

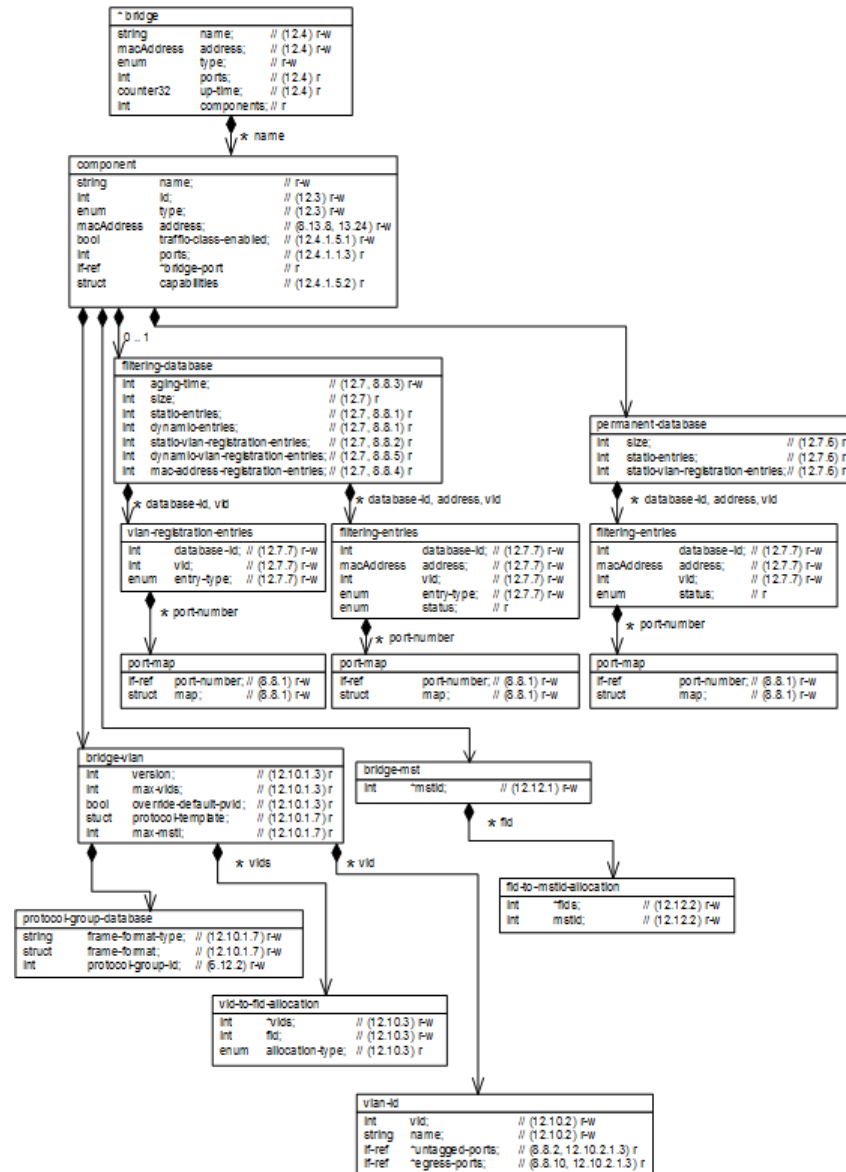
IEEE 802.1Qcx (CFM) YANG Model Overview

Marc Holness (mholness@ciena.com)

Version 1.0

January 23, 2018

IEEE 802.1Q-2017 (Generic) BRIDGE Model



IEEE 802.1Q-2014 BRIDGE Model

- Extended filtering services
- Traffic classes
- Static entry individual port
- IVL capable
- SVL capable
- Hybrid capable
- Configurable PVID tagging
- Local VLAN capable

- I-component
- B-component
- C-VLAN component
- S-VLAN component
- VLAN unaware component
- EVB station edge relay component

- Customer VLAN Bridge
- Provider Bridge
- Provider Edge Bridge
- Backbone Edge Bridge
- Backbone Core Bridge
- Two-Port MAC Relay Bridge

BRIDGE		
macAddress	address	rw
str	name	rw
enum	type	rw
int	number-of-ports	rw
list*	port-list	ro
counter	up-time	ro
ResetBridge()		

COMPONENT		
int32	component-id	ro
enum	component-type	ro
macAddress	address	ro
int	number-of-ports	ro
boolean-array	capabilities	ro
boolean	traffic-class-enabled	rw
boolean	mrrp-enabled-status	rw

PERMANENT DATABASE		
int	size	ro
int	static-entries	ro
int	static-VLAN-registration-entries	ro

FILTERING ENTRY		
int	database-id	rw
macAddress	address	rw
int	vid	rw
map	port-map	rw

FILTERING DATABASE		
int	size	ro
int	static-entries, dynamic-entries	ro
int	static-VLAN-registration-entries, dynamic-VLAN-registration-entries	ro
int	ageing-time	rw
int	mac-address-registration-entries	ro

FILTERING ENTRY		
int	database-id	rw
macAddress	address	rw
int	vid	rw
enum	entry-type	rw
map	port-map	rw

VLAN REGISTRATION ENTRY		
int	database-id	rw
int	vid	rw
enum	entry-type	rw
map	port-map	rw

INTERFACES

BRIDGE VLAN		
int	version	ro
int	max	ro
boolean	override-default-pvid	ro
format	protocol-template	ro
int	max-MSTI	ro

VLAN ID		
int	vid	rw
str	name	rw
list	untagged-ports	ro
list	egress-ports	ro

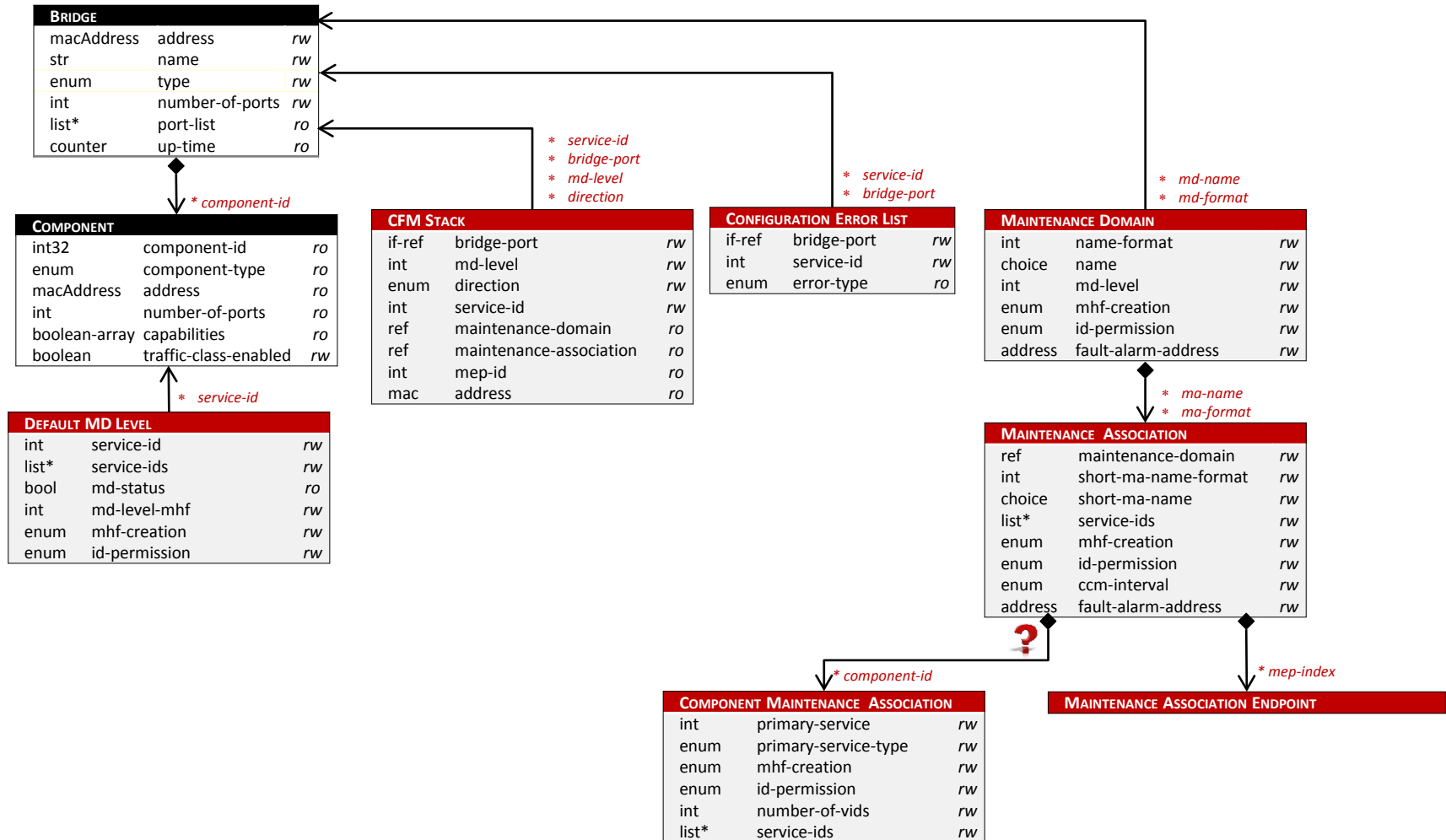
VID ↔ FID ALLOCATION		
int*	vid	rw
int	fid	rw
enum	type	ro

PROTOCOL GROUP DATABASE		
format	frame-format	rw
struct	type	rw
int	protocol-group-id	rw

- Undefined
- Fixed
- Dynamic

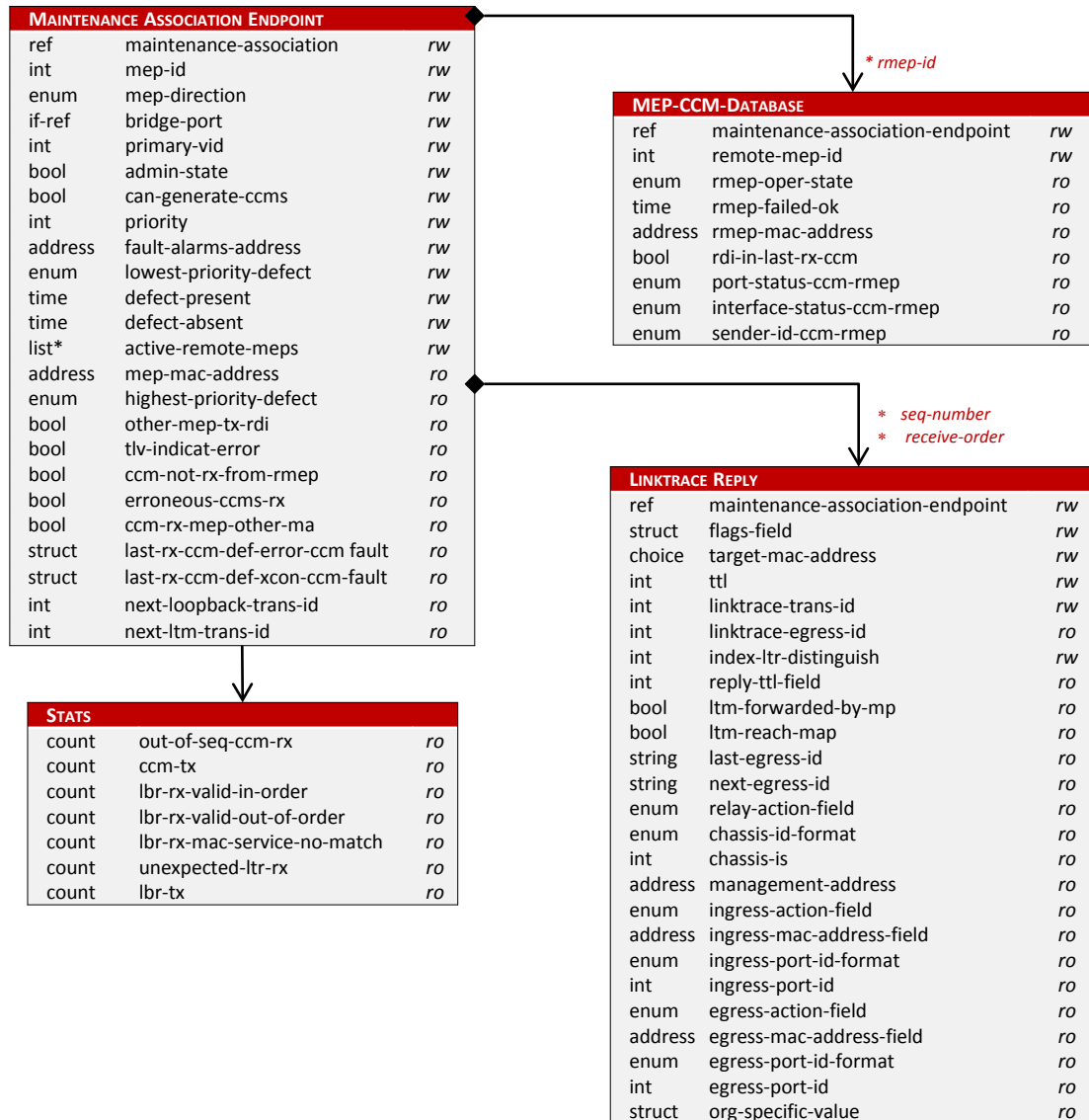
IEEE 802.1Qcx Model

BRIDGE (COMPONENT) TO CFM RELATIONSHIPS



IEEE 802.1Qcx Model

CFM MEP OBJECT RELATIONSHIPS



IEEE 802.1Qcx Model

CFM TRANSMIT COMMANDS AND NOTIFICATIONS

TRANSMIT-LOOPBACK-MESSAGE and
TRANSMIT-LINKTRACE-MESSAGE
commands can be implemented
by RPCs (Remote Procedure Calls)

TRANSMIT-LOOPBACK()

ref	maintenance-association-endpoint	<i>rw</i>
choice	target-address	<i>rw</i>
int	lbm-tx-number	<i>rw</i>
str	data	<i>rw</i>
bool	data-tlv	<i>rw</i>
int	priority	<i>rw</i>
int	dei	<i>rw</i>
bool	lbm-result-ok	<i>ro</i>
int	loopback-trans-id	<i>ro</i>

TRANSMIT-LINKTRACE()

ref	maintenance-association-endpoint	<i>rw</i>
choice	target-address	<i>rw</i>
int	ltm-flags	<i>rw</i>
str	ltm-ttl	<i>rw</i>
bool	ltm-result-ok	<i>ro</i>
int	ltm-seq-number	<i>ro</i>
string	ltm-egress-identifier	<i>ro</i>

Notification (Traps/Alarms) for
MEP FAULT ALARM

MEP FAULT ALARM

ref	maintenance-association-endpoint	<i>ro</i>
enum	mep-priority-defect	<i>ro</i>

Outstanding Areas

1. There are data types (e.g., LinkTrace Reply output) referenced by the CFM YANG model that are defined by IEEE 802.1AB-2005 (LLDP)
 - For example, chassis ID subtype, chassis ID, port ID subtype, and port ID
 - These data types should be defined as part of P802.1Qcu project, and thus introduces a dependency on this project

2. There are data types (e.g., LinkTrace Reply output) referenced by the CFM YANG model that defines a “management address information of the Bridge” transmitting the LTR
 - 802.1Q-2017, Clause 21.5.3.7, provides a description of the Management Address, and references RFC 3419 (Textual Conventions for Transport Addresses) and RFC 4789 (SNMP over IEEE 802 Networks)
 - Transport domain definitions are provided for SNMP/MIBs
 - Transport address definitions are provided for SNMP/MIBs
 - Are similar definitions already provided for YANG?

Outstanding Areas

3. What is the best way to associate YANG CFM objects to existing (802.1Qcp) Bridge YANG object?
 - There is a single CFM Stack object per Bridge
 - There is a single Configuration Error List object per Bridge
 - There is a single Default MD Level object per Bridge Component
 - There is a single MD List object per Bridge

Proposal: Major CFM YANG objects (e.g., CFM Stack, Default MD Level, Configuration Error List, MD List) will define a pointer/reference to the associated Bridge or Bridge Component

- I believe this approach is advantageous over augmenting the Bridge (or Bridge Component), since it could gracefully facilitate other switching/routing devices to utilize the CFM YANG model
- I believe this approach is advantageous over including the pointer in the Bridge or Bridge Component YANG objects, because there is no need to (continuously) amend the Bridge and Bridge Component objects with pointers of users of the Bridge

Outstanding Areas

4. In the CFM-v2 MIB (802.1Q-2017), a “CFM Maintenance Association (MA) Component Table” is defined
 - As stated in 802.1Q-2017, Clause 17.7.7

“... That part of the MA table that can vary from Bridge component to Bridge component is contained in the [CFM MA Component Table] ...”

“... This is the part of the complete MA table that is variable across the Bridges in the MD, or across the components of a single Bridge ...”

- What is the best way to model this in YANG?

Outstanding Areas

5. In the CFM-v2 MIB (802.1Q-2017), a “CFM VLAN Table” is defined
 - As stated in 802.1Q-2017, Clause 17.7.7

“... This table defines the association of VIDs into VLANs. There is an entry in this table, for each component of the Bridge, for each VID that is:

- a) a VID belonging to a VLAN associated with more than one VID; and*
- b) not the Primary VID of that VLAN.*

The entry in this table contains the Primary VID of the VLAN.

*... VLANs that are associated with only one VID **should not** have an entry in this table ...”*

- **What?**
- How should this be effectively modeled in YANG?

Preliminary P802.1Qcx YANG module

- Initial P802.1Qcx YANG module definitions



- P802.1Qcx YANG tree schema

