDB.";
67         reference
68             "12.7.6 of IEEE Std 802.1Q-2018";
69     }
70 }
```

```
1         leaf static-entries {
2             type yang:gauge32;
3             config false;
4             description
5                 "The number of Static Filtering entries currently in the
6                 FDB.";
7             reference
8                 "12.7.6 of IEEE Std 802.1Q-2018";
9         }
10        leaf static-vlan-registration-entries {
11            type yang:gauge32;
12            config false;
13            description
14                "The number of Static VLAN Registration entries
15                currently in the FDB.";
16            reference
17                "12.7.6 of IEEE Std 802.1Q-2018";
18        }
19        list filtering-entry {
20            key "database-id vids address";
21            description
22                "Information for the entries associated with the
23                Permanent Database.";
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 address {
32                type ieee:mac-address;
33                description
34                    "A MAC address (unicast, multicast, broadcast) for
35                    which the device has forwarding and/or filtering
36                    information.";
37                reference
38                    "12.7.7 of IEEE Std 802.1Q-2018";
39            }
40            leaf vids {
41                type dot1qt-types:vid-range-type;
42                description
43                    "The set of VLAN identifiers to which this entry
44                    applies.";
45                reference
46                    "12.7.7 of IEEE Std 802.1Q-2018";
47            }
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    uses dot1qtypes:port-map-grouping;
14 }
15 container bridge-vlan {
16     when "../bridge-type != 'two-port-mac-relay-bridge'" {
17         description
18             "Applies to non TPMRs.";
19     }
20     description
21         "The Bridge VLAN container models configuration
22         information that modify, or inquire about, the overall
23         configuration of the Bridges VLAN resources. There is a
24         single Bridge VLAN Configuration managed object per
25         Bridge.";
26     reference
27         "12.10 of IEEE Std 802.1Q-2018";
28     leaf version {
29         type uint16;
30         config false;
31         description
32             "The version number supported.";
33         reference
34             "12.10.1.3 of IEEE Std 802.1Q-2018";
35     }
36     leaf max-vids {
37         type uint16;
38         config false;
39         description
40             "The maximum number of VIDs supported.";
41         reference
42             "12.10.1.3 of IEEE Std 802.1Q-2018";
43     }
44     leaf override-default-pvid {
45         type boolean;
46         default "false";
47         config false;
48         description
49             "Indicates if the default PVID can be overridden, and
50             its egress status (VLAN-tagged or untagged) on each
51             port.";
52         reference
53             "12.10.1.3 of IEEE Std 802.1Q-2018";
54     }
55     leaf protocol-template {
56         if-feature "port-and-protocol-based-vlan";
57         type dot1qtypes:protocol-frame-format-type;
58         config false;
59         description
60             "The data-link encapsulation format or the
61             detagged_frame_type in a Protocol Template";
62         reference
63             "12.10.1.7 of IEEE Std 802.1Q-2018";
64     }
65     leaf max-msti {
66         type uint16;
67         config false;
68         description
69             "The maximum number of MSTIs supported within an MST
70             region (i.e., the number of spanning tree instances that
```



```
1         can be supported in addition to the CIST), for MST
2         Bridges. For SST Bridges, this parameter may be either
3         omitted or reported as 0.";
4         reference
5         "12.10.1.7 of IEEE Std 802.1Q-2018";
6     }
7     list vlan {
8         key "vid";
9         description
10        "List of VLAN related configuration nodes associated
11        with the Bridge.";
12        reference
13        "12.10.2 of IEEE Std 802.1Q-2018";
14        leaf vid {
15            type dot1qtotypes:vlan-index-type;
16            description
17            "The VLAN identifier to which this entry applies.";
18            reference
19            "12.10.2 of IEEE Std 802.1Q-2018";
20        }
21        leaf name {
22            type dot1qtotypes:name-type;
23            description
24            "A text string of up to 32 characters of locally
25            determined significance.";
26            reference
27            "12.10.2 of IEEE Std 802.1Q-2018";
28        }
29        leaf-list untagged-ports {
30            type if:interface-ref;
31            config false;
32            description
33            "The set of ports in the untagged set for this VID.";
34            reference
35            "12.10.2.1.3 of IEEE Std 802.1Q-2018
36            8.8.2 of IEEE Std 802.1Q-2018";
37        }
38        leaf-list egress-ports {
39            type if:interface-ref;
40            config false;
41            description
42            "The set of egress ports in the member set for this
43            VID.";
44            reference
45            "12.10.2.1.3 of IEEE Std 802.1Q-2018
46            8.8.10 of IEEE Std 802.1Q-2018";
47        }
48    }
49    list protocol-group-database {
50        if-feature "port-and-protocol-based-vlan";
51        key "db-index";
52        description
53        "List of the protocol group database entries.";
54        reference
55        "12.10.1.7 of IEEE Std 802.1Q-2018
56        6.12.3 of IEEE Std 802.1Q-2018";
57        leaf db-index {
58            type uint16;
59            description
60            "The protocol group database index.";
61        }
62        leaf frame-format-type {
63            type dot1qtotypes:protocol-frame-format-type;
64            description
65            "The data-link encapsulation format or the
66            detagged_frame_type in a Protocol Template";
67            reference
68            "12.10.1.7 of IEEE Std 802.1Q-2018";
69        }
70        choice frame-format {
71            description
72            "The identification of the protocol above the
```

```
1         data-link layer in a Protocol Template. Depending on
2         the frame type, the octet string will have one of the
3         following values: - For ethernet, rfc1042 and
4         snap8021H, this is the 16-bit (2-octet) IEEE 802
5         Clause 9.3 EtherType field. - For snapOther, this is
6         the 40-bit (5-octet) PID. - For llcOther, this is the
7         2-octet IEEE 802.2 Link Service Access Point (LSAP)
8         pair: first octet for Destination Service Access Point
9         (DSAP) and second octet for Source Service Access
10        Point (SSAP).";
11        reference
12        "12.10.1.7 of IEEE Std 802.1Q-2018";
13        case ethernet-rfc1042-snap8021H {
14            when
15                "frame-format-type = 'Ethernet' or "+
16                "frame-format-type = 'rfc1042' or "+
17                "frame-format-type = 'snap8021H'" {
18                description
19                    "Applies to Ethernet, RFC 1042, SNAP 8021H frame
20                    formats.";
21            }
22            description
23                "Identifier used if Ethenet, RFC1042, or SNAP 8021H.";
24            leaf ethertype {
25                type dot1qtTypes:ethertype-type;
26                description
27                    "Format containing the 16-bit IEEE 802 EtherType
28                    field.";
29                reference
30                    "9.3 of IEEE Std 802-2014";
31            }
32        }
33        case snap-other {
34            when "frame-format-type = 'snapOther'" {
35                description
36                    "Applies to Snap Other frame formats.";
37            }
38            description
39                "Identifier used if SNAP other.";
40            leaf protocol-id {
41                type string {
42                    pattern "[0-9a-fA-F]{2}-[0-9a-fA-F]{2}";
43                }
44                description
45                    "Format containing the 40-bit protocol identifier
46                    (PID). The canonical representation uses uppercase
47                    characters.";
48                reference
49                    "12.10.1.7.1 of IEEE Std 802.1Q-2018";
50            }
51        }
52        case llc-other {
53            when "frame-format-type = 'llcOther'" {
54                description
55                    "Applies to LLC Other frame formats";
56            }
57            description
58                "Identifier used if LLC other.";
59            container dsap-ssap-pairs {
60                description
61                    "A pair of ISO/IEC 8802-2 DSAP and SSAP address
62                    field values, for matching frame formats of
63                    LLC Other.";
64                leaf llc-address {
65                    type string {
66                        pattern "[0-9a-fA-F]{2}-[0-9a-fA-F]{2}";
67                    }
68                    description
69                        "A pair of ISO/IEC 8802-2 DSAP and SSAP address
70                        field values, for matching frame formats of
71                        LLC_Other. The canonical representation uses
72                        uppercase characters.";
```

```
1         reference
2         "12.10.1.7.1 of IEEE Std 802.1Q-2018";
3     }
4 }
5 }
6 leaf group-id {
7     type uint32;
8     description
9     "Designates a group of protocols in the Protocol Group
10    Database.";
11    reference
12    "6.12.2 of IEEE Std 802.1Q-2018";
13 }
14 }
15 list vid-to-fid-allocation {
16     key "vids";
17     description
18     "This list allows inquiries about VID to FID
19     allocations.";
20     leaf vids {
21         type dot1qtTypes:vid-range-type;
22         description
23         "Range of VLAN identifiers.";
24         reference
25         "12.10.3 of IEEE Std 802.1Q-2018";
26     }
27     leaf fid {
28         type uint32;
29         config false;
30         description
31         "The Filtering Database used by a set of VIDs.";
32         reference
33         "12.10.3 of IEEE Std 802.1Q-2018";
34     }
35     leaf allocation-type {
36         type enumeration {
37             enum undefined {
38                 description
39                 "No allocation defined.";
40             }
41             enum fixed {
42                 description
43                 "A fixed allocation to FID is defined.";
44             }
45             enum dynamic {
46                 description
47                 "A dynamic allocation to FID is defined.";
48             }
49         }
50         config false;
51         description
52         "The type of allocation used";
53         reference
54         "12.10.3 of IEEE Std 802.1Q-2018";
55     }
56 }
57 }
58 list fid-to-vid-allocation {
59     key "fid";
60     description
61     "The FID to VID allocations managed object models
62     operations that inquire about FID to VID allocations.";
63     leaf fid {
64         type uint32;
65         description
66         "The Filtering Database used by a set of VIDs.";
67         reference
68         "12.10.3 of IEEE Std 802.1Q-2018";
69     }
70     leaf allocation-type {
71         type enumeration {
72             enum undefined {
```

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

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

```
1         when "../component-name != 'd-bridge-component'" {
2             description
3                 "Applies to non TPMRs";
4         }
5         type dot1qtypes:vlan-index-type;
6         default "1";
7         description
8             "The primary (default) VID assigned to a specific Bridge
9             Port.";
10        reference
11            "12.10.1 of IEEE Std 802.1Q-2018
12            5.4, item m) of IEEE Std 802.1Q-2018";
13    }
14    leaf default-priority {
15        type dot1qtypes:priority-type;
16        default "0";
17        description
18            "The default priority assigned to a specific Bridge Port.";
19        reference
20            "12.6.2 of IEEE Std 802.1Q-2018";
21    }
22    container priority-regeneration {
23        description
24            "The Priority Regeneration Table parameters associated with
25            a specific Bridge Port. A list of Regenerated User
26            Priorities for each received priority on each port of a
27            Bridge. The regenerated priority value may be used to index
28            the Traffic Class Table for each input port. This only has
29            effect on media that support native priority. The default
30            values for Regenerated User Priorities are the same as the
31            User Priorities";
32        reference
33            "12.6.2 of IEEE Std 802.1Q-2018
34            6.9.4 of IEEE Std 802.1Q-2018";
35        uses dot1qtypes:priority-regeneration-table-grouping;
36    }
37    leaf pcp-selection {
38        type dot1qtypes:pcp-selection-type;
39        default "8P0D";
40        description
41            "The Priority Code Point selection assigned to a specific
42            Bridge Port. This object identifies the rows in the PCP
43            encoding and decoding tables that are used to remark frames
44            on this port if this remarking is enabled";
45        reference
46            "12.6.2 of IEEE Std 802.1Q-2018
47            6.9.3 of IEEE Std 802.1Q-2018";
48    }
49    container pcp-decoding-table {
50        description
51            "The Priority Code Point Decoding Table parameters
52            associated with a specific Bridge Port.";
53        uses dot1qtypes:pcp-decoding-table-grouping;
54    }
55    container pcp-encoding-table {
56        description
57            "The Priority Code Point Encoding Table parameters
58            associated with a specific Bridge Port.";
59        uses dot1qtypes:pcp-encoding-table-grouping;
60    }
61    leaf use-dei {
62        type boolean;
63        default "false";
64        description
65            "The Drop Eligible Indicator. If it is set to True, then the
66            drop_eligible parameter is encoded in the DEI of transmitted
67            frames, and the drop_eligible parameter shall be true(1) for
68            a received frame if the DEI is set in the VLAN tag or the
69            Priority Code Point Decoding Table indicates drop_eligible
70            True for the received PCP value. If this parameter is False,
71            the DEI shall be transmitted as zero and ignored on receipt.";
72        reference
```

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

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



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

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

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

```
1         config false;
2         description
3             "The type of feature capabilities supported with port.
4             Indicates the capabilities of this port.";
5         reference
6             "12.4.2 of IEEE Std 802.1Q-2018";
7     }
8     leaf external {
9         type boolean;
10        config false;
11        description
12            "A boolean indicating whether the port is external. A value
13            of True means the port is external. A value of False means
14            the port is internal.";
15        reference
16            "12.4.2 of IEEE Std 802.1Q-2018";
17    }
18    leaf oper-point-to-point {
19        type boolean;
20        config false;
21        description
22            "For a port running spanning tree, this object represents
23            the operational point-to-point status of the LAN segment
24            attached to this port. It indicates whether a port is
25            considered to have a point-to-point connection.
26
27            If admin-point-to-point is set to auto(2), then the value of
28            oper-point-to-point is determined in accordance with the
29            specific procedures defined for the MAC entity concerned, as
30            defined in IEEE Std 802.1AC.
31
32            The value is determined dynamically; that is, it is
33            re-evaluated whenever the value of admin-point-to-point
34            changes, and whenever the specific procedures defined for
35            the MAC entity evaluate a change in its point-to-point
36            status.";
37        reference
38            "IEEE Std 802.1AC
39            12.4.2 of IEEE Std 802.1Q-2018";
40    }
41    container statistics {
42        config false;
43        description
44            "Container of operational state node information associated
45            with the bridge port.";
46        uses dot1qtypes:bridge-port-statistics-grouping;
47        leaf discard-on-ingress-filtering {
48            when "../component-name != 'd-bridge-component'" {
49                description
50                    "Applies to non TPMRs";
51            }
52            if-feature "ingress-filtering";
53            type yang:counter64;
54            description
55                "The number of frames that were discarded as a result of
56                Ingress Filtering being enabled.
57
58                Discontinuities in the value of this counter can occur at
59                re-initialization of the management system, and at other
60                times as indicated by the value of 'discontinuity-time'.";
61            reference
62                "12.6.1.1.3 of IEEE Std 802.1Q-2018";
63        }
64    }
65    list vid-translations {
66        when "../component-name != 'd-bridge-component'" {
67            description
68                "Applies to non TPMRs";
69        }
70        key "local-vid";
71        description
72            "To configure the VID Translation Table (6.9) associated
```

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

#### 48.7.4 Definitions for the ieee802-dot1q-tpmr YANG module

```
1
2
3 module ieee802-dot1q-tpmr {
4     namespace urn:ieee:std:802.1Q:yang:ieee802-dot1q-tpmr;
5     prefix dot1q-tpmr;
6     import ieee802-dot1q-bridge {
7         prefix dot1q;
8     }
9     import ietf-yang-types {
10        prefix yang;
11    }
12    import ietf-interfaces {
13        prefix if;
14    }
15    organization
16        "IEEE 802.1 Working Group";
17    contact
18        "WG-URL: http://www.ieee802.org/1/
19        WG-EMail: stds-802-1-L@ieee.org
20
21        Contact: IEEE 802.1 Working Group Chair
22        Postal: C/O IEEE 802.1 Working Group
23                IEEE Standards Association
24                445 Hoes Lane
25                P.O. Box 1331
26                Piscataway
27                NJ 08854
28                USA
29
30        E-mail: STDS-802-1-L@IEEE.ORG";
31    description
32        "This YANG module describes the bridge configuration model for the
33        Two Port MAC Relays.";
34    revision 2018-03-07 {
35        description
36            "Published as part of IEEE Std 802.1Q-2018.
37            Initial version.";
38        reference
39            "IEEE Std 802.1Q-2018, Bridges and Bridged Networks.";
40    }
41    augment "/if:interfaces/if:interface/dot1q:bridge-port" {
42        when "dot1q:port-type = 'dot1q:d-bridge-port'" {
43            description
44                "Applies to TPMRs ports";
45        }
46        description
47            "Augment Interface model with TPMR port configuration
48            specific nodes.";
49        leaf managed-address {
50            type boolean;
51            default "true";
52            description
53                "A Boolean value, which is TRUE if the MAC address is the
54                management address for the TPMR, and is otherwise FALSE.
55
56                The TPMR management entity may make use of one or both Ports
57                of a TPMR to transmit and receive management frames. However,
58                the MAC address used by the TPMR management entity as the
59                source MAC address in transmitted management frames (the
60                management MAC address) is the individual MAC address
61                associated with one of the Ports of the TPMR";
62            reference
63                "12.19.1.1.1.3 of IEEE Std 802.1Q-2018";
64        }
65        container mac-status-propagation {
66            description
67                "MAC status propagation configuration node parameters.";
68            leaf link-notify {
69                type boolean;
70                default "true";
71            }
72        }
73    }
74 }
```

```
1         description
2             "The current value (Boolean) of LinkNotify (23.5.1) being
3             used by the MSP state machines.";
4         reference
5             "12.19.4.1.1.3 of IEEE Std 802.1Q-2018
6             12.19.4.1.2.2 of IEEE Std 802.1Q-2018";
7     }
8     leaf link-notify-wait {
9         type yang:timeticks {
10            range "20..100";
11        }
12        default "40";
13        description
14            "The current value, in centiseconds, of LinkNotifyWait
15            (23.5.2) being used by the MSP state machines.";
16        reference
17            "12.19.4.1.1.3 of IEEE Std 802.1Q-2018
18            12.19.4.1.2.2 of IEEE Std 802.1Q-2018";
19    }
20    leaf link-notify-retry {
21        type yang:timeticks {
22            range "10..100";
23        }
24        default "100";
25        description
26            "The current value, in centiseconds, of LinkNotifyRetry
27            (23.5.3) being used by the MSP state machines.";
28        reference
29            "12.19.4.1.1.3 of IEEE Std 802.1Q-2018
30            12.19.4.1.2.2 of IEEE Std 802.1Q-2018";
31    }
32    leaf mac-notify {
33        type boolean;
34        default "true";
35        description
36            "The current value (Boolean) of MACNotify (23.5.4) being
37            used by the MSP state machines.";
38        reference
39            "12.19.4.1.1.3 of IEEE Std 802.1Q-2018
40            12.19.4.1.2.2 of IEEE Std 802.1Q-2018";
41    }
42    leaf mac-notify-time {
43        type yang:timeticks {
44            range "1..50";
45        }
46        default "20";
47        description
48            "The current value, in centiseconds, of MACNotifyTime
49            (23.5.5) being used by the MSP state machines.";
50        reference
51            "12.19.4.1.1.3 of IEEE Std 802.1Q-2018
52            12.19.4.1.2.2 of IEEE Std 802.1Q-2018";
53    }
54    leaf mac-recover-time {
55        type yang:timeticks {
56            range "2..50";
57        }
58        default "10";
59        description
60            "The current value, in centiseconds, of MACRecoverTime
61            (23.5.6) being used by the MSP state machines.";
62        reference
63            "12.19.4.1.1.3 of IEEE Std 802.1Q-2018
64            12.19.4.1.2.2 of IEEE Std 802.1Q-2018";
65    }
66    }
67    }
68    augment
69        "/if:interfaces/if:interface/dot1q:bridge-port/dot1q:statistics" {
70        when "../dot1q:port-type = 'dot1q:d-bridge-port'" {
71            description
72                "Applies to TPMRs ports";
```

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



```
1         config false;
2         description
3             "The number of adds received (23.6.11) by the Ports Receive
4             Process.
5
6             Discontinuities in the value of this counter can occur at
7             re-initialization of the management system, and at other times
8             as indicated by the value of 'discontinuity-time.';
9             reference
10            "12.19.4.1.3.3 of IEEE Std 802.1Q-2018";
11        }
12    leaf loss-notification-rx {
13        type yang:counter64;
14        config false;
15        description
16            "The number of losses received (23.6.13) by the Ports Receive
17            Process.
18
19            Discontinuities in the value of this counter can occur at
20            re-initialization of the management system, and at other times
21            as indicated by the value of 'discontinuity-time.';
22            reference
23            "12.19.4.1.3.3 of IEEE Std 802.1Q-2018";
24        }
25    leaf loss-confirmation-rx {
26        type yang:counter64;
27        config false;
28        description
29            "The number of loss confirms received (23.6.14) by the Ports
30            Receive Process.
31
32            Discontinuities in the value of this counter can occur at
33            re-initialization of the management system, and at other times
34            as indicated by the value of 'discontinuity-time.';
35            reference
36            "12.19.4.1.3.3 of IEEE Std 802.1Q-2018";
37        }
38    leaf add-events {
39        type yang:counter64;
40        config false;
41        description
42            "The number of transitions to STM:ADD directly from STM:DOWN
43            or STM:LOSS (23.8).
44
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
58            Discontinuities in the value of this counter can occur at
59            re-initialization of the management system, and at other times
60            as indicated by the value of 'discontinuity-time.';
61            reference
62            "12.19.4.1.3.3 of IEEE Std 802.1Q-2018";
63        }
64    leaf mac-status-notifications {
65        type yang:counter64;
66        config false;
67        description
68            "The number of transitions to SNM:MAC_NOTIFICATION (23.9).
69
70            Discontinuities in the value of this counter can occur at
71            re-initialization of the management system, and at other times
72            as indicated by the value of 'discontinuity-time.';
```

```
1         reference
2           "12.19.4.1.3.3 of IEEE Std 802.1Q-2018";
3       }
4   }
5
```

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

```
6
7
8   module ieee802-dot1q-vlan-bridge {
9       namespace urn:ieee:std:802.1Q:yang:ieee802-dot1q-vlan-bridge;
10      prefix dot1q-vlan-bridge;
11      organization
12        "IEEE 802.1 Working Group";
13      contact
14        "WG-URL: http://www.ieee802.org/1/
15        WG-EMail: stds-802-1-L@ieee.org
16
17        Contact: IEEE 802.1 Working Group Chair
18        Postal: C/O IEEE 802.1 Working Group
19               IEEE Standards Association
20               445 Hoes Lane
21               P.O. Box 1331
22               Piscataway
23               NJ 08854
24               USA
25
26        E-mail: STDS-802-1-L@IEEE.ORG";
27      description
28        "This YANG module describes the bridge configuration model for
29        Customer VLAN Bridges.";
30      revision 2018-03-07 {
31          description
32            "Published as part of IEEE Std 802.1Q-2018.
33            Initial version.";
34          reference
35            "IEEE Std 802.1Q-2018, Bridges and Bridged Networks.";
36      }
37   }
```

#### 48.7.6 Definitions for the ieee802-dot1q-pb YANG module

```
38
39
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
65               USA
66
67        E-mail: STDS-802-1-L@IEEE.ORG";
68      description
```

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

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

```
1         enum port-based-rcsi {
2             description
3                 "Port-based RCSI";
4         }
5         enum c-tagged-rcsi {
6             description
7                 "C-tagged RCSI";
8         }
9         enum pnp {
10            description
11                "Provider Network Port";
12        }
13        enum discard {
14            description
15                "Discard (external S-VID is not associated with an
16                internal port).";
17        }
18    }
19    description
20        "A value indicating the type of internal interface
21        associated with the external S-VID.";
22    reference
23        "12.13.3.2 of IEEE Std 802.1Q-2018";
24    }
25 }
26 }
```

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

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

```
1      description
2          "The bridge component supports gate state closed.";
3      reference
4          "IEEE Std 802.1Qcr";
5  }
6
7  /* Types and groupings */
8  typedef ipv-type {
9      type enumeration {
10         enum zero {
11             value 0;
12             description
13                 "Priority 0";
14         }
15         enum one {
16             value 1;
17             description
18                 "Priority 1";
19         }
20         enum two {
21             value 2;
22             description
23                 "Priority 2";
24         }
25         enum three {
26             value 3;
27             description
28                 "Priority 3";
29         }
30         enum four {
31             value 4;
32             description
33                 "Priority 4";
34         }
35         enum five {
36             value 5;
37             description
38                 "Priority 5";
39         }
40         enum six {
41             value 6;
42             description
43                 "Priority 6";
44         }
45         enum seven {
46             value 7;
47             description
48                 "Priority 7";
49         }
50         enum wildcard {
51             description
52                 "No Priority";
53         }
54     }
```

```
1      description
2      "An IPV can be either of the following:
3          1) The null value. For a frame that passes through the gate, the
4             priority value associated with the frame is used to determine
5             the frame's traffic class, using the Traffic Class Table as
6             specified in 8.6.6.
7          2) An internal priority value. For a frame that passes through the
8             gate, the IPV is used, in place of the priority value
9             associated with the frame, to determine the frame's traffic
10            class, using the Traffic Class Table as specified in 8.6.6.";
11      reference
12      "8.6.5.2 of IEEE Std 802.1Qcr";
13  }
14  typedef gate-state-value-type {
15      type enumeration {
16          enum open {
17              description
18              "Gate open";
19          }
20          enum closed {
21              description
22              "Gate closed";
23          }
24      }
25      description
26      "The StreamGateStatesValue indicates the desired gate state, open or
27      closed, for the stream gate.";
28      reference
29      "12.31.3.2.1 of IEEE Std 802.1Qcr";
30  }
31  typedef stream-gate-ref {
32      type leafref {
33          path
34          '/dot1q:bridges'+
35          '/dot1q:bridge'+
36          '/dot1q:component'+
37          '/sfsg:stream-gates'+
38          '/sfsg:stream-gate-instance-table'+
39          '/sfsg:stream-gate-instance-id';
40      }
41      description
42      "This type is used to refer to a stream gate instance.";
43  }
44  augment "/dot1q:bridges/dot1q:bridge/dot1q:component" {
45      description
46      "Augments the Bridge component with Stream Filters and Stream Gates.";
47      container stream-filters {
48          description
49          "This container encapsulates all nodes related to Stream Filters.";
50          reference
51          "12.31.1 of IEEE Std 802.1Qcr
52          12.31.2 of IEEE Std 802.1Qcr
53          12.31.3 of IEEE Std 802.1Qcr";
54          list stream-filter-instance-table {
55              key "stream-filter-instance-id";
56              description
57              "There is one Stream Filter Instance Table per Bridge component.
58              Each table row contains a set of parameters that defines a single
59              Stream Filter (8.6.5.1), as detailed in Table 12-31. The table
```

```
1      rows form an ordered list of filter instances, the order being
2      determined by the StreamFilterInstance parameter. Tables can be
3      created or removed dynamically in implementations that support
4      dynamic configuration of Bridge components. Rows in the table can
5      be created or removed dynamically in implementations that support
6      dynamic configuration of stream filters. The value of the
7      stream-handle-spec and priority-spec parameters associated with a
8      received frame determine which stream filter is selected by the
9      frame, and therefore what combination of filtering and policing
10     actions is applied to the frame. If the stream-handle-spec and
11     priority-spec parameters associated with a received frame match
12     more than one stream filter, the stream filter that is selected
13     is the one that appears earliest in the ordered list. If a
14     received frame's stream-handle-spec and priority-spec does not
15     match any of the stream filters in the table, the frame is
16     processed as if Stream Filters and Stream Gates would not be
17     supported.";
18     reference
19     "12.31.2 of IEEE Std 802.1Qcr";
20     leaf stream-filter-instance-id {
21         type uint32;
22         mandatory true;
23         description
24         "An integer index value that determines the place of the stream
25         filter in the ordered list of stream filter instances. The
26         values are ordered according to their integer value; smaller
27         values appear earlier in the ordered list.";
28         reference
29         "12.31.2.1 of IEEE Std 802.1Qcr";
30     }
31     choice stream-handle-spec {
32         description
33         "The stream_handle specification data type allows either of the
34         following to be represented:
35         a) A stream_handle value, represented as an integer.
36         b) The wild card value, which matches any frame";
37         reference
38         "12.31.2.2 of IEEE Std 802.1Qcr";
39
40         /* NOTE: The mapping of the wildcard literal is
41          *         other than in the MIB definition, where
42          *         the wildcard value is mapped to -1.
43          */
44         case wildcard {
45             leaf wildcard {
46                 type empty;
47                 description
48                 "The stream handle specification represents a wild card
49                 value.";
50             }
51         }
52         case stream-handle {
53             leaf stream-handle {
54                 type uint32;
55                 mandatory true;
56                 description
57                 "The stream handle specification refers to a stream_handle
58                 value.";
59             }
60         }
61     }
62 }
```



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

```
1           StreamBlockedDueToOversizeFrame function is enabled; a
2           value of FALSE indicates that the
3           StreamBlockedDueToOversizeFrame function is disabled. The
4           default value of StreamBlockedDueToOversizeFrameEnable is
5           FALSE.";
6           reference
7           "8.6.5.1 of IEEE Std 802.1Qcr
8           8.6.5.3.1 of IEEE Std 802.1Qcr";
9       }
10      leaf stream-blocked-due-to-oversize-frame {
11          type boolean;
12          default "false";
13          config false;
14          description
15              "If StreamBlockedDueToOversizeFrameEnable is TRUE, a
16              value of TRUE in StreamBlockedDueToOversizeFrame
17              indicates that all frames are to be dropped (i.e., the
18              behavior is identical as if the maximum SDU size would be
19              set to 0 octets). If StreamBlockedDueToOversizeFrame is
20              FALSE, it has no effect. The default value of
21              StreamBlockedDueToOversizeFrame is FALSE; if any frame is
22              discarded because it exceeds the maximum SDU size for the
23              stream, then StreamBlockedDueToOversizeFrame is set TRUE.";
24          reference
25              "8.6.5.1 of IEEE Std 802.1Qcr
26              8.6.5.3.1 of IEEE Std 802.1Qcr";
27      }
28  }
29  }
30  leaf max-stream-filter-instances {
31      type uint32;
32      config false;
33      description
34          "The maximum number of Stream Filter instances supported by this
35          Bridge component.";
36      reference
37          "12.31.1.1 of IEEE Std 802.1Qcr
38          8.6.5.1 of IEEE Std 802.1Qcr";
39  }
40  container stream-gates {
41      description
42          "This container encapsulates all nodes related to Stream Gates.";
43      list stream-gate-instance-table {
44          key "stream-gate-instance-id";
45          description
46              "There is one Stream Gate Instance Table per Bridge component.
47              Each table row contains a set of parameters that defines a single
48              Stream Gate (8.6.5.1.2), as detailed in Table 12-32. Tables can
49              be created or removed dynamically in implementations that support
50              dynamic configuration of Bridge components. Rows in the table can
51              be created or removed dynamically in implementations that support
52              dynamic configuration of stream gates.";
53          reference
54              "12.31.3 of IEEE Std 802.1Qcr";
55          leaf stream-gate-instance-id {
56              type uint32;
```

```
1         description
2             "An integer table index that allows the stream gate to be
3             referenced from Stream Filter Instance Table entries.";
4         reference
5             "12.31.2.4 of IEEE Std 802.1Qcr
6             8.6.5.1 of IEEE Std 802.1Qcr
7             8.6.5.2 of IEEE Std 802.1Qcr";
8     }
9     leaf gate-enable {
10        type boolean;
11        default "false";
12        description
13            "A Boolean variable that indicates whether the operation of the
14            state machines is enabled (TRUE) or disabled (FALSE). This
15            variable is set by management. The default value of this
16            variable is FALSE.";
17        reference
18            "8.6.9.4.14 of IEEE Std 802.1Q-2018";
19    }
20    leaf admin-gate-states {
21        type gate-state-value-type;
22        default "open";
23        description
24            "The administrative state associated with this gate, as set by
25            the management.";
26        reference
27            "12.31.3.2.1 of IEEE Std 802.1Qcr
28            8.6.10.4 of IEEE Std 802.1Qcr";
29    }
30    }
31    leaf admin-ipv {
32        type ipv-type;
33        default "wildcard";
34        description
35            "The administrative internal priority value specification.";
36        reference
37            "12.31.3.3 of IEEE Std 802.1Qcr
38            8.6.10.6 of IEEE Std 802.1Qcr
39            8.6.5.2 of IEEE Std 802.1Qcr";
40    }
41    }
42    leaf max-stream-gate-instances {
43        type uint32;
44        config false;
45        description
46            "The maximum number of Stream Gate instances supported by this
47            Bridge component.";
48        reference
49            "12.31.1.2 of IEEE Std 802.1Qcr
50            8.6.5.2 of IEEE Std 802.1Qcr";
51    }
52    }
53 }
54 }
```

## 48.7.8 Definitions for the ieee802-dot1q-psfp YANG module

## 48.7.9 Definitions for the ieee802-dot1q-ats YANG module

```
1
2
3
4
5
6 module ieee802-dot1q-ats {
7     yang-version "1.1";
8     namespace urn:ieee:std:802.1Q:yang:ieee802-dot1q-ats;
9     prefix ats;
10    import ietf-yang-types {
11        prefix yang;
12    }
13    import ietf-interfaces {
14        prefix if;
15    }
16    import ieee802-dot1q-types {
17        prefix dot1qtypes;
18    }
19    import ieee802-dot1q-bridge {
20        prefix dot1q;
21    }
22    import ieee802-dot1q-stream-filters-gates {
23        prefix sfsg;
24    }
25    organization
26        "IEEE 802.1 Working Group";
27    contact
28        "WG-URL: https://1.ieee802.org/
29        WG-EMail: stds-802-1@ieee.org
30        Contact: IEEE 802.1 Working Group Chair
31        Postal: C/O IEEE 802.1 Working Group
32                IEEE Standards Association
33                445 Hoes Lane
34                P.O. Box 1331
35                Piscataway
36                NJ 08855-1331
37                USA
38
39        E-mail: STDS-802-1-L@LISTSERV.IEEE.ORG";
40    description
41        "This module provides management of 802.1Q bridge components that
42        support Asynchronous Traffic Shaping (ATS). NOTICE: This YANG module is
43        part of an unapproved IEEE Standards Draft and is subject to change.";
44    revision 2019-04-13 {
45        description
46            "Initial revision from IEEE P802.1Qcr.";
47        reference
48            "IEEE Std 802.1Qcr";
49    }
50    typedef scheduler-ref-type {
51        type leafref {
52            path
53                '/dot1q:bridges'+
54                '/dot1q:bridge'+
55                '/dot1q:component'+
56                '/ats:schedulers'+
57                '/ats:scheduler-instance-table'+
58                '/ats:scheduler-instance-id';
59        }
60    }
```

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

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

```
1         "12.31.1.5 of IEEE Std 802.1Qcr
2         8.6.5.3.3 of IEEE Std 802.1Qcr";
3     }
4 }
5 container scheduler-groups {
6     description
7         "This container comprises all ATS scheduler group related nodes.";
8     list scheduler-group-instance-table {
9         key "scheduler-group-instance-id";
10        description
11            "Each table row in the Scheduler Group Instance Table comprises a
12            set of parameters that defines a single ATS scheduler group
13            instance (8.6.5.3.3).";
14        reference
15            "12.31.6 of IEEE Std 802.1Qcr
16            8.6.5.3.3 of IEEE Std 802.1Qcr";
17        leaf scheduler-group-instance-id {
18            type uint32;
19            description
20                "An integer table index that allows the scheduler group
21                instance to be referenced from Scheduler Instance Table
22                entries.";
23            reference
24                "12.31.6.1 of IEEE Std 802.1Qcr
25                8.6.5.3.3 of IEEE Std 802.1Qcr";
26        }
27        leaf max-residence-time {
28            type uint32;
29            mandatory true;
30            description
31                "The maximum residence time parameter of the scheduler group,
32                in nanoseconds.";
33            reference
34                "8.6.11.2.13 of IEEE Std 802.1Qcr
35                8.6.5.3.3 of IEEE Std 802.1Qcr";
36        }
37    }
38    leaf max-scheduler-group-instances {
39        type uint32;
40        config false;
41        description
42            "The maximum number of scheduler group instances supported by
43            this Bridge component.";
44        reference
45            "12.31.1.6 of IEEE Std 802.1Qcr
46            8.6.5.3.3 of IEEE Std 802.1Qcr";
47    }
48    container scheduler-timing-characteristics {
49        description
50            "This container comprises all ATS scheduler timing
51            characteristics related nodes.";
52        list scheduler-timing-characteristics-table {
53            key "reception-port transmission-port";
54            config false;
55            description
56                "Each row in this table comprises the timing characteristics of
57                a reception Port transmission Port pair, as detailed in Table
58                12-36.";
59            reference
```

```
1         "12.31.8 of IEEE Std 802.1Qcr
2         8.6.11 of IEEE Std 802.1Qcr";
3     leaf reception-port {
4         type dot1qtypes:port-number-type;
5         config false;
6         mandatory true;
7         description
8             "A reference to the associated reception Port.";
9         reference
10            "12.31.8.1 of IEEE Std 802.1Qcr";
11    }
12    leaf transmission-port {
13        type dot1qtypes:port-number-type;
14        config false;
15        mandatory true;
16        description
17            "A reference to the associated transmission Port.";
18        reference
19            "12.31.8.2 of IEEE Std 802.1Qcr";
20    }
21    leaf clock-offset-variation-max {
22        type uint32;
23        config false;
24        mandatory true;
25        description
26            "The maximum clock offset variation associated with the
27            reception Port transmission Port pair, in nanoseconds.";
28        reference
29            "12.31.8.3 of IEEE Std 802.1Qcr";
30    }
31    leaf clock-rate-deviation-max {
32        type uint32;
33        config false;
34        mandatory true;
35        description
36            "The maximum clock rate deviation associated with the
37            reception Port transmission Port pair, in ppm.";
38        reference
39            "12.31.8.4 of IEEE Std 802.1Qcr";
40    }
41    leaf arrival-recognition-delay-max {
42        type uint32;
43        config false;
44        mandatory true;
45        description
46            "The maximum arrival time recognition delay associated with
47            the reception Port transmission Port pair, in nanoseconds.";
48        reference
49            "12.31.8.5 of IEEE Std 802.1Qcr";
50    }
51    leaf processing-delay-min {
52        type uint32;
53        config false;
54        mandatory true;
55        description
56            "The minimum processing delay associated with the reception
57            Port transmission Port pair, in nanoseconds.";
58        reference
59            "12.31.8.6 of IEEE Std 802.1Qcr";
```



```
1         }
2     leaf processing-delay-max {
3         type uint32;
4         config false;
5         mandatory true;
6         description
7             "The maximum processing delay associated with the reception
8             Port transmission Port pair, in nanoseconds.";
9         reference
10            "12.31.8.7 of IEEE Std 802.1Qcr";
11    }
12 }
13 }
14 }
15 }
```

#### 48.7.10 Definitions for the ieee802-dot1q-sched YANG module

N/A

#### 48.7.11 Definitions for the ieee802-dot1q-preemption YANG module

N/A

#### 48.7.12 Definitions for the ieee802-dot1q-cfm-types YANG module

N/A

#### 48.7.13 Definitions for the ieee802-dot1q-cfm YANG module

N/A

#### 48.7.14 Definitions for the ieee802-dot1q-cfm-bridge YANG module

N/A

#### 48.7.15 Definitions for the ieee802-dot1q-cfm-alarms YANG module

N/A

40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54

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

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

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