



Technical Report

TR-383

Common YANG Modules for Access Networks

Issue: 1 Amendment 9
July 2025

Notice

The Broadband Forum is a non-profit corporation organized to create guidelines for broadband network system development and deployment. This Technical Report has been approved by members of the Forum. This Technical Report is subject to change. This Technical Report is owned and copyrighted by the Broadband Forum, and all rights are reserved. Portions of this Technical Report may be owned and/or copyrighted by Broadband Forum members.

Intellectual Property

Recipients of this Technical Report are requested to submit, with their comments, notification of any relevant patent claims or other intellectual property rights of which they may be aware that might be infringed by any implementation of this Technical Report, or use of any software code normatively referenced in this Technical Report, and to provide supporting documentation.

Terms of Use

1. License

Broadband Forum hereby grants you the right, without charge, on a perpetual, non-exclusive and worldwide basis, to utilize the Technical Report for the purpose of developing, making, having made, using, marketing, importing, offering to sell or license, and selling or licensing, and to otherwise distribute, products complying with the Technical Report, in all cases subject to the conditions set forth in this notice and any relevant patent and other intellectual property rights of third parties (which may include members of Broadband Forum). This license grant does not include the right to sublicense, modify or create derivative works based upon the Technical Report except to the extent this Technical Report includes text implementable in computer code, in which case your right under this License to create and modify derivative works is limited to modifying and creating derivative works of such code. For the avoidance of doubt, except as qualified by the preceding sentence, products implementing this Technical Report are not deemed to be derivative works of the Technical Report.

2. NO WARRANTIES

THIS TECHNICAL REPORT IS BEING OFFERED WITHOUT ANY WARRANTY WHATSOEVER, AND IN PARTICULAR, ANY WARRANTY OF NONINFRINGEMENT AND ANY IMPLIED WARRANTIES ARE EXPRESSLY DISCLAIMED. ANY USE OF THIS TECHNICAL REPORT SHALL BE MADE ENTIRELY AT THE USER'S OR IMPLEMENTER'S OWN RISK, AND NEITHER THE BROADBAND FORUM, NOR ANY OF ITS MEMBERS OR SUBMITTERS, SHALL HAVE ANY LIABILITY WHATSOEVER TO ANY USER, IMPLEMENTER, OR THIRD PARTY FOR ANY DAMAGES OF ANY NATURE WHATSOEVER, DIRECTLY OR INDIRECTLY, ARISING FROM THE USE OF THIS TECHNICAL REPORT, INCLUDING BUT NOT LIMITED TO, ANY CONSEQUENTIAL, SPECIAL, PUNITIVE, INCIDENTAL, AND INDIRECT DAMAGES.

3. THIRD PARTY RIGHTS

Without limiting the generality of Section 2 above, BROADBAND FORUM ASSUMES NO RESPONSIBILITY TO COMPILE, CONFIRM, UPDATE OR MAKE PUBLIC ANY THIRD PARTY ASSERTIONS OF PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS THAT MIGHT NOW OR IN THE FUTURE BE INFRINGED BY AN IMPLEMENTATION OF THE TECHNICAL REPORT IN ITS CURRENT, OR IN ANY FUTURE FORM. IF ANY SUCH RIGHTS ARE DESCRIBED ON THE TECHNICAL REPORT, BROADBAND FORUM TAKES NO POSITION AS TO THE VALIDITY OR INVALIDITY OF SUCH ASSERTIONS, OR THAT ALL SUCH ASSERTIONS THAT HAVE OR MAY BE MADE ARE SO LISTED.

All copies of this Technical Report (or any portion hereof) must include the notices, legends, and other provisions set forth on this page.

Issue History

Issue Number	Approval Date	Issue Editor	Changes
Issue 1	June 2017	<ul style="list-style-type: none"> Joey Boyd, Adtran Ludwig Pauwels, Nokia 	<ul style="list-style-type: none"> Initial version
Amendment 1	July 2018	<ul style="list-style-type: none"> Joey Boyd, Adtran Ludwig Pauwels, Nokia 	<ul style="list-style-type: none"> Provide YANG model updates for Layer 2 Forwarding and QoS; publish initial model for Layer 2 Multicast Management; remove YANG models with dependencies on a draft revision of ietf-hardware.
Amendment 2	December 2018	<ul style="list-style-type: none"> Joey Boyd, Adtran Ludwig Pauwels, Nokia 	<ul style="list-style-type: none"> Add 'ethernet-like' abstract interface type.
Amendment 3	October 2020	<ul style="list-style-type: none"> Nick Hancock, Adtran Ludwig Pauwels, Nokia 	<ul style="list-style-type: none"> Enhancements to existing models. New models for ANCP and Hardware Management.
Amendment 4	June 2021	<ul style="list-style-type: none"> Nick Hancock, Adtran Ludwig Pauwels, Nokia 	<ul style="list-style-type: none"> Enhancements to existing models. New modules added to common equipment, QoS, sub-interfaces and types.
Amendment 5	March 2022	<ul style="list-style-type: none"> Nick Hancock, Adtran Ludwig Pauwels, Nokia 	<ul style="list-style-type: none"> Enhancements to existing models. New models for Device Aggregation, Connectivity Fault Management (CFM) and Software Management. New modules added to ANCP and interfaces.
Amendment 6	March 2023	<ul style="list-style-type: none"> Nick Hancock, Adtran Ludwig Pauwels, Nokia 	<ul style="list-style-type: none"> Enhancements to existing models, including extensions to the VLAN sub-interface profiles for DHCPv4, DHCPv6, PPPoE, and QoS. New models for FastDSL hardware support, frame processing, and editing.
Amendment 7	December 2023	<ul style="list-style-type: none"> Nick Hancock, Adtran Ludwig Pauwels, Nokia 	<ul style="list-style-type: none"> Enhancements to existing models. Modifications to existing models to enable reuse of common YANG schemas.

Issue Number	Approval Date	Issue Editor	Changes
			<ul style="list-style-type: none"> • New models for ICMPv6 and network functions. • New modules added to DHCP, layer 2 forwarding, layer 2 multicast, PPPoE and subscribers.
Amendment 8	November 2024	<ul style="list-style-type: none"> • Shiya Ashraf, Nokia • Nick Hancock, Adtran • Kevin Noll, CableLabs • Ludwig Pauwels, Nokia 	<ul style="list-style-type: none"> • Enhancements to existing models. • New models for Ethernet OAM, IP interfaces and IPFIX.
Amendment 9	July 2025	<ul style="list-style-type: none"> • Shiya Ashraf, Nokia • Kevin Noll, CableLabs 	<ul style="list-style-type: none"> • Enhancements to existing models. • New models for management of voice services.

Comments or questions about this Broadband Forum Technical Report should be directed to info@broadband-forum.org.

Provider Cloud Work Area Directors

- Bruno Cornaglia, Vodafone
- Mengmeng Li, China Mobile
- Haomian Zheng, Huawei

Cloud Interfaces Project Stream Leaders

- Sven Ooghe, Nokia
- Joey Boyd, Adtran

Editors

- Shiya Ashraf, Nokia
- Kevin Noll, CableLabs

Acknowledgments

The editors wish to acknowledge the following individuals for their contributions towards the Technical Report and the corresponding YANG data models.

Name	Company	Contributed to
Jeff Hartley	CommScope	Technical Report, YANG Models
Norbert Voigt	Adtran	YANG Models
Anand Arokiaraj	Nokia	Technical Report, YANG Models
Nick Hancock	Adtran	Technical Report, YANG Models
Ludwig Pauwels	Nokia	Technical Report, YANG Models

Table of Contents

Executive Summary	10
1 Purpose and Scope	11
1.1 Purpose	11
1.2 Scope	11
2 References and Terminology	12
2.1 Conventions	12
2.2 References	12
2.3 Definitions	17
2.4 Abbreviations	18
3 Technical Report Impact	20
3.1 Energy Efficiency	20
3.2 IPv6	20
3.3 Security	20
3.4 Privacy	20
4 Modules	21
4.1 DHCP	21
4.1.1 bbf-l2-dhcpv4-relay	21
4.1.2 bbf-l2-dhcpv4-relay-forwarding	21
4.1.3 bbf-ldra	21
4.1.4 bbf-vlan-sub-interface-profile-dhcpv4	21
4.1.5 bbf-vlan-sub-interface-profile-dhcpv6	21
4.1.6 bbf-ldra-profile-common	21
4.1.7 bbf-l2-dhcpv4-relay-profile-common	22
4.2 Equipment	22
4.2.1 bbf-hardware	22
4.2.2 bbf-hardware-cpu	22
4.2.3 bbf-hardware-storage-drives	22
4.2.4 bbf-hardware-transceivers	22
4.2.5 bbf-hardware-types	23
4.2.6 bbf-hardware-transceiver-alarm-types	23
4.3 Ethernet	23
4.3.1 bbf-ethernet-performance-management	23
4.4 Layer 2 Forwarding	23
4.4.1 bbf-l2-forwarding	23
4.4.2 bbf-l2-forwarding-shared-fdb	24
4.4.3 bbf-l2-forwarding-alarm-types	24
4.5 Interfaces	24
4.5.1 bbf-interfaces-performance-management	25
4.5.2 bbf-interfaces-statistics-management	25
4.5.3 bbf-interface-usage	25
4.5.4 bbf-ptm	25
4.5.5 bbf-l2-terminations	25
4.5.6 bbf-interfaces-remote-hardware-state	25
4.5.7 bbf-fastdsl-hardware	25
4.5.8 bbf-vlan-sub-interface-profile-usage	26

4.5.9 bbf-ip-interfaces	26
4.6 PPPoE	26
4.6.1 bbf-pppoe-intermediate-agent	26
4.6.2 bbf-vlan-sub-interface-profile-pppoe	26
4.6.3 bbf-pppoe-intermediate-agent-profile-common	26
4.7 QoS	26
4.7.1 bbf-qos-classifiers	27
4.7.2 bbf-qos-filters	27
4.7.3 bbf-qos-policies	27
4.7.4 bbf-qos-policies-sub-interfaces	27
4.7.5 bbf-qos-rate-control	27
4.7.6 bbf-qos-traffic-mngt	27
4.7.7 bbf-qos-enhanced-scheduling	27
4.7.8 bbf-qos-policer-envelope-profiles	28
4.7.9 bbf-qos-policing-types	28
4.7.10 bbf-qos-policing	28
4.7.11 bbf-qos-shaping	28
4.7.12 bbf-qos-types	28
4.7.13 bbf-qos-composite-filters	28
4.7.14 bbf-qos-policies-sub-interface-rewrite	28
4.7.15 bbf-qos-traffic-mngt-state	29
4.7.16 bbf-qos-enhanced-scheduling-state	29
4.7.17 bbf-qos-policies-state	29
4.7.18 bbf-qos-policing-state	29
4.7.19 bbf-vlan-sub-interface-profile-qos	29
4.8 Sub-interfaces	29
4.8.1 bbf-frame-classification	30
4.8.2 bbf-sub-interface-tagging	30
4.8.3 bbf-sub-interfaces	30
4.8.4 bbf-frame-processing-profiles	30
4.8.5 bbf-frame-editing	30
4.8.6 bbf-frame-processing	30
4.8.7 bbf-vlan-sub-interface-profile-fp	31
4.8.8 bbf-vlan-sub-interface-profiles	31
4.9 Subscribers	31
4.9.1 bbf-subscriber-profiles	31
4.9.2 bbf-subscriber-types	31
4.9.3 bbf-subscriber-profile-common	31
4.10 Types	32
4.10.1 bbf-dot1q-types	32
4.10.2 bbf-if-type	32
4.10.3 bbf-inet-types	32
4.10.4 bbf-yang-types	32
4.10.5 bbf-device-types	32
4.10.6 bbf-location-types	32
4.10.7 bbf-network-types	33
4.10.8 bbf-node-types	33
4.10.9 bbf-frame-processing-types	33

4.11 Common	33
4.11.1 bbf-availability	33
4.11.2 bbf-contact	33
4.11.3 bbf-device	33
4.11.4 bbf-end-user	33
4.11.5 bbf-location	33
4.12 Layer 2 Multicast	34
4.12.1 bbf-mgmd	34
4.12.2 bbf-mgmd-types	35
4.12.3 bbf-mgmd-mrd	35
4.12.4 bbf-mgmd-common	35
4.13 Alarms	35
4.13.1 bbf-alarm-types	35
4.14 ANCP	35
4.14.1 bbf-ancp	35
4.14.2 bbf-ancp-interfaces	36
4.14.3 bbf-ancp-fastdsl-access-extensions	36
4.14.4 bbf-ancp-fastdsl-threshold	36
4.14.5 bbf-ancp-alarm-types	36
4.15 Aggregation	36
4.15.1 bbf-device-aggregation	36
4.16 CFM	37
4.16.1 bbf-dot1q-cfm	37
4.16.2 bbf-dot1q-cfm-alarm-types	37
4.16.3 bbf-dot1q-cfm-interfaces	37
4.16.4 bbf-dot1q-cfm-interface-state	37
4.16.5 bbf-dot1q-cfm-l2-forwarding	37
4.16.6 bbf-eth-oam	38
4.16.7 bbf-eth-oam-interfaces	38
4.17 Software	38
4.17.1 bbf-software-management	38
4.17.2 bbf-software-management-voice	38
4.18 ICMPv6	38
4.18.1 bbf-icmpv6	38
4.18.2 bbf-icmpv6-forwarding	39
4.18.3 bbf-icmpv6-profile-common	39
4.18.4 bbf-vlan-sub-interface-profile-icmpv6	39
4.19 Network Functions	39
4.19.1 bbf-grpc-client	39
4.19.2 bbf-grpc-server	39
4.19.3 bbf-kafka-agent	40
4.19.4 bbf-network-function	40
4.19.5 bbf-network-function-client	40
4.19.6 bbf-network-function-server	40
4.19.7 bbf-network-function-types	40
4.20 IPFIX	40
4.20.1 bbf-ipfix-export	41
4.21 Voice	41

4.21.1 bbf-pots-alarm-types.yang	41
4.21.2 bbf-pots.yang	41
4.21.3 bbf-ringing-patterns-common.yang	41
4.21.4 bbf-ringing-patterns.yang	41
4.21.5 bbf-tone-patterns-common.yang	41
4.21.6 bbf-tone-patterns.yang	42
4.21.7 bbf-voip-media-alarm-types.yang	42
4.21.8 bbf-voip-media.yang	42
4.21.9 bbf-voip-sip-alarm-types.yang	42
4.21.10 bbf-voip-sip.yang	42
4.21.11 bbf-voip-types.yang	42
5 Documentation	44
6 Dependencies on related YANG modules and Standards	45
7 Layer 2 Forwarding Data Model	46
7.1 Sub-interfaces	46
7.1.1 Interface Usage	47
7.2 Forwarders	47
7.2.1 Forwarder Ports and Port Groups	47
7.2.2 Split Horizon Profiles	47
7.2.3 MAC Learning	47
7.2.4 Flooding	48
7.3 Interface Usage	48
7.4 Layer 2 Termination Interfaces	48
8 Ethernet-like Interfaces	50
9 Alarms	53
9.1 Alarms and Alarm Types	53
9.1.1 Common Alarm Types	53
9.1.2 Application-specific Alarm Types	55
10 Access Node Control Protocol	57
10.1 Partitions, Sessions, and Adjacencies	57
10.1.1 Create a Partition	57
10.1.2 Assigning Access Lines to a Partition	58
10.1.3 Create a Session	58
10.2 Topology Discovery	58
10.3 Access Line Identification	58
10.3.1 Access-Loop-Circuit-ID	58
10.3.2 Access-Loop-Remote-ID	59
10.3.3 Access-Aggregation-Circuit-ID-Binary and Access-Aggregation-Circuit-ID-ASCII	59
10.3.4 Additional Formatting	59
10.3.5 Supporting FastDSL Bonding	59
10.4 Controlling Port Messages	59
10.4.1 Threshold-based Reporting	59
10.4.2 Delaying the Initial Port Up Message	60
10.4.3 Dampening Mechanism	60
10.5 Statistics	60
10.6 Alarms and the Operational State of a Session	60

11 Software Management	63
11.1 Components, Software and Revisions	63
11.2 Management Capabilities	65
11.3 Managing Revisions	65
11.3.1 Download	66
11.3.2 Activate	67
11.3.3 Commit	67
11.3.4 Delete	68
11.4 Supporting the Management of Software Upgrade Processes of FastDSL NTs	68
12 IPFIX	70
12.1 Exporting Process	72
12.2 Exporter	72
12.3 Templates	72
12.4 Options Templates	72
12.5 Security	72
12.6 Transport Session	72
13 QoS Policy Data model	74
13.1 QoS Policy Profiles	74
13.1.1 Ingress QoS Policy Profile References	75
13.1.2 Egress QoS Policy Profile References	75
13.2 QoS Classifiers	75
13.2.1 QoS Classifier Filters	75
13.2.2 QoS Classifier Action List	76
13.3 QoS Metadata field associated with frames	76
13.3.1 P-Bit Marking List	76
13.3.2 DEI Marking List	77
13.3.3 Flow Color	77
13.3.4 BAC Color	77
13.3.5 Scheduling Traffic Class.	77

List of Figures

Figure 1 – YANG Data Model Relationships	11
Figure 2 – Sub-Interface Example	46
Figure 3 – Forwarder Ports	47
Figure 4 – Relationships Between Partitions, Sessions, and Interfaces	57
Figure 5 – Statechart of an ANCP Session (Informational Only)	60
Figure 6 – UML Diagram Showing the Relationship Between Software and Hardware Components	64
Figure 7 – State Machine of a Revision	65
Figure 8 – UML Diagram Showing Relationship Between the Main Components of bbfx-ipfix-data-export	70
Figure 9 – QoS Policy Profile	74
Figure 10 – QoS Profile Association Points	74

List of Tables

Table 1 – Abstract BBF Alarm Types and Associated Alarm Information	54
Table 2 – Capabilities Required to Support the Management of Fastdsl NT Software Images According to ITU-T G.997.2 Annex S	69

Executive Summary

This Technical Report defines YANG data models for the management of Broadband-Forum-specified access network equipment and network functions used across many deployment scenarios. There is no assumption for Broadband Forum YANG modules to apply outside the scope of Broadband Forum requirements.

The models specified in this Technical Report are independent of any management protocol, such as RESTCONF and NETCONF.

Amendment 9 to Issue 1 of this Technical Report expands on Amendment 8 to Issue 1 of this Technical Report as follows:

- adds initial revisions of the following YANG modules:
 - voice
 - bbf-pots-alarm-types
 - bbf-pots
 - bbf-ringing-patterns-common
 - bbf-ringing-patterns
 - bbf-tone-patterns-common
 - bbf-tone-patterns
 - bbf-voip-media-alarm-types
 - bbf-voip-media
 - bbf-voip-sip-alarm-types
 - bbf-voip-sip
 - bbf-voip-types
- modifies the following YANG modules as described below:
 - equipment
 - bbf-hardware-transceivers
 - extend transceiver compliance codes to include "Fiber Channel Speed 2" and "Extended Specification" codes.
 - add support for setting the thresholds on diagnostic values for raising alarms also at the transceiver-link level.
 - forwarding
 - bbf-l2-forwarding
 - remove the must constraint that restricted two forwarder ports referencing the same sub-interface but add this to the description.
- makes solely editorial changes to the following YANG modules:
 - QoS
 - bbf-qos-classifiers
 - bbf-qos-policies
 - bbf-qos-policing

1 Purpose and Scope

1.1 Purpose

This Technical Report defines YANG data models for the management of Broadband-Forum-specified access network equipment used across many deployment scenarios. Broadband-Forum-specified access network equipment comprises Access Nodes, FTTdp DPU's and Access Network Functions.

The models specified in this Technical Report are independent of any management protocol.

1.2 Scope

The data models defined by this Technical Report support the Broadband Forum requirements as applicable to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPU's) and form the set of core models which can be used for a multitude of other applications. It is intended that data models which are application-specific can be built on, reference, and/or function alongside the common models.

The figure below provides a high level view of the functionality covered by this Technical Report and also how they relate to YANG modules of other Standards Developing Organizations (BBF YANG in green):

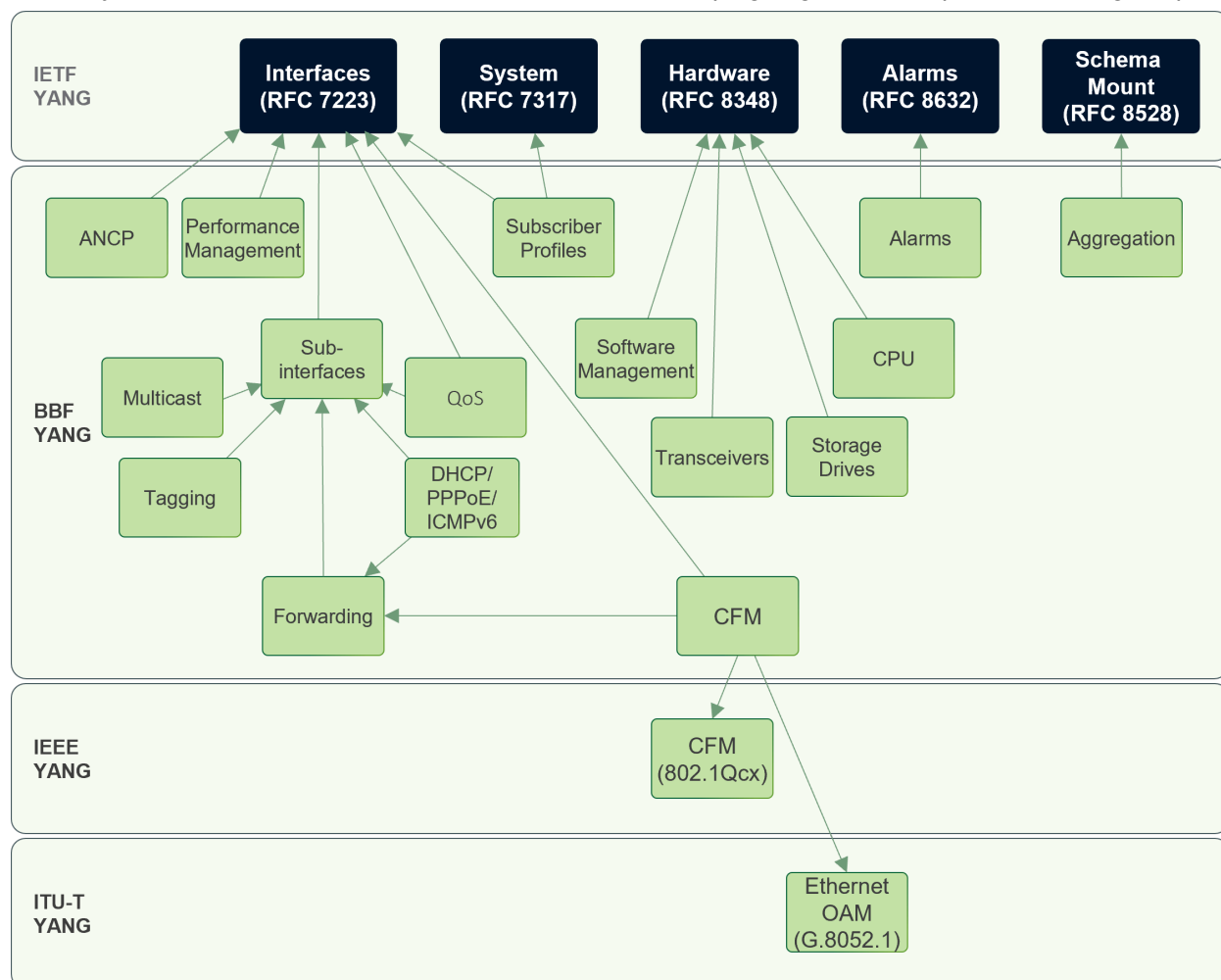


Figure 1 – YANG Data Model Relationships

2 References and Terminology

2.1 Conventions

In this Technical Report, several words are used to signify the requirements of the specification and RFC 8174[105]. These words are always capitalized. More information can be found in RFC 2119[31]. The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119][31][RFC8174][105] when, and only when, they appear in all capitals, as shown here.

2.2 References

The following references are of relevance to this Technical Report. At the time of publication, the editions indicated were valid. All references are subject to revision; users of this Technical Report are therefore encouraged to investigate the possibility of applying the most recent edition of the references listed below.

A list of currently valid Broadband Forum Technical Reports is published at <https://www.broadband-forum.org>.

- [1] TR-101 Issue 2, *Migration to Ethernet-Based Broadband Aggregation*, Broadband Forum, 2011
- [2] TR-147, *Layer 2 Control Mechanism for Broadband Multi-Service Architectures*, Broadband Forum, 2008
- [3] TR-156 Issue 4, *Using GPON Access in the context of TR-101*, Broadband Forum, 2017
- [4] TR-177 Corrigendum 1, *IPv6 in the context of TR-101*, Broadband Forum, 2017
- [5] TR-178 Issue 2, *Multi-service Broadband Network Architecture and Nodal Requirements*, Broadband Forum, 2017
- [6] TR-254, *Functionality Tests for Ethernet Based Access Nodes*, Broadband Forum, 2012
- [7] TR-301 Issue 2 Amendment 1, *Architecture and Requirements for Fiber to the Distribution Point*, Broadband Forum, 2020
- [8] TR-352 Issue 2 Corrigendum 1, *Multi-wavelength PON Inter-Channel-Termination Protocol (ICTP) Specification*, Broadband Forum, 2021
- [9] TR-355 Amendment 4, *YANG Modules for FTTdp Management*, Broadband Forum, 2022
- [10] TR-451 Amendment 1, *vOMCI Specification*, Broadband Forum, 2025
- [11] TR-489, *ONU Authentication and Selection of eOMCI or vOMCI*, Broadband Forum, 2023
- [12] 3GPP TR 32.859, *Telecommunication management; Study on Alarm Management*, 3GPP, 2013
- [13] 3GPP TS 32.111-2, *Telecommunication management; Fault Management; Part 2: Alarm Integration Reference Point (IRP): Information Service (IS) (Release 14)*, 3GPP, 2017
- [14] *Assigned Names and Numbers*, BBF; BBF
- [15] ETSI EN 300 089 V3.1.1, *Integrated Services Digital Network (ISDN); Calling Line Identification Presentation (CLIP) supplementary service; Service description*, ETSI, 2000
- [16] ETSI ES 201 235-3 V1.3.1, *Access and Terminals (AT); Specification of Dual-Tone Multi-Frequency (DTMF) Transmitters and Receivers; Part 3: Receivers*, ETSI, 2006

- [17] ETSI TR 101 183 V1.1.1, Public Switched Telephone Network (PSTN); Analogue ringing signals, ETSI, 1998
- [18] ETSI TR 101 728 V1.2.1, Access and Terminals (AT); Study for the specification of the low pass section of POTS/ADSL splitters, ETSI, 2002
- [19] ETSI TR 101 973-1 V1.1.1, Access and Terminals (AT); Public Switched Telephone Network; Support of legacy terminals by BroadBand IP equipment; Listing of the most relevant features and functionalities; Part 1: General, ETSI, 2002
- [20] ETSI TS 124 410 V7.0.0, Digital cellular telecommunications system (Phase 2+); Universal Mobile Telecommunications System (UMTS); TISPAN; NGN Signalling Control Protocol; Communication HOLD (HOLD) PSTN/ISDN simulation services; Protocol specification, ETSI, 2008
- [21] ETSI TS 183 043 V2.5.1, Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN); IMS-based PSTN/ISDN Emulation; Stage 3 specification, ETSI, 2011
- [22] gRPC Connection Backoff Protocol, gRPC Authors
- [23] gRPC Keepalive, gRPC Authors
- [24] Private Enterprise Numbers (PENs), IANA
- [25] IP Flow Information Export (IPFIX) Entities, IANA
- [26] IEEE Std 802.1Q-2018, IEEE Standard for Local and metropolitan area networks--Bridges and Bridged Networks, IEEE, 2018
- [27] IEEE Std 802.1Qcx-2020, YANG Data Model for Connectivity Fault Management, IEEE, 2020
- [28] IEEE Std 802.3-2015, IEEE Standard for Ethernet, IEEE, 2015
- [29] RFC 768, User Datagram Protocol, IETF, 1980
- [30] RFC 793, Transmission Control Protocol, IETF, 1981
- [31] RFC 2119, Key words for use in RFCs to Indicate Requirement Levels, IETF, 1997
- [32] RFC 2474, Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers, IETF, 1998
- [33] RFC 2543, SIP: Session Initiation Protocol, IETF, 1999
- [34] RFC 2617, HTTP Authentication: Basic and Digest Access Authentication, IETF, 1999
- [35] RFC 2697, A Single Rate Three Color Marker, IETF, 1999
- [36] RFC 2698, A Two Rate Three Color Marker, IETF, 1999
- [37] RFC 2710, Multicast Listener Discovery (MLD) for IPv6, IETF, 1999
- [38] RFC 2790, Host Resources MIB, IETF, 2000
- [39] RFC 2819, Remote Network Monitoring Management Information Base, IETF, 2000
- [40] RFC 2863, The Interfaces Group MIB, IETF, 2000
- [41] RFC 2933, Internet Group Management Protocol MIB, IETF, 2000
- [42] RFC 3046, DHCP Relay Agent Information Option, IETF, 2001
- [43] RFC 3261, SIP: Session Initiation Protocol, IETF, 2002

- [44] RFC 3323, *A Privacy Mechanism for the Session Initiation Protocol (SIP)*, IETF, 2002
- [45] RFC 3376, *Internet Group Management Protocol, Version 3*, IETF, 2002
- [46] RFC 3515, *The Session Initiation Protocol (SIP) Refer Method*, IETF, 2003
- [47] RFC 3551, *RTP Profile for Audio and Video Conferences with Minimal Control*, IETF, 2003
- [48] RFC 3590, *Source Address Selection for the Multicast Listener Discovery (MLD) Protocol*, IETF, 2003
- [49] RFC 3680, *A Session Initiation Protocol (SIP) Event Package for Registrations*, IETF, 2004
- [50] RFC 3810, *Multicast Listener Discovery Version 2 (MLDv2) for IPv6*, IETF, 2004
- [51] RFC 3842, *A Message Summary and Message Waiting Indication Event Package for the Session Initiation Protocol (SIP)*, IETF, 2004
- [52] RFC 3856, *A Presence Event Package for the Session Initiation Protocol (SIP)*, IETF, 2004
- [53] RFC 3857, *A Watcher Information Event Template-Package for the Session Initiation Protocol (SIP)*, IETF, 2004
- [54] RFC 3910, *The SPIRITS (Services in PSTN requesting Internet Services) Protocol*, IETF, 2004
- [55] RFC 3966, *The tel URI for Telephone Numbers*, IETF, 2004
- [56] RFC 4028, *Session Timers in the Session Initiation Protocol (SIP)*, IETF, 2005
- [57] RFC 4119, *A Presence-based GEOPRIV Location Object Format*, IETF, 2005
- [58] RFC 4235, *An INVITE-Initiated Dialog Event Package for the Session Initiation Protocol (SIP)*, IETF, 2005
- [59] RFC 4243, *Vendor-specific Information Suboption for the Dynamic Host Configuration Protocol (DHCP) Relay Agent Option*, IETF, 2005
- [60] RFC 4286, *Multicast Router Discovery*, IETF, 2005
- [61] RFC 4353, *A Framework for Conferencing with the Session Initiation Protocol (SIP)*, IETF, 2006
- [62] RFC 4541, *Considerations for Internet Group Management Protocol (IGMP) and Multicast Listener Discovery (MLD) Snooping Switches*, IETF, 2006
- [63] RFC 4566, *SDP: Session Description Protocol*, IETF, 2006
- [64] RFC 4575, *A Session Initiation Protocol (SIP) Event Package for Conference State*, IETF, 2006
- [65] RFC 4580, *Dynamic Host Configuration Protocol for IPv6 (DHCPv6) Relay Agent Subscriber-ID Option*, IETF, 2006
- [66] RFC 4605, *Internet Group Management Protocol (IGMP) / Multicast Listener Discovery (MLD)-Based Multicast Forwarding ('IGMP/MLD Proxying')*, IETF, 2006
- [67] RFC 4649, *Dynamic Host Configuration Protocol for IPv6 (DHCPv6) Relay Agent Remote-ID Option*, IETF, 2006
- [68] RFC 4730, *A Session Initiation Protocol (SIP) Event Package for Key Press Stimulus (KPML)*, IETF, 2006
- [69] RFC 4733, *RTP Payload for DTMF Digits, Telephony Tones, and Telephony Signals*, IETF, 2006
- [70] RFC 4960, *Stream Control Transmission Protocol*, IETF, 2007
- [71] RFC 4961, *Symmetric RTP / RTP Control Protocol (RTCP)*, IETF, 2007

- [72] RFC 5103, *Bidirectional Flow Export Using IP Flow Information Export (IPFIX)*, IETF, 2008
- [73] RFC 5125, *Reclassification of RFC 3525 to Historic*, IETF, 2008
- [74] RFC 5362, *The Session Initiation Protocol (SIP) Pending Additions Event Package*, IETF, 2008
- [75] RFC 5473, *Reducing Redundancy in IP Flow Information Export (IPFIX) and Packet Sampling (PSAMP) Reports*, IETF, 2009
- [76] RFC 5519, *Multicast Group Membership Discovery MIB*, IETF, 2009
- [77] RFC 5589, *Session Initiation Protocol (SIP) Call Control - Transfer*, IETF, 2009
- [78] RFC 5610, *Exporting Type Information for IP Flow Information Export (IPFIX) Information Elements*, IETF, 2009
- [79] RFC 5630, *The Use of the SIPS URI Scheme in the Session Initiation Protocol (SIP)*, IETF, 2009
- [80] RFC 5851, *Framework and Requirements for an Access Node Control Mechanism in Broadband Multi-Service Networks*, IETF, 2010
- [81] RFC 5875, *An Extensible Markup Language (XML) Configuration Access Protocol (XCAP) Diff Event Package*, IETF, 2010
- [82] RFC 5989, *A SIP Event Package for Subscribing to Changes to an HTTP Resource*, IETF, 2010
- [83] RFC 6026, *Correct Transaction Handling for 2xx Responses to Session Initiation Protocol (SIP) INVITE Requests*, IETF, 2010
- [84] RFC 6035, *Session Initiation Protocol Event Package for Voice Quality Reporting*, IETF, 2010
- [85] RFC 6072, *Certificate Management Service for the Session Initiation Protocol (SIP)*, IETF, 2011
- [86] RFC 6080, *A Framework for Session Initiation Protocol User Agent Profile Delivery*, IETF, 2011
- [87] RFC 6221, *Lightweight DHCPv6 Relay Agent*, IETF, 2011
- [88] RFC 6320, *Protocol for Access Node Control Mechanism in Broadband Networks*, IETF, 2011
- [89] RFC 6337, *Session Initiation Protocol (SIP) Usage of the Offer/Answer Model*, IETF, 2011
- [90] RFC 6665, *SIP-specific Event Notification*, IETF, 2012
- [91] RFC 6724, *Default Address Selection for Internet Protocol Version 6 (IPv6)*, IETF, 2012
- [92] RFC 6728, *Configuration Data Model for the IP Flow Information Export (IPFIX) and Packet Sampling (PSAMP) Protocols*, IETF, 2012
- [93] RFC 6795, *A Session Initiation Protocol (SIP) Event Package for Session-Specific Policies*, IETF, 2012
- [94] RFC 6910, *Completion of Calls for the Session Initiation Protocol (SIP)*, IETF, 2013
- [95] RFC 6933, *Entity MIB (Version 4)*, IETF, 2013
- [96] RFC 6991, *Common YANG Data Types*, IETF, 2013
- [97] RFC 7011, *Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of Flow Information*, IETF, 2013
- [98] RFC 7012, *Information Model for IP Flow Information Export (IPFIX)*, IETF, 2013
- [99] RFC 7200, *A Session Initiation Protocol (SIP) Load-Control Event Package*, IETF, 2014
- [100] RFC 7223, *A YANG Data Model for Interface Management*, IETF, 2014

- [101] RFC 7224, *IANA Interface Type YANG Module*, IETF, 2014
- [102] RFC 7277, *A YANG Data Model for IP Management*, IETF, 2014
- [103] RFC 7317, *A YANG Data Model for System Management*, IETF, 2014
- [104] RFC 7950, *The YANG 1.1 Data Modeling Language*, IETF, 2016
- [105] RFC 8174, *Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words*, IETF, 2017
- [106] RFC 8341, *Network Configuration Access Control Model*, IETF, 2018
- [107] RFC 8342, *Network Management Datastore Architecture (NMDA)*, IETF, 2018
- [108] RFC 8348, *A YANG Data Model for Hardware Management*, IETF, 2018
- [109] RFC 8415, *Dynamic Host Configuration Protocol for IPv6 (DHCPv6)*, IETF, 2018
- [110] RFC 8446, *The Transport Layer Security (TLS) Protocol Version 1.3*, IETF, 2018
- [111] RFC 8519, *YANG Data Model for Network Access Control Lists (ACLs)*, IETF, 2019
- [112] RFC 8528, *YANG Schema Mount*, IETF, 2019
- [113] RFC 8632, *A YANG Data Model for Alarm Management*, IETF, 2019
- [114] RFC 9640, *YANG Data Types and Groupings for Cryptography*, IETF, 2024
- [115] RFC 9641, *A YANG Data Model for a Truststore*, IETF, 2024
- [116] RFC 9642, *A YANG Data Model for a Keystore*, IETF, 2024
- [117] RFC 9645, *YANG Groupings for TLS Clients and TLS Servers*, IETF, 2024
- [118] E.164, *The international public telecommunication numbering plan*, ITU-T, 2010
- [119] G.711, *Pulse code modulation (PCM) of voice frequencies*, ITU-T, 1988
- [120] G.722, *7 kHz audio-coding within 64 kbit/s*, ITU-T, 2012
- [121] G.722.1, *Low-complexity coding at 24 and 32 kbit/s for hands-free operation in systems with low frame loss*, ITU-T, 2005
- [122] G.722.2, *Wideband coding of speech at around 16 kbit/s using Adaptive Multi-Rate Wideband (AMR-WB)*, ITU-T, 2003
- [123] G.723.1, *Dual rate speech coder for multimedia communications transmitting at 5.3 and 6.3 kbit/s*, ITU-T, 2006
- [124] G.726, *40, 32, 24, 16 kbit/s Adaptive Differential Pulse Code Modulation (ADPCM)*, ITU-T, 1990
- [125] G.729, *Coding of speech at 8 kbit/s using conjugate-structure algebraic-code-excited linear prediction (CS-ACELP)*, ITU-T, 2012
- [126] G.988, *ONU management and control interface (OMCI) specification*, ITU-T, 2017
- [127] G.993.2, *Very high speed digital subscriber line transceivers 2 (VDSL2)*, ITU-T, 2019
- [128] G.997.1, *Physical layer management for digital subscriber line transceivers*, ITU-T, 2019
- [129] G.997.2, *Physical layer management for G.fast transceivers*, ITU-T, 2019
- [130] G.8013/Y.1731 (2023) Corrigendum 1 (01/24), *Operation, administration and maintenance (OAM) functions and mechanisms for Ethernet-based networks*, ITU-T, 2024

- [131] G.8021/Y.1341 (2022) Amendment 1 (01/24), *Characteristics of Ethernet transport network equipment functional blocks*, ITU-T, 2024
- [132] G.8052.1/Y.1346.1 (2021) Amendment 1 (01/2023), *Operation, administration, maintenance (OAM) management information and data models for the Ethernet-transport network element*, ITU-T, 2023
- [133] G.9701, *Fast access to subscriber terminals (G.fast) - Physical layer specification*, ITU-T, 2019
- [134] *ITU Operational Bulletin No.955*, ITU-T, 2010
- [135] Q.552, *Transmission characteristics at 2-wire analogue interfaces of digital exchanges*, ITU-T, 2001
- [136] Q.731.3, *Stage 3 description for number identification supplementary services using Signalling System No. 7 - Calling line identification presentation*, ITU-T, 2019
- [137] V.152, *Procedures for supporting voice-band data over IP networks*, ITU-T, 2010
- [138] X.731, *Information technology - Open Systems Interconnection - Systems Management: State management function*, ITU-T, 1992
- [139] X.733, *Information technology - Open Systems Interconnection - Systems Management: Alarm reporting function*, ITU-T, 1992
- [140] X.736, *Information technology - Open Systems Interconnection - Systems Management: Security alarm reporting function*, ITU-T, 1992
- [141] MEF 10.4, *Subscriber Ethernet Service Attributes*, MEF, 2018
- [142] NFPA-72, *National Fire Alarm and Signaling Code*, NFPA
- [143] INF-8077i (Revision 4.5), *10 Gigabit Small Form Factor Pluggable Module*, SFF, 2005
- [144] INF-8438i (Revision 1.0), *QSFP (Quad Small Formfactor Pluggable) Transceiver*, SFF, 2006
- [145] INF-8628 (Revision 0.0), *QSFP-DD 8X Transceiver (QSFP Double Density)*, SFF, 2016
- [146] SFF-8024 (Revision 4.11), *SFF Module Management Reference Code Tables*, SFF, 2023
- [147] SFF-8079 (Revision 1.7), *SFP Rate and Application Selection*, SFF, 2005
- [148] SFF-8431 (Revision 4.1A), *SFP+ 10 Gb/s and Low Speed Electrical Interface*, SFF, 2013
- [149] SFF-8436 (Revision 4.9), *QSFP+ 4X 10 Gb/s Pluggable Transceiver*, SFF, 2018
- [150] SFF-8472 (Revision 12.4), *Management Interface for SFP+*, SFF, 2023
- [151] SFF-8635 (Revision 0.6), *QSFP+ 4X 10 Gb/s Pluggable Transceiver Solution (QSFP10)*, SFF, 2015
- [152] SFF-8636 (Revision 2.11), *Management Interface for 4-lane Modules and Cables*, SFF, 2023
- [153] SFF-8665 (Revision 1.9), *QSFP+ 28 Gb/s 4X Pluggable Transceiver Solution (QSFP28)*, SFF, 2019
- [154] SFF-8685 (Revision 0.6), *QSFP+ 14 Gb/s 4X Pluggable Transceiver Solution (QSFP14)*, SFF, 2015
- [155] GR-31-CORE, *CLASS Feature: Calling Number Delivery (ESD 01-02-1051)*, Telcordia, 2000
- [156] GR-506-CORE, *Signaling for Analog Interfaces*, Telcordia, 2011

2.3 Definitions

The following terminology is used throughout this Technical Report.

Model	A YANG data model.
-------	--------------------

Module	A YANG module defines the hierarchy (schema tree) or part of the hierarchy of data for a data model.
Submodule	A YANG module may be broken up into multiple submodules for ease of maintainability. The overall YANG data model is comprised of a module and zero or more submodules.

2.4 Abbreviations

This Technical Report uses the following abbreviations:

AN	Access Node
ANCP	Access Node Control Protocol
BAC	Buffer Acceptance Control
CFM	Connectivity Fault Management
DEI	Drop Eligible Indicator
DHCP	Dynamic Host Configuration Protocol
DHCPRA	DHCPv4 Relay Agent
DPU	Distribution Point Unit
DSL	Digital Subscriber Line
FastDSL	DSL or G.fast
FRA	Fast Rate Adaptation
FTTdp	Fiber to the Distribution Point
FTU	FAST Transceiver Unit
ICMPv6	Internet Control Message Protocol version 6
ICTP	Inter-Channel-Termination Protocol
IE	Information Element
IGMP	Internet Group Management Protocol
IPFIX	Internet Protocol Flow Information Export
IPv4	Internet Protocol version 4
IPv6	Internet Protocol version 6
L2	Layer 2
LAG	Link Aggregation Group
LDRA	Lightweight DHCPv6 Relay Agent
MGMD	Multicast Group Membership Discovery
MLD	Multicast Listener Discovery
NAS	Network Access Server
NMDA	Network Management Datastore Architecture

NMS	Network Management System
NF	Network Function
NT	Network Termination
OLT	Optical Line Termination
OMCI	Optical network unit Management and Control Interface
ONU	Optical Network Unit
PMA	Persistent Management Agent
POTS	Plain Old Telephone Service
PPPoE	Point-to-Point Protocol over Ethernet
PPPoEIA	PPPoE Intermediate Agent
QoS	Quality of Service
RPC	Remote Procedure Call
SIP	Session Initiation Protocol
SRA	Seamless Rate Adaptation
TCP	Transmission Control Protocol
TLS	Transport Layer Security
TLV	Type-Length-Value
TR	Technical Report
UML	Unified Modeling Language™
URL	Uniform Resource Locator
VoIP	Voice over Internet Protocol
VSI	VLAN sub-interface
WA	Work Area
WT	Working Text

3 Technical Report Impact

3.1 Energy Efficiency

TR-383 has no impact on energy efficiency.

3.2 IPv6

TR-383 includes YANG modules that support IPv6 deployments.

3.3 Security

TR-383 has no impact on security.

3.4 Privacy

Any issues regarding privacy are not affected by TR-383.

4 Modules

The YANG modules contained in TR-383 are briefly described here. These modules are published on GitHub at <https://github.com/BroadbandForum/yang/>.

4.1 DHCP

These modules provide functionality to manage DHCP and can be found in the *networking* directory on GitHub.

4.1.1 bbf-l2-dhcpv4-relay

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on subscriber management via the DHCPv4 protocol as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

4.1.2 bbf-l2-dhcpv4-relay-forwarding

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on subscriber management via the DHCPv4 protocol as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module augments bbf-l2-forwarding with subscriber management via the DHCPv4 protocol [42].

4.1.3 bbf-ldra

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on subscriber management via the DHCPv6 protocol as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

This functionality is also known as a Lightweight DHCPv6 Relay Agent (LDRA) [87].

4.1.4 bbf-vlan-sub-interface-profile-dhcpv4

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on the management of sub-interfaces as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module extends the VLAN sub-interface profile with DHCPv4 data nodes.

4.1.5 bbf-vlan-sub-interface-profile-dhcpv6

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on the management of sub-interfaces as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module extends the VLAN sub-interface profile with DHCPv6 data nodes.

4.1.6 bbf-ldra-profile-common

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on subscriber management via the DHCPv6 protocol as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

This functionality is also known as a Lightweight DHCPv6 Relay Agent (LDRA).

Specifically, this module contains the definition of a profile for subscriber information used by an LDRA in creating an Agent Circuit ID and Remote ID as described in TR-101i2 [1].

4.1.7 bbf-l2-dhcpv4-relay-profile-common

This module contains a collection of common YANG definitions for supporting the Broadband Forum requirements on subscriber management via the DHCPv4 protocol as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPU's).

Specifically, this module contains the definition of a profile for subscriber information used by an L2 DHCPv4 Relay Agent in creating an Agent Circuit ID and Remote ID as described in TR-101i2 [1].

4.2 Equipment

These modules provide management extensions related to hardware components as defined in the IETF RFC 8348 [108] and can be found in the *equipment* directory on GitHub.

4.2.1 bbf-hardware

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on hardware and interface management as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPU's).

Specifically, this module augments the ietf-hardware model with additional management common to multiple classes of hardware components and augments the ietf-interfaces model to enable interfaces to reference hardware components.

4.2.2 bbf-hardware-cpu

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on hardware management as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPU's).

Specifically, this module augments the ietf-hardware model with the management of a CPU processor (with single or multiple cores).

4.2.3 bbf-hardware-storage-drives

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on hardware management as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPU's).

Specifically, this module augments the ietf-hardware model with the management of hardware components with data storage capability as their main functionality, e.g., hard disk drive (HDD), solid-state device (SSD), solid-state hybrid drive (SSHD), object storage device (OSD), or other.

4.2.4 bbf-hardware-transceivers

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on hardware management as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPU's).

Specifically, this module augments the ietf-hardware model with management of compact transceivers.

4.2.5 bbf-hardware-types

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on reusable data types as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module specializes types defined in the *iana-hardware* model.

4.2.6 bbf-hardware-transceiver-alarm-types

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on hardware management as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module contains a set of alarm definitions related to the management of compact transceivers.

4.3 Ethernet

These modules are specific to the management of Ethernet interfaces as defined by IEEE 802.3 [28] and can be found in the *interface* directory on GitHub.

4.3.1 bbf-ethernet-performance-management

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on Ethernet interface management as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module augments *bbf-interface-performance-management* with Ethernet-specific counters.

4.4 Layer 2 Forwarding

These modules and their submodules are used for the management of Layer 2 (L2) Forwarding and can be found in the *networking* directory on GitHub.

4.4.1 bbf-l2-forwarding

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on layer 2 forwarding as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

4.4.1.1 bbf-l2-forwarding-base

This submodule contains a collection of YANG definitions for defining the top-level nodes for forwarding.

4.4.1.2 bbf-l2-forwarding-flooding-policies

This submodule contains a collection of YANG definitions for managing flooding policies.

Flooding policies define how the system forwards frames in case other forwarding mechanisms do not arrive at a forwarding decision.

4.4.1.3 bbf-l2-forwarding-forwarders

This submodule contains a collection of YANG definitions for managing forwarders.

A forwarder is used to forward traffic between two or more interfaces.

4.4.1.4 bbf-l2-forwarding-forwarding-databases

This submodule contains a collection of YANG definitions for managing forwarding databases.

A forwarding database contains the necessary information regarding the MAC addresses which are used in the forwarding decision.

4.4.1.5 bbf-l2-forwarding-mac-learning-control

This submodule contains a collection of YANG definitions for managing MAC address learning constraints, i.e., to constrain MAC learning rules compared with the standard IEEE MAC learning.

4.4.1.6 bbf-l2-forwarding-mac-learning

This submodule contains a collection of YANG definitions for managing MAC learning.

For a forwarder, it specifies the forwarding database to use for the specified forwarder. For an interface, it provides the ability to enable/disable MAC learning as well as specifies other parameters associated with MAC learning.

4.4.1.7 bbf-l2-forwarding-split-horizon-profiles

This submodule contains a collection of YANG definitions for managing split horizon profiles.

These profiles allow (or disallow) forwarding between various forwarder ports based on the underlying interface usage.

4.4.1.8 bbf-l2-forwarding-shared-fdb

Replaced by [Section 4.4.2](#).

4.4.2 bbf-l2-forwarding-shared-fdb

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on layer 2 forwarding as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module contains a collection of YANG definitions for managing shared forwarding databases.

4.4.3 bbf-l2-forwarding-alarm-types

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on layer 2 forwarding as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module contains a set of alarm definitions related to the management of L2 forwarders.

4.5 Interfaces

These modules augment `ietf-interfaces` [100] with additional interface management and can be found in the *interface* directory on GitHub.

4.5.1 bbf-interfaces-performance-management

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on interface management as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module reports performance management of statistics defined by the IETF interfaces data model, ietf-interfaces (RFC 7223) [100].

4.5.2 bbf-interfaces-statistics-management

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on interface management as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module augments ietf-interfaces [100] with a reset action for statistics.

4.5.3 bbf-interface-usage

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on interface management as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module contains a collection of YANG definitions defining how interfaces are used.

4.5.4 bbf-ptm

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on interface management as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module augments the IETF interfaces data model, ietf-interfaces (RFC 7223) [100], with nodes for managing Packet Transfer Mode (PTM) interfaces.

4.5.5 bbf-l2-terminations

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on the management of Layer 2 terminations as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

4.5.6 bbf-interfaces-remote-hardware-state

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on interface management as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module augments the IETF interfaces data model, ietf-interfaces (RFC 7223), with nodes for monitoring the software of remote hardware.

4.5.7 bbf-fastdsl-hardware

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on interface management as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module augments the IETF interfaces data model, *ietf-interfaces* (RFC7223), with nodes for managing the software of NTs connected to FAST or to DSL lines.

4.5.8 bbf-vlan-sub-interface-profile-usage

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on the management of sub-interfaces as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module extends the VLAN sub-interface profile with interface usage data nodes.

4.5.9 bbf-ip-interfaces

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on the management of IP host interfaces as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module defines data nodes used to manage the linkage between a Layer 3 IP host interface and its underlying Layer 2 interface, i.e., a VLAN sub-interface or a Layer 2 termination.

Managing the Layer 2 and Layer 3 interfaces as separate interfaces results in statistics being reported both at Layer 2 (frame) and at Layer 3 (packet).

4.6 PPPoE

These modules provide functionality to manage Point-to-Point Protocol over Ethernet and can be found in the *networking* directory on GitHub.

4.6.1 bbf-pppoe-intermediate-agent

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on subscriber management via the PPPoE protocol as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified access Nodes and FTTdp DPUs).

4.6.2 bbf-vlan-sub-interface-profile-pppoe

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on the management of sub-interfaces as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module extends the VLAN sub-interface profile with PPPoE data nodes.

4.6.3 bbf-pppoe-intermediate-agent-profile-common

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on subscriber management via the PPPoE protocol as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module contains the definition of a profile for subscriber information used by a PPPoE Intermediate Agent in creating an Agent Circuit ID and Remote ID as described in TR-101i2 [1].

4.7 QoS

These modules provide functionality to manage Quality of Service (QoS) and can be found in the *networking* directory on GitHub.

4.7.1 bbf-qos-classifiers

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on the management of Quality of Service (QoS) as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module contains a collection of classifiers that can be used to classify frames and assign actions to be applied to those frames.

4.7.2 bbf-qos-filters

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on the management of Quality of Service (QoS) as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module contains definitions of filter criteria.

4.7.3 bbf-qos-policies

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on the management of Quality of Service (QoS) as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module contains a collection of policies that can be used to control the flow of packets.

4.7.4 bbf-qos-policies-sub-interfaces

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on the management of Quality of Service (QoS) as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module augments sub-interfaces to support policies to control the flow of packets.

4.7.5 bbf-qos-rate-control

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on the management of Quality of Service (QoS) as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module augments classifiers to control frame rates.

4.7.6 bbf-qos-traffic-mngt

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on the management of Quality of Service (QoS) as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module contains a collection of functions for QoS traffic management (TM).

4.7.7 bbf-qos-enhanced-scheduling

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on the management of Quality of Service (QoS) as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module augments interfaces to add additional configuration to manage enhanced traffic scheduling.

4.7.8 bbf-qos-policer-envelope-profiles

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on the management of Quality of Service (QoS) as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module augments classifiers to add management of envelope policing as described in MEF 10.4 [141].

4.7.9 bbf-qos-policing-types

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on the management of Quality of Service (QoS) as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module contains common types used for management of policers.

4.7.10 bbf-qos-policing

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on the management of Quality of Service (QoS) as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module augments classifiers to manage the policing of flows.

4.7.11 bbf-qos-shaping

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on the management of Quality of Service (QoS) as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module augments traffic management profiles with shaper profiles and augments interfaces to reference a shaper profile.

4.7.12 bbf-qos-types

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on the management of Quality of Service (QoS) as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module contains type definitions used in multiple QoS modules.

4.7.13 bbf-qos-composite-filters

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on the management of Quality of Service (QoS) as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module contains definitions of filter criteria for Ethernet header fields, IPv4 and IPv6 header fields, some IP packet payload fields, and it contains filters composed of a combination of these fields.

4.7.14 bbf-qos-policies-sub-interface-rewrite

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on the management of Quality of Service (QoS) as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module contains augments to sub-interfaces to support policies applied to packets.

4.7.15 bbf-qos-traffic-mngt-state

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on the management of Quality of Service (QoS) as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module contains a collection of functions for monitoring QoS traffic management (TM). This module is to be used, along with 'bbf-qos-traffic-mngt,' when the server does not support Network Management Datastore Architecture (NMDA) as defined in RFC 8342 [107].

4.7.16 bbf-qos-enhanced-scheduling-state

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on the management of Quality of Service (QoS) as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module augments interfaces to add monitoring of enhanced traffic scheduling. This module is to be used, along with 'bbf-qos-enhanced-scheduling,' when the server does not support Network Management Datastore Architecture (NMDA) as defined in RFC 8342 [107].

4.7.17 bbf-qos-policies-state

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on the management of Quality of Service (QoS) as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module augments interfaces to add monitoring of policies that can be used to control the flow of packets. This module is to be used, along with 'bbf-qos-policies,' when the server does not support Network Management Datastore Architecture (NMDA) as defined in RFC 8342 [107].

4.7.18 bbf-qos-policing-state

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on the management of Quality of Service (QoS) as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module augments interfaces to add monitoring the policing of flows. This module is to be used, along with 'bbf-qos-policing,' when the server does not support Network Management Datastore Architecture (NMDA) as defined in RFC 8342 [107].

4.7.19 bbf-vlan-sub-interface-profile-qos

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on the management of sub-interfaces as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module extends the VLAN sub-interface profile with QoS data nodes.

4.8 Sub-interfaces

These modules provide management definitions for sub-interfaces and can be found in the *interface* directory on GitHub.

4.8.1 bbf-frame-classification

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on frame classification as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module contains reusable groupings defined for frame classification.

4.8.2 bbf-sub-interface-tagging

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on the management of sub-interfaces as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module augments the frame processing configuration of a (sub-)interface with additional criteria and adds VLAN-specific ingress and egress rewrite actions.

4.8.3 bbf-sub-interfaces

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on the management of sub-interfaces as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module augments an interface with sub-interface-specific frame processing configuration.

4.8.4 bbf-frame-processing-profiles

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on the management of sub-interfaces as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module contains the definition of a profile for classifying frames and performing ingress- and egress-rewrite actions, and it augments the frame processing configuration of a (VLAN sub-)interface for using this profile.

4.8.5 bbf-frame-editing

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on frame classification as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module contains reusable groupings defined for frame editing.

4.8.6 bbf-frame-processing

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on the management of sub-interfaces as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module defines generic groupings that can be used when defining frame processing data nodes.

4.8.7 bbf-vlan-sub-interface-profile-fp

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on the management of sub-interfaces as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module extends the VLAN sub-interface profile with data nodes supporting the frame processing, i.e., data nodes for matching frames received on a lower-layer interface into a VLAN sub-interface, data nodes indicating the associated ingress rewrite actions, and data nodes for egress rewrite actions.

4.8.8 bbf-vlan-sub-interface-profiles

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on the management of sub-interfaces as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module extends the methods for configuring a VLAN sub-interface with a method that enables management of the configuration through a profile.

4.9 Subscribers

These modules provide management of subscriber-related functionality and can be found in the *networking* directory on GitHub.

4.9.1 bbf-subscriber-profiles

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on the management of subscribers as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module adds support for subscriber profiles and augments an interface to enable an interface to reference a subscriber profile. It also augments ietf-system to add system-specific subscriber management.

4.9.2 bbf-subscriber-types

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on the management of subscribers as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module defines common types associated with subscribers and subscriber protocols.

4.9.3 bbf-subscriber-profile-common

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on the management of subscribers as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module contains the definition of a profile for subscriber information used by a L2 DHCP Relay Agent, Lightweight DHCPv6 Relay Agent, PPPoE Intermediate Agent, or another Agent in creating an Agent Circuit ID and Remote ID as described in TR-101i2 [1].

4.10 Types

These modules provide reusable type definitions for use across all BBF YANG models and can be found in the *common* directory on GitHub unless otherwise specified.

4.10.1 bbf-dot1q-types

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on reusable data types as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module defines common types for support of IEEE 802.1Q [26].

4.10.2 bbf-if-type

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on reusable data types as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module defines interface types that are needed for BBF applications but are not defined in *iana-if-type*.

This module can be found in the *interface* directory on GitHub.

4.10.3 bbf-inet-types

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on reusable data types as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module defines additional YANG data types that are useful in managing Internet-Protocol-related configuration that are not defined by the IETF.

4.10.4 bbf-yang-types

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on reusable data types as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module defines common types used throughout BBF data models.

4.10.5 bbf-device-types

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on managing physical devices.

Specifically, this module defines common types associated with device management.

4.10.6 bbf-location-types

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on providing location information associated with network functions.

Specifically, this module defines common types associated with location information.

4.10.7 bbf-network-types

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on providing network information associated with network functions.

Specifically, this module defines common types associated with network information.

4.10.8 bbf-node-types

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on providing network node information associated with network functions.

Specifically, this module defines common types associated with network nodes.

4.10.9 bbf-frame-processing-types

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on reusable data types as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module defines common types used throughout the modeling of frame processing.

4.11 Common

These modules provide support for common requirements for use across all BBF YANG models and can be found in the *common* directory on GitHub.

4.11.1 bbf-availability

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on the general availability of specific resources as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

4.11.2 bbf-contact

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on providing contact information regarding network functions.

4.11.3 bbf-device

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on managing physical devices.

4.11.4 bbf-end-user

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on providing information for end users of network functions.

4.11.5 bbf-location

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on providing location information associated with network functions.

4.12 Layer 2 Multicast

These modules and their submodules are used for the management of Layer 2 (L2) Multicast and can be found in the *networking* directory on GitHub.

4.12.1 bbf-mgmd

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on layer 2 multicast management as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module contains a collection of YANG definitions for supporting the Multicast Group Membership Discovery (MGMD) Protocols. In particular, it describes data nodes used for managing the Internet Group Management Protocol (IGMP) and the Multicast Listener Discovery (MLD) protocol in systems that act as a multicast proxy, snooper, or a snooper with proxy reporting.

4.12.1.1 bbf-mgmd-configuration-interface-to-host

This submodule has been removed.

Note that the grouping definition within the module has been moved to the main module, *bbf-mgmd*, and its status set to deprecated. Consequently, this grouping should not be used for new implementations; the same-named grouping in module *bbf-mgmd-common* should be preferred.

4.12.1.2 bbf-mgmd-configuration-interface-to-router

This submodule has been removed.

Note that the grouping definition within the module has been moved to the main module, *bbf-mgmd*, and its status set to deprecated. Consequently, this grouping should not be used for new implementations; the same-named grouping in module *bbf-mgmd-common* should be preferred.

4.12.1.3 bbf-mgmd-configuration-multicast-snoop

This submodule has been removed.

Note that the feature and grouping definitions within the module have been moved to the main module, *bbf-mgmd*, and the status of the grouping set to deprecated. Consequently, this grouping should not be used for new implementations; the same-named grouping in module *bbf-mgmd-common* should be preferred.

4.12.1.4 bbf-mgmd-operational-interface-to-host

This submodule has been removed.

Note that the grouping definitions within the module have been moved to the main module, *bbf-mgmd*, and their status set to deprecated. Consequently, these groupings should not be used for new implementations; the same-named groupings in module *bbf-mgmd-common* should be preferred.

4.12.1.5 bbf-mgmd-operational-interface-to-router

This submodule has been removed.

Note that the grouping definition within the module has been moved to the main module, *bbf-mgmd*, and its status set to deprecated. Consequently, this grouping should not be used for new implementations; the same-named grouping in module *bbf-mgmd-common* should be preferred.

4.12.2 bbf-mgmd-types

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on L2 multicast management as applicable to access network equipment. This module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module contains a collection of YANG type and feature definitions for use in modules supporting the Multicast Group Membership Discovery (MGMD) Protocols. In particular, it describes data nodes used for managing the Internet Group Management Protocol (IGMP) and the Multicast Listener Discovery (MLD) protocol.

4.12.3 bbf-mgmd-mrd

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on layer 2 multicast management as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module contains a collection of YANG definitions for supporting the Multicast Group Membership Discovery (MGMD) Protocols. In particular, it describes data nodes used for managing the Internet Group Management Protocol (IGMP) and the Multicast Listener Discovery (MLD) protocol in systems that act as a multicast proxy, snooper, or a snooper with proxy reporting.

4.12.4 bbf-mgmd-common

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on Multicast Group Membership Discovery (MGMD) protocols.

Specifically, this module describes reusable groupings containing configuration and operational data nodes used for managing the Internet Group Management Protocol (IGMP) and the Multicast Listener Discovery (MLD) protocol in case the system acts as a snooper.

4.13 Alarms

These modules add BBF-specific alarm definitions based on ietf-alarms (RFC 8632 [113]) and can be found in the *common* directory on GitHub.

4.13.1 bbf-alarm-types

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on the management of alarms as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module defines abstract alarm types that are needed for BBF applications to be able to define their own specific abstract and concrete alarm types.

4.14 ANCP

These modules provide management of the Access Node Control Protocol (ANCP) and can be found in the *networking* directory on GitHub.

4.14.1 bbf-ancp

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on the Access Node Control Protocol (ANCP) as defined in RFC 6320 [88]. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

This data model is designed for the Network Management Datastore Architecture defined in RFC 8342 [107].

4.14.2 bbf-ancp-interfaces

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on the Access Node Control Protocol (ANCP) as defined in RFC 6320 [88]. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module augments ietf-interfaces [100] to manage individual access lines that participate in ANCP.

4.14.3 bbf-ancp-fastdsl-access-extensions

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on the Access Node Control Protocol (ANCP) as defined in RFC 6320 [88]. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module augments bbf-ancp to extend the definitions for FastDSL access technologies, which are used in the management of the Access Node side of the protocol.

4.14.4 bbf-ancp-fastdsl-threshold

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on the Access Node Control Protocol (ANCP) as defined in RFC 6320 [88]. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module augments bbf-ancp to add the data nodes to manage line state reporting of the Access Node.

This data model is designed for the Network Management Datastore Architecture defined in RFC 6320 [88]. The line state reporting requirements are defined in BBF TR-147 [2].

4.14.5 bbf-ancp-alarm-types

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on the Access Node Control Protocol (ANCP) as defined in RFC 6320 [88]. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module contains a set of alarm definitions.

4.15 Aggregation

This module supports the aggregation of devices and can be found in the *common* directory on GitHub.

4.15.1 bbf-device-aggregation

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on device aggregation. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module can be used in entities that have a need to maintain a list of devices expected to be managed.

For example, this module could be used to support the Persistent Management Agent (PMA) aggregation function described in TR-301 [7]. As another example, this module could also be used in the Management

Plane to support the Optical Network Unit (ONU) aggregation function required by TR-489 [11] in some scenarios.

4.16 CFM

These modules bind the IEEE 802.1Qcx CFM YANG data model [27] to the Broadband Forum Layer 2 Forwarding and Alarm Management and can be found in the *networking* directory on GitHub.

4.16.1 bbf-dot1q-cfm

This module is part of a collection of YANG definitions for supporting the Broadband Forum requirements on Connectivity Fault Management (CFM) Operations, Administration, and Maintenance (OAM) as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module augments the IEEE 802.1Qcx CFM YANG data model [27] with data nodes to support BBF-specific requirements.

4.16.2 bbf-dot1q-cfm-alarm-types

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on Connectivity Fault Management (CFM) Operations, Administration, and Maintenance (OAM) as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module contains a set of alarm definitions.

4.16.3 bbf-dot1q-cfm-interfaces

This module is part of a collection of YANG definitions for supporting the Broadband Forum requirements on Connectivity Fault Management (CFM) Operations, Administration, and Maintenance (OAM) as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module augments the IEEE 802.1Qcx CFM YANG data model [27] to link a Maintenance Association (MA) End Point (MEP) to an interface.

4.16.4 bbf-dot1q-cfm-interface-state

This module is part of a collection of YANG definitions for supporting the Broadband Forum requirements on Connectivity Fault Management (CFM) Operations, Administration, and Maintenance (OAM) as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module augments the IEEE 802.1Qcx CFM YANG data model [27] to provide data nodes to display Maintenance Points present on an interface, i.e., the CFM stack of an interface.

4.16.5 bbf-dot1q-cfm-l2-forwarding

This module is part of a collection of YANG definitions for supporting the Broadband Forum requirements on Connectivity Fault Management (CFM) Operations, Administration, and Maintenance (OAM) as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module augments the IEEE 802.1Qcx CFM YANG data model [27] to link a Maintenance Association (MA) to a forwarder.

4.16.6 bbf-eth-oam

This module is part of a collection of YANG definitions for supporting the Broadband Forum requirements on Connectivity Fault Management (CFM) Operations, Administration, and Maintenance (OAM) as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module augments the IEEE 802.1Qcx CFM YANG data model [27] with data nodes to support BBF-specific requirements relating to Ethernet OAM capabilities defined in ITU-T G.8013 [130] and ITU-T G.8021 [131].

4.16.7 bbf-eth-oam-interfaces

This module is part of a collection of YANG definitions for supporting the Broadband Forum requirements on Connectivity Fault Management (CFM) Operations, Administration, and Maintenance (OAM) as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module augments the ITU-T G.8052.1 Carrier Ethernet OAM YANG data model [132] to link a Maintenance Entity Group (MEG) Intermediate Point (MIP) to an interface.

4.17 Software

These modules provide management of software on hardware components and can be found in the *equipment* directory on GitHub.

4.17.1 bbf-software-management

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on software management as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

This module augments ietf-hardware.

4.17.2 bbf-software-management-voice

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on software management specific to voice applications as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

4.18 ICMPv6

These modules support subscriber management via the Internet Control Message Protocol version 6 (ICMPv6) and can be found in the *networking* directory on GitHub.

4.18.1 bbf-icmpv6

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on subscriber management via the Internet Control Message Protocol version 6 (ICMPv6) as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

The functionality to realize BBF's ICMPv6 protocol requirements is called 'ICMPv6 Relay'.

4.18.2 bbf-icmpv6-forwarding

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on subscriber management via the Internet Control Message Protocol version 6 (ICMPv6) protocol as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module augments bbf-l2-forwarding with data nodes controlling the forwarding of ICMPv6 messages.

4.18.3 bbf-icmpv6-profile-common

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on subscriber management via the Internet Control Message Protocol version 6 (ICMPv6) as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

The functionality to realize BBF's ICMPv6 protocol requirements is called 'ICMPv6 Relay'.

Specifically, this module contains the definition of a profile for subscriber information used by an ICMPv6 Relay in creating an Agent Circuit ID and Remote ID as described in TR-101i2 [1].

4.18.4 bbf-vlan-sub-interface-profile-icmpv6

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on the management of sub-interfaces as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module extends the VLAN sub-interface profile with ICMPv6 data nodes.

4.19 Network Functions

These modules support the management of network functions (NF) and can be found in the *common* directory on GitHub.

4.19.1 bbf-grpc-client

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on network functions as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module defines a set of common groupings to support the gRPC client interface to initiate and maintain gRPC channels to gRPC servers.

4.19.2 bbf-grpc-server

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on network functions as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module defines a set of common groupings to support the gRPC server interface to listen for and maintain gRPC channels from gRPC clients.

4.19.3 bbf-kafka-agent

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on network functions as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module defines a set of common groupings to support Kafka agents that operate as Kafka clients to read, write, and process streams of events via Kafka brokers (servers) to communicate between individual network function.

4.19.4 bbf-network-function

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on network functions as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module defines a set of common groupings to support the topology of network functions within a network. For example, use-cases such as Disaggregated OLT or Virtual OMCI, which use cloud-based remote network functions via network connectivity, use this module as the basis for adding additional clients and servers for that network connectivity. The specific endpoints, protocols, and transport features are described within their respective documents and modules.

4.19.5 bbf-network-function-client

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on network functions as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module defines a set of common groupings to support client endpoints of network connections between network functions.

4.19.6 bbf-network-function-server

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on network functions as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module defines a set of common groupings to support server endpoints of network connections between network functions.

4.19.7 bbf-network-function-types

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on network functions as applicable to access network equipment. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module defines a set of common types related to network functions and network topologies.

4.20 IPFIX

This module supports the management of Internet Protocol Flow Information Export (IPFIX) to fulfill the Broadband Forum requirements for access network equipment that streams data/information as defined in IETF RFC 7011 [97] and RFC 7012 [98].

4.20.1 bbf-ipfix-export

This module contains a collection of YANG definitions to manage exporting data using the Internet Protocol Flow Information Export (IPFIX) protocol [97].

This data model is designed for the Network Management Datastore Architecture defined in RFC 8342 [107].

4.21 Voice

These modules are used for the management of voice services for access network equipment.

4.21.1 bbf-pots-alarm-types.yang

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on voice service management. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module contains a set of alarm definitions for Plain Old Telephone Service (POTS).

4.21.2 bbf-pots.yang

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on voice service management. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

This data model is designed for the Network Management Datastore Architecture defined in RFC 8342 [107].

Specifically, this module augments the IETF interfaces YANG data model defined in RFC 7223 [100] with data nodes to support the management of physical PPlain Old Telephone Service (POTS) interfaces defined in ITU-T G.988 [126].

4.21.3 bbf-ringing-patterns-common.yang

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on voice service management. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module describes reusable groupings containing configuration and operational data nodes used for managing ringing patterns.

4.21.4 bbf-ringing-patterns.yang

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on voice service management. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module supports a stand-alone definition of ringing patterns.

4.21.5 bbf-tone-patterns-common.yang

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on voice service management. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module describes reusable groupings containing configuration and operational data nodes used for managing tone patterns.

4.21.6 bbf-tone-patterns.yang

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on voice service management. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module supports a stand-alone definition of tone patterns.

4.21.7 bbf-voip-media-alarm-types.yang

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on voice service management. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module contains a set of alarm definitions for Voice over IP (VoIP) media.

4.21.8 bbf-voip-media.yang

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on voice service management. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

This data model is designed for the Network Management Datastore Architecture defined in RFC 8342 [107].

Specifically, this module adds support for the management of Voice over IP (VoIP) media and media capabilities defined in ITU-T G.988 [126].

4.21.9 bbf-voip-sip-alarm-types.yang

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on voice service management. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module contains a set of alarm definitions for VoIP Session Initiation Protocol(SIP).

4.21.10 bbf-voip-sip.yang

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on voice service management. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

This data model is designed for the Network Management Datastore Architecture defined in RFC 8342 [107].

Specifically, this module adds support for the management of Voice over IP (VoIP) Session Initiation Protocol(SIP) and VoIP SIP capabilities defined in ITU-T G.988 [126].

4.21.11 bbf-voip-types.yang

This module contains a collection of YANG definitions for supporting the Broadband Forum requirements on voice service management. As such, this module is specific to access network equipment (e.g., BBF-specified Access Nodes and FTTdp DPUs).

Specifically, this module defines some VoIP specific types to align with types defined in ITU-T G.988 [126] and some common types used by the VoIP modules.

5 Documentation

This section is intentionally left blank.

6 Dependencies on related YANG modules and Standards

TR-383 is based on YANG 1.1 (RFC 7950 [104]).

The following YANG modules are used by TR-383:

- iana-hardware@2018-03-13 [108]
- iana-if-type@2021-06-21 [101]
- ieee802-dot1q-cfm@2020-06-04 [27]
- ieee802-dot1q-cfm-types@2020-06-04 [27]
- ieee802-dot1q-types@2020-06-04 [27]
- ieee802-types@2020-06-04 [27]
- ietf-alarms@2019-09-11 [113]
- ietf-alarms-x733@2019-09-11 [113]
- ietf-hardware@2018-03-13 [108]
- ietf-inet-types@2013-07-15 [96]
- ietf-interfaces@2014-05-08 [100]
- ietf-netconf-acm@2018-02-14 [106]
- ietf-packet-fields@2019-03-04 [111]
- ietf-system@2014-08-06 [103]
- ietf-yang-schema-mount@2019-01-14 [112]
- ietf-yang-types@2013-07-15 [96]
- itut-eth-oam@2023-01-13 [132]
- iana-tls-cipher-suite-algs@2024-03-16 [117]
- ietf-tls-client@2024-10-10 [117]
- ietf-crypto-types@2024-10-10 [114]
- ietf-truststore@2024-10-10 [115]
- ietf-keystore@2024-10-10 [116]
- ietf-tls-common@2024-10-10 [117]
- ietf-ip@2014-06-16 [102]
- iana-crypt-hash@2014-08-06 [103]

7 Layer 2 Forwarding Data Model

The intent of this section is to provide some general information regarding the usage of the layer 2 forwarding data model. It is not possible to describe every possible application that would use the model, but rather, it provides the theory behind the model and illustrates some general use cases.

7.1 Sub-interfaces

Before traffic can be forwarded, it must first be classified to determine what to forward, where to forward, and how to manipulate the packet if so desired. The concept of a VLAN sub-interface realized in YANG as an interface of the type 'vlan-sub-interface' has been introduced for providing an interface which can be used as the source or destination interface of a forwarding decision. Each VLAN sub-interface classifies traffic from a particular lower-layer interface into a forwarder. This classification consists of a set of rules specified using match criteria on to packet fields (e.g., VLAN-ID, p-bit). The lower-layer interface can be either a non-aggregated physical or logical interface (e.g., Ethernet), an aggregation of physical or logical interfaces (e.g., LAG), or can be another VLAN sub-interface.

A VLAN sub-interface is created each time a new forwarding context is required (e.g., 1:1 VLAN). Each VLAN sub-interface can then have multiple rules associated with it if different classification results in the same forwarding decision. For example, one rule can catch frames tagged with a particular VLAN-ID, a second rule can catch untagged frames, and a third rule can catch priority-tagged frames. The second and third rules in this example cover the concept of a port default VLAN.

As stated above, multiple VLAN sub-interfaces can refer to the same lower-layer interface in order to provide multiple traffic classifications based on different, but potentially overlapping, match criteria. In order to provide deterministic classification, each rule can be given a priority. The scope of the priority is over all rules defined within all VLAN sub-interfaces referring to the same lower-layer interface. A packet ingressing the lower-layer interface would then be compared to each rule, starting with the highest priority rule and proceeding to the lowest priority rule. If a match occurs, the packet is processed accordingly. If no match occurs, the packet is dropped.

The figure below shows how two VLAN sub-interfaces are associated with the same physical interface, classifying traffic for two different forwarding decisions.

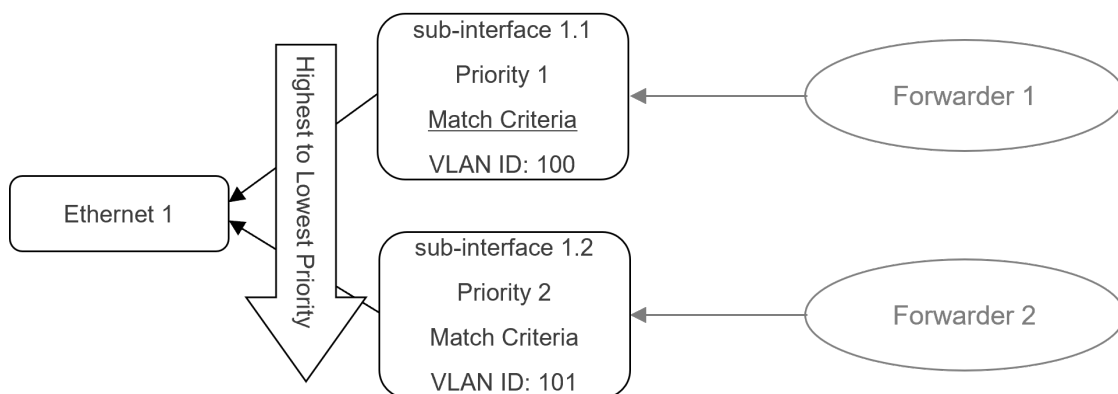


Figure 2 – Sub-Interface Example

In addition to the classification of traffic, the sub-interface also contains rules for any ingress or egress actions to take on each matched packet. These actions include pushing or popping tags, rewrite of p-bits, or rewrite of Drop Eligible Indication (DEI) bits.

7.1.1 Interface Usage

Replaced by [Section 7.3](#).

7.2 Forwarders

Once traffic has been classified and possibly manipulated, it needs to be forwarded appropriately to another sub-interface. A forwarder is used to determine how traffic is routed between two or more forwarder ports, each of which is associated with a sub-interface. This forwarder can be used to handle 1:1 VLAN, N:1 VLAN and N:M VLAN applications.

7.2.1 Forwarder Ports and Port Groups

Each forwarder port is associated with a sub-interface whose underlying interface is either a user port, a network port, or a subtended node port. Forwarder ports with similar forwarding characteristics can be placed into forwarding groups and referenced collectively when configuring the forwarder.

Figure 3 shows the relationships between a forwarder, its forwarder ports, and the referenced sub-interfaces and their lower-layer interfaces.

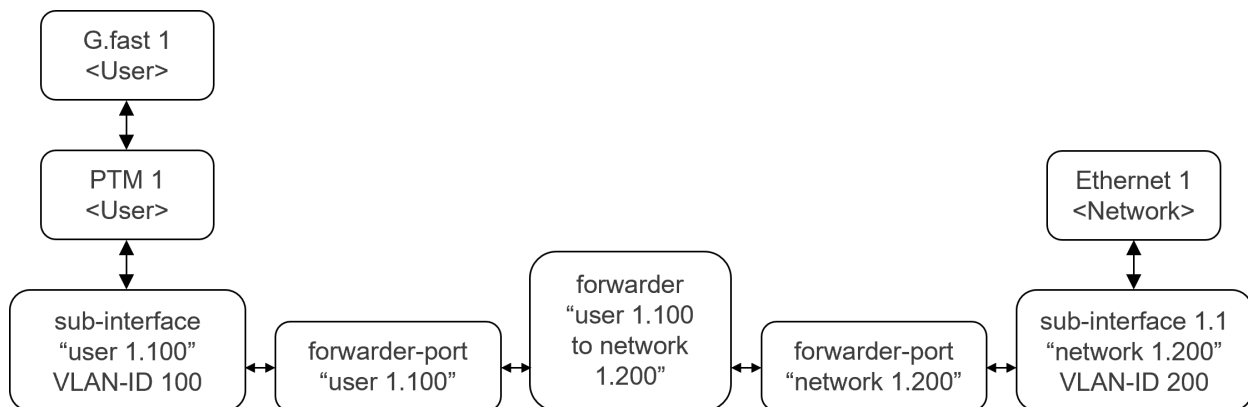


Figure 3 – Forwarder Ports

In the simplest use case of a 1:1 VLAN, this is all the forwarder needs to forward traffic between two sub-interfaces. The sub-interfaces determine which packets will be forwarded and how they will be manipulated. The forwarder just provides the means to associate the sub-interfaces.

7.2.2 Split Horizon Profiles

Once the interface usage is configured, a split horizon profile can be created and applied per forwarder to configure how traffic is forwarded between the various types of interfaces. Each profile specifies the usage of the ingress interface then lists the usages for which egress of the packet is not allowed from the ingress interface. For example, a profile could specify that for an ingress interface that is a user port, it is not allowed to send traffic to interfaces that are also user ports.

7.2.3 MAC Learning

In addition to the usage of an interface, the source and destination MAC addresses are key to making correct forwarding decisions for N:1 and N:M VLAN forwarding. Each forwarder contains configuration that determines how and if MAC source addresses are learned. It can also prevent traffic from being forwarded if it contains a certain MAC source address.

Once a MAC source address is learned, it is installed in the forwarding database for a given forwarder port. When a packet egresses a forwarder port, its MAC destination address is compared to the addresses in each of the other forwarder port's forwarding database to determine where the packet should be forwarded.

7.2.4 Flooding

In the case that the destination MAC address is not found in any forwarding database, it may be desired to flood the packet to all appropriate forwarder ports. To determine when and how this flooding occurs, a flooding policy profile can be created and associated with each forwarder. Each policy can be specified for a particular interface usage (e.g., user port) and/or a specific destination MAC address. It then assigns an appropriate action of either discarding the packet or flooding it to all interfaces of specified usage(s). For example, a forwarder may be configured to flood all packets with an unknown MAC address coming from a network port to all user ports.

7.3 Interface Usage

For the case of N:1 or N:M VLAN forwarding, the role each interface plays in the network is important to determine how traffic flow is managed. For example, in the context of an Access Node, traffic ingressing a user port should not normally be forwarded to another user port. Certain mechanisms, as discussed above, will be used to enforce this restriction. First, however, the way in which an interface is used must be explicitly known. For this, the interface usage must be configured either by the user or by the system if the usage is already known. The 3 types of interface usage are:

- user port: The interface connects an Access Node to a user.
- network port: The interface connects an Access Node to a network.
- subtended-node port: The interface connects an Access Node to another Access Node.

Interface usage is applicable only to interface types derived from 'bbfift:ethernet-like' and for the following types and those derived from these types:

- 'ianaift:ethernetCsmacd'
- 'ianaift:ieee8023adLag'
- 'ianaift:ptm'
- 'bbfift:vlan-sub-interface'
- 'bbfift:l2-termination'.

7.4 Layer 2 Termination Interfaces

In some situations, Layer 2 VLAN traffic has to be terminated within a device, and the device needs to manipulate the payload of the received frames, such as when IP packets are needed in the context of the in-band management NETCONF/YANG traffic, ANCP sessions, etc.

In the simple case, this can be realized at Layer 2 by creating a VLAN sub-interface on the network port and creating an IP interface on top of the VLAN sub-interface.

However, in other situations, not all frames received via the VLAN sub-interface are to be processed by the device. For example, when the VLAN is a shared resource to manage a hub node and all its subtending nodes. In this case, the destination MAC address (and IP address) of some of the frames will have the address of the hub node, while other frames will have the address of the subtending node. Consequently, received frames first need to be subject to a Layer 2 forwarding decision, whereas only frames addressed to the hub node are to be processed by the hub node at the IP layer.

Terminating Layer 2 after a forwarding decision is therefore modeled as a 'Layer 2 termination interface'.

A Layer 2 termination interface differs from a VLAN sub-interface as it does not have a lower-layer interface and hence also has no match criteria to classify frames received from this lower-layer interface. It also has other characteristics that are used in the context of the common forwarding decisions, e.g., it has an interface usage that determines how forwarding decisions such as split-horizon-profile, flooding, etc., have to be taken inside the forwarder, i.e., it will have a QoS profile, ingress/egress VLAN tag manipulation, etc.

8 Ethernet-like Interfaces

There are several instances in the Common YANG modules where the interface list from `ietf-interfaces` [100] is augmented with a constraint on the types of interfaces to which the augmented nodes apply. For example, the type of an interface can be limited to the type of interfaces that transport Ethernet frames, as shown below.

```
augment '/if:interfaces/if:interface' {
  when
    "derived-from-or-self(if:type, 'ianaift:ethernetCsmacd') or
     derived-from-or-self(if:type, 'ianaift:ieee8023adLag') or
     derived-from-or-self(if:type, 'ianaift:ptm') or
     derived-from-or-self(if:type, 'bbfift:vlan-sub-interface')" {
    description
      "Interfaces that can have QoS policy profiles assigned.";
  }
}
```

The augmentation to add a QoS policy reference to an interface is constrained to interfaces which are of one of four types or derived from those types. See RFC 7950 [104] for the full definition and usage of the `derived-from-or-self()` function.

Similarly, there are nodes which are references to an interface whose type is also constrained to those that transport Ethernet frames.

```
leaf interface {
  type if:interface-ref;
  must
    "derived-from-or-self(
      /if:interfaces/if:interface[if:name = current()]
      /if:type, 'ianaift:ethernetCsmacd')
     or
     derived-from-or-self(
      /if:interfaces/if:interface[if:name = current()]
      /if:type, 'ianaift:ieee8023adLag')
     or
     derived-from-or-self(
      /if:interfaces/if:interface[if:name = current()]
      /if:type, 'ianaift:ptm')
     or
     derived-from-or-self(
      /if:interfaces/if:interface[if:name = current()]
      /if:type, 'bbfift:sub-interface') ";
  mandatory true;
  description
    "References the lower-layer interface.";
}
```

In this example, the reference to the interface must be one of four types or derived from those types.

While this approach works well, it does not lend itself to extensibility when new interface types are defined either by the Broadband Forum, a vendor, an operator, or some other organization that is defining YANG data models. In order for these augments and must statements to be applicable to those interface types, either the new type or types need to be derived from one of these existing types or the new ones have to be added to the modeled constraints. This presents a challenge of keeping these models aligned and may not even be possible depending on the source of the newly defined interface type.

One solution that has been introduced is the creation of an abstract Ethernet type from which new interface types can be defined.

```
identity ethernet-like {
  base bbf-interface-type;
  description
    "An abstract identity defining a class of interfaces which
    represents a logical interface transporting Ethernet frames,
    i.e. frames with a destination and source MAC address, an
    Ethernet type or length field, and a payload. This
    'interface type' is intended only to be used to define
    constraints against a class of interfaces, each of which have
    their 'type' derived from this identity (as well as potentially
    others). At no time should this identity be used as the 'type'
    for an interface.";
}
```

This abstract type is added to the constraints.

Updated augment example:

```
augment '/if:interfaces/if:interface' {
  when
    "derived-from-or-self(if:type, 'ianaift:ethernetCsmacd') or
    derived-from-or-self(if:type, 'ianaift:ieee8023adLag') or
    derived-from-or-self(if:type, 'ianaift:ptm') or
    derived-from-or-self(if:type, 'bbfift:vlan-sub-interface') or
    derived-from(if:type, 'bbfift:ethernet-like')" {
    description
      "Interfaces that can have QoS policy profiles assigned.";
  }
}
```

Updated must statement example:

```
leaf interface {
  type if:interface-ref;
  must
    "derived-from-or-self(
      /if:interfaces/if:interface[if:name = current()]
      /if:type, 'ianaift:ethernetCsmacd')
    or
    derived-from-or-self(
      /if:interfaces/if:interface[if:name = current()]
      /if:type, 'ianaift:ieee8023adLag')
    or
    derived-from-or-self(
      /if:interfaces/if:interface[if:name = current()]
      /if:type, 'ianaift:ptm')
    or
    derived-from-or-self(
      /if:interfaces/if:interface[if:name = current()]
      /if:type, 'bbfift:sub-interface')
    or
    derived-from(
      /if:interfaces/if:interface[if:name = current()]
      /if:type, 'bbfift:ethernet-like') ";
  mandatory true;
  description
```

```
    "References the lower-layer interface.";
}
```

The use of 'derived-from' [104] stems from the identity's definition which states it is an abstract identity which is not to be used as an actual interface type.

The method of using this abstract type is to add it as a base identity [104] to any Ethernet type definition that satisfies the definition of 'ethernet-like'. For example,

```
identity new-ethernet-type {
  base bbft:ethernet-like;
  description
    "A new Ethernet type.";
}
```

By utilizing the abstract interface type, the Common YANG modules that define these constraints no longer have to be updated when a new Ethernet interface type is created.

9 Alarms

The intent of this section is to provide some general information regarding alarm management as applicable to BBF-specified access network equipment.

BBF alarm management is based on the IETF "YANG Data Model for Alarm Management" (RFC 8632) [113], which defines a standardized alarm interface for network devices that can be easily integrated into management applications.

The design of this data model addresses usability requirements, such as those discussed in 3GPP TR 32.859 [12] for example, improving the management of alarm overload through alarm shelving.

9.1 Alarms and Alarm Types

RFC 8632 [113] defines an alarm is an undesirable state of a resource and uniquely identifies an instance of an alarm by

- a fine-grained identification of the alarming resource, such as a specific interface
- an alarm type, which defines a possible undesirable state of the resource, such as 'loss of signal'.

where alarm type is defined by

- an alarm type identifier ('alarm-type-id')
- an alarm type qualifier ('alarm-type-qualifier').

An alarm type identifier is modeled as a YANG identity, is defined at design time, and can be abstract or concrete. Abstract alarms are a means of categorizing alarms and may also be used by a client for alarm filtering purposes.

An alarm type qualifier is a string that may be used if the alarm type identifier alone cannot uniquely identify the alarm type, for example, for alarms not known at design time.

As described in Section 3.2 of RFC 8632 [113], abstract alarms are generally not used for alarms. However, if an alarm is instrumented that was not known at design time, i.e., for which no concrete alarm type identifier has been defined in the YANG model, an abstract alarm type identifier qualified with an alarm type qualifier would be used. This practice, however, should be generally avoided to ensure that all possible alarms are known at design time.

9.1.1 Common Alarm Types

A YANG identity hierarchy of common abstract alarm type identities is defined to categorize alarm types defined by BBF applications based on the requirements of alarm reporting parameters associated with an alarm type as discussed in ITU-T X.733 [139], ITU-T X.736 [140], and 3GPP TS 32.111-2 [13]. These common abstract alarm type identities implicitly specify the alarm information (or alarm payload) defined in the IETF modules ietf-alarms and ietf-alarms-x733 that is applicable to alarm types based on these abstract alarm type identities.

The following abstract alarm type identifier identities are defined based on the identity alarm-type-id defined in ietf-alarms:

```
+--ietf-alarms:alarm-type-id
  +--bbf-alarm-types:bbf-alarm-type-id
    +--bbf-alarm-types:bbf-threshold-crossing-alarm-type-id
    +--bbf-alarm-types:bbf-security-alarm-type-id
```

The common abstract alarm types shown above do not define a fixed hierarchy of alarm types based on Event Types defined in ITU-T X.733 [139] and ITU-T X.736 [140], because the RFC 8632 [113] supports manageable Event Types for individual alarm types in ietf-alarms-x733. This allows a network operator to map the default vendor-specified Event Types associated with specific alarm types according to the operator's own requirements.

The alarm information to be associated with alarm type identities based on 'bbf-threshold-crossing-alarm-type-id' and 'bbf-security-alarm-type-id' is supported in the module ietf-alarms-x733.

Other abstract alarm type sub-categories may be defined to further categorize alarm types by BBF applications, but these sub-categories will be based directly or indirectly on one or more of the BBF alarm types listed above. Alarm types based on these sub-categories inherit the alarm information applicable to the alarm type on which they are based, but may refine this for abstract application-specific alarm types accordingly. Alarm type identities based on more than one of these abstract alarm type identities inherit the alarm information specification from each of these abstract identities.

Alarm information will be associated with each instance of an alarm and is implemented in ietf-alarms and ietf-alarms-x733 as data nodes within the alarm list and shelved-alarm list and carried within the 'alarm-notification'. [Table 1](#) below indicates whether this alarm information is mandatory (M), optional (O), or is not present (NP) for a concrete alarm type identifier when it is based on one or more of these abstract alarm type identifiers. If an abstract or concrete alarm type identifier is based on more than one abstract alarm type identifier, then the alarm information associated with that alarm type identifier will be a combination of the alarm information associated with each abstract alarm type identifier, where 'mandatory' has precedence over 'optional' has precedence over 'not present'.

Table 1 – Abstract BBF Alarm Types and Associated Alarm Information

Data node	Module	bbf-alarm-type-id	bbf-threshold-crossing-alarm-type-id	bbf-security-alarm-type-id
resource	ietf-alarms	M	M	M
alarm-type-id	ietf-alarms	M	M	M
alarm-type-qualifier	ietf-alarms	M (NOTE 1)	M (NOTE 1)	M (NOTE 1)
alt-resource	ietf-alarms	O	O	O
related-alarm	ietf-alarms	O	O	O
impacted-resource	ietf-alarms	O	O	O
root-cause-resource	ietf-alarms	O	O	O
time-created	ietf-alarms	M	M	M
is-cleared	ietf-alarms	M	M	M
last-raised	ietf-alarms	M	M	M
last-changed	ietf-alarms	M	M	M
perceived-severity	ietf-alarms	M	M	M
alarm-text	ietf-alarms	M	M	M
event-type	ietf-alarms-x733	O	O	O

Data node	Module	bbf-alarm-type-id	bbf-threshold-crossing-alarm-type-id	bbf-security-alarm-type-id
probable-cause	ietf-alarms-x733	O	O	O
monitored-attributes	ietf-alarms-x733	O	O	O
proposed-repair-actions	ietf-alarms-x733	O	O	O
trend-indication	ietf-alarms-x733	O	O	O
backedup-status	ietf-alarms-x733	O	O	O
backup-object	ietf-alarms-x733	O	O	O
additional-information	ietf-alarms-x733	O	O	O
threshold-information	ietf-alarms-x733	NP	M	NP
security-alarm-detector	ietf-alarms-x733	NP	NP	M
service-user	ietf-alarms-x733	NP	NP	M
service-provider	ietf-alarms-x733	NP	NP	M

NOTES:

1. The data node 'alarm-type-qualifier' is mandatory because it is a key to the list 'alarm', but in most cases it will not be needed, in which case it will be set to an empty string.

9.1.2 Application-specific Alarm Types

Concrete alarm types are to be defined by BBF applications and must either be based on at least one of the abstract alarm types defined in the module, bbf-alarm-types, or be based on an abstract alarm type defined by a BBF application and derived from at least one of those types.

As an example of an alarm hierarchy using abstract and concrete alarms is the following hierarchy of alarms defined in TR-355 [9]:

```

+--al:alarm-type-id                                (abstract)
  +--bbf-alt:bbf-alarm-type-id                      (abstract)
    +--bbf-fast-al:fast                             (abstract)
      +--bbf-fast-al:fast-ftu-o-failures             (abstract)
        | +--bbf-fast-al:fast-ftu-o-line-initialization (concrete)
        | +--bbf-fast-al:fast-ftu-o-loss-of-signal    (concrete)
        | +--bbf-fast-al:fast-ftu-o-loss-of-rmc      (concrete)
        | +--bbf-fast-al:fast-ftu-o-loss-of-margin   (concrete)
        | +--bbf-fast-al:fast-ftu-o-loss-of-power    (concrete)
      +--bbf-fast-al:fast-ftu-r-failures             (abstract)
        +--bbf-fast-al:fast-ftu-r-loss-of-signal    (concrete)
        +--bbf-fast-al:fast-ftu-r-loss-of-rmc      (concrete)
        +--bbf-fast-al:fast-ftu-r-loss-of-margin   (concrete)
        +--bbf-fast-al:fast-ftu-r-loss-of-power    (concrete)

```

where 'al:alarm-type-id' is the base alarm type identifier for all alarms managed by ietf-alarms and is located in the module ietf-alarms (RFC 8632) [113] 'bbf-alt:bbf-alarm-type-id' is the base alarm type identifier for all

BBF-defined alarms and is located in the module `bbf-alarm-types`. `'bbf-fast-al:fast'` is the base alarm type identifier for all FAST line alarms and is defined in module `bbf-fast-alarms`. This example also shows how application-specific alarms can be further categorized, i.e., FAST alarms are further categorized into alarms for local and remote FAST Transceiver Unit (FTU) failures.

10 Access Node Control Protocol

The intent of this section is to provide some general information regarding the use of the ANCP YANG data model to manage the Access Node Control Protocol (ANCP) [2] [88] on Access Nodes.

The ANCP YANG data model published as part of this revision of the Technical Report supports only the topology discovery capability defined in RFC 6320 [88].

10.1 Partitions, Sessions, and Adjacencies

A partition collects a set of access lines that are to be managed together by one or more Network Access Server (NAS). In RFC 6320 [88] an Access Node (AN) may or may not support partitions. In the ANCP YANG model access lines on an AN that are to take part in ANCP must always be explicitly assigned to a partition. In the case where the AN does not support partitions, a single 'global' partition will need to be configured in the model.

If the single partition with partition-id = 'global' is configured, the PType and Partition ID in the ANCP Adjacency Message must be set to 0 (no partition) and 0 respectively.

A given partition may only collect access lines together of the same technology, e.g., FastDSL.

Configuration and operational state that depends on a specific technology, such as FastDSL, are augmented into the main ANCP YANG module bbf-ancp by technology-specific ANCP YANG modules.

An adjacency between the AN and a NAS for a given partition is managed in the model through the configuration of a session. The session manages the connection to the remote NAS, including which line attributes are to be reported in the Port Up and Port Down event messages sent by the session to the NAS.

The relationship between partitions, sessions and access lines, the latter of which is represented by an interface, is shown below.

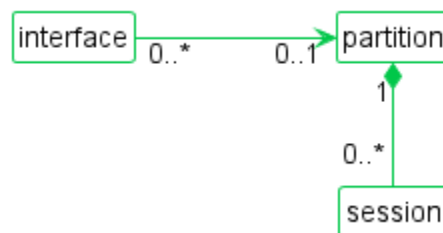


Figure 4 – Relationships Between Partitions, Sessions, and Interfaces

The workflow to create an adjacency is as follows:

1. create a partition;
2. assign access lines to that partition;
3. create one or more sessions to manage an adjacency from the partition to a remote NAS.

10.1.1 Create a Partition

Before ANCP can be used on an AN at least one partition must be created. When creating the partition, the technology of the access lines to be assigned to the partition must be specified, but access lines of different technologies cannot be mixed. An AN may support one or more technologies and this can be determined through the capabilities advertised by the AN.

10.1.2 Assigning Access Lines to a Partition

Access lines can be assigned to a partition through the interfaces in ietf-interfaces that represent the access lines. The assignment is made by referencing the partition from an interface. Configuring this reference automatically enables ANCP functionality for that access line.

10.1.3 Create a Session

For each adjacency between an AN and a NAS for a given partition, a session must be created for that partition.

The Transmission Control Protocol (TCP) connection to the NAS for which adjacency is to be attained is configured in the container 'network-access-server' in the list 'session'. This container also reports the identification of the remote NAS once adjacency has been achieved.

10.2 Topology Discovery

Topology discovery is enabled for the session by configuring at least one line attribute for the Port Up messages specific to the technology configured for the given partition. Line attributes are configured in the technology-specific containers found as child nodes to the container 'port-up' within the container 'topology-discovery' of a session. Similarly, line attributes to be included in Port Down messages are configured in the technology-specific containers found as child nodes to the container 'port-down' within the container 'topology-discovery' of a session.

10.3 Access Line Identification

The identification of an access lines on the AN must be configured and is defined through a combination of logical port information on the user side as well as on the NAS side of the AN. RFC 6320 [88] defines four ANCP TLVs for access line identification:

- Access-Loop-Circuit-ID
- Access-Loop-Remote-ID
- Access-Aggregation-Circuit-ID-Binary
- Access-Aggregation-Circuit-ID-ASCII.

Access line identification is also required to be supported for DHCP and PPPoE as specified in TR-101 Issue 2 Section 3.9 [1]. To ensure a consistent identification of access lines across ANCP, DHCP and PPPoE, common access line identification parameters can be configured within a subscriber profile, which is assigned to the VLAN sub-interface associated with the access line.

The specific access line identification TLVs to be sent in ANCP messages for a given session is configured in the leaf-list 'line-identification' within the container 'access-line-identification'.

10.3.1 Access-Loop-Circuit-ID

The value inserted into this TLV must be the value configured for the leaf 'circuit-id' within the subscriber-profile that is referenced from the VLAN sub-interface associated with the access line. If no such subscriber-profile has been configured, then a TLV must be generated according to the syntax defined in the leaf 'access-loop-circuit-id' within the container 'access-line-identification' of the session. If this leaf is also not defined, then an empty TLV must be inserted, i.e., a TLV with Length = 0.

10.3.2 Access-Loop-Remote-ID

The value inserted into this TLV must be the value configured for the leaf 'remote-id' within the subscriber-profile that is referenced from the VLAN sub-interface associated with the access line. If no such subscriber-profile has been configured, then an empty TLV must be inserted, i.e., a TLV with Length = 0.

10.3.3 Access-Aggregation-Circuit-ID-Binary and Access-Aggregation-Circuit-ID-ASCII

Access-Aggregation-Circuit-ID-Binary identifies or partially identifies a specific access line by means of the VLAN IDs of the inner and outer VLAN tags of the data frames coming from that access line on the NAS side of the AN. The format of Access-Aggregation-Circuit-ID-Binary is specified in RFC 6320 [88].

Access-Aggregation-Circuit-ID-ASCII is an ASCII equivalent of Access-Aggregation-Circuit-ID-Binary TLV, the format of which is explicitly configured in 'access-aggregation-circuit-id-ascii' within the container 'access-line-identification' of a session. As per RFC 6320 [88], it shall contain VLAN IDs, e.g., 'S-VID:C-VID', but may contain any characters and variables in a format as specified by TR-101 Issue 2 Section 3.9.3 [1].

If frames received on the subscriber interface are forwarded to multiple VLAN sub-interfaces, then the AN would need to know how to select which VLAN sub-interface to use to derive the VLAN-IDs for Access-Aggregation-Circuit-ID-Binary and Access-Aggregation-Circuit-ID-ASCII. This information is configured in through the choice 'access-aggregation-circuit-id' within the container 'anccp' on the interface representing the access line. The choice has two cases

- 'auto-derived' the VLAN IDs are determined from the VLAN sub-interface that classifies ingress frames with the lowest VLAN ID value, combined with the related forwarding and network-side VLAN sub-interface configuration.
- 'explicit' up to two VLAN IDs can be explicitly configured.

10.3.4 Additional Formatting

If the Access-Loop-Circuit-ID or Access-Aggregation-Circuit-ID-ASCII use the variables for a slot, a port or other numbered variable, the configuration 'start-numbering-from-zero' controls whether the number begins with 0 or 1 and 'use-leading-zero' whether or not leading zeroes are to be used when representing the numbers.

10.3.5 Supporting FastDSL Bonding

For a bonding group that bonds multiple access lines, a primary line for the bonding group must be selected, which will be used to generate the Access-Loop-Circuit-ID. This is configured in the leaf 'primary-line' of a bonding group interface defined in module bbf-gbond, first available in TR-355 Amendment 3 [9].

10.4 Controlling Port Messages

The ANCP model also supports additional configurations to control how and when Port Up and Port Down event messages are sent.

10.4.1 Threshold-based Reporting

For some specific line attributes, TR-301 [7] requires that if the measurement on the port changes by more than a configurable threshold, the port state must be reported to the PMA. In the ANCP YANG data model

shift-up and shift-down thresholds can be configured for specific line attributes per partition, applying to all sessions of that partition.

The configuration is made in the technology-specific containers 'vdsl' and 'fast' containers within the container 'port-message-control' of a partition.

10.4.2 Delaying the Initial Port Up Message

Unstable connections that go in and out of sync and line characteristics that are unstable during the synchronization process can cause a flood of Port Up and Port Down messages.

To limit unnecessary Port Up and Port Down event messages during the synchronization process, it is possible to configure an 'initial-port-up-delay' which requires that the line be synchronized for a given period, before the first Port Up message is sent following synchronization of the line.

The configuration is made in the technology-specific containers 'vdsl' and 'fast' within the container 'port-message-control' of a partition.

10.4.3 Dampening Mechanism

Seamless Rate Adaptation (SRA) and Fast Rate Adaptation (FRA) of FastDSL access lines may result in rapid and continuous changes in the data rates. RFC 6320 [88] recommends that a dampening mechanism be supported to limit the rate at which state changes of access lines are reported to the NAS. This is supported in the ANCP YANG data model through the configuration of a 'port-up-port-down-withholding-interval' within the container 'port-message-control' of a partition.

The withholding interval applies to each access line independently and defines an interval which begins when a Port Up message is sent for that access line. During this interval no further Port Up message will be sent to the NAS for that given access line. If, at the end of the withholding interval, there has been a change in line state of the given line to that when the last Port Up message was sent, a Port Up message is sent with the new state (with the withholding interval for that line applying again).

10.5 Statistics

The ANCP YANG data model supports the reporting of statistics per session for adjacency messages sent and received and statistics for topology discovery messages sent per access line per session.

10.6 Alarms and the Operational State of a Session

If a session encounters an issue and is unable to operate correctly as intended by the configuration, the session will raise an alarm to the Network Management System (NMS). The YANG data model supports the following alarms:

- 'adjacency-failure'
- 'capability-negotiation-failure'
- 'message-type-not-supported'
- 'nas-not-reachable'.

If an alarm is raised, the session may cease operation and come to a halt, and as a result may need to be reset by the NMS once the issue has been resolved.

Figure 5 below summarizes the behavior of an ANCP session from the viewpoint of the operation state defined in the YANG data model, illustrating how alarms impact states and how or when these alarms may be cleared.

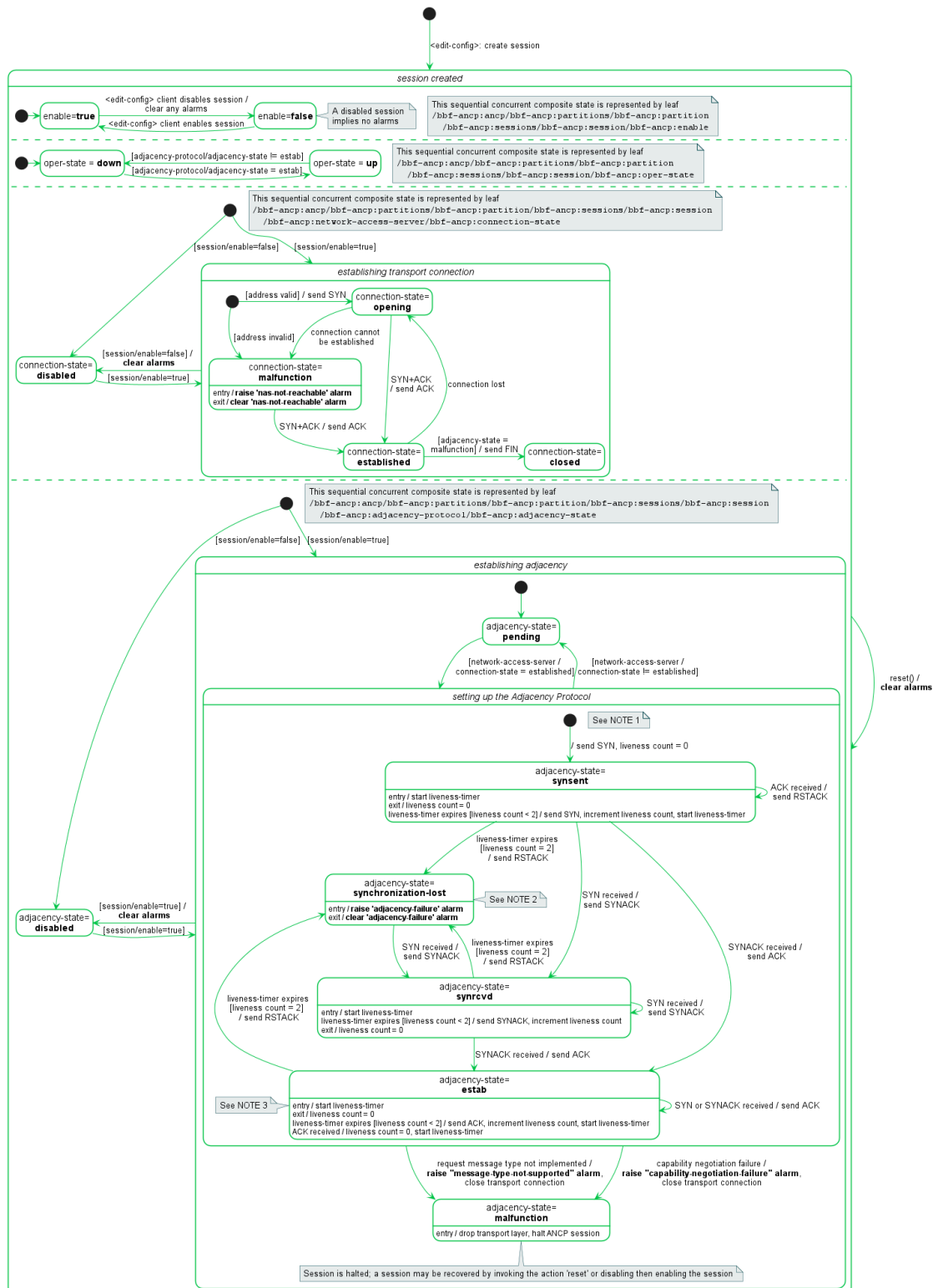


Figure 5 – Statechart of an ANCP Session (Informational Only)

- NOTE 1: Only significant state transitions are shown that are key to understanding the behavior of the YANG data model.
- NOTE 2: This state is entered when a session declares a loss of synchronization as defined in Section 3.5.2.7 of RFC 6320 [88]. RFC 6320 [88] is not specific as to how a session re-establishes synchronization; how this is achieved does not need specific representation in the YANG data model.
- NOTE 3: Section 3.5.2.2 of RFC 6320 [88] specifies that no more than one ACK should be sent within any time period of length defined by the liveness timer.

11 Software Management

The intent of this section is to provide some general information regarding the use of the Software Management YANG data model. Although the main requirements driving the data model were to enable the PMA to manage software images on a DPU according to requirement R-214 of TR-301 [7] and to support the Software Image Management requirements of ITU-T G.997.2 Annex S [129] and ITU-T G.988 clause 9.1.4 and Appendix I [126], the data model has nevertheless been intentionally designed to generally support the management of software on any hardware component supported by a device. It is, therefore, not possible here to describe every possible application of the model, but rather the information below provides the theory behind the model and illustrates some general use cases.

The management of software as applicable to access network equipment is based on the IETF "A YANG Data Model for Hardware Management" (RFC 8348) [108].

11.1 Components, Software and Revisions

The Software Management YANG data model is based on the concept that a hardware component of a device may support one or more pieces of manageable software that may be replaced during the lifetime of the device, whereby each software is defined by a set of revisions of that software that is present on the component. Since the model does not make any assumptions about what the 'software' is, the 'software' could be an executable code image or other operating information, such as a script or even, for example, a text-based profile. The format of software files and how they are processed, such as extracting software image from an archive, is assumed to be an integral part of the download activity and as such is an implementation aspect not within the scope of this Technical Report.

Consequently, the Software Management YANG data model augments a 'component' within the module `ietf-hardware` [108] with a container 'software' containing a list 'software'. Each entry in the list is uniquely identified by a name (string), whereby this name should also be representative of its purpose and identify its association with the hardware component, e.g., "firmware". The maximum number of revisions supported for a specific software would be device- and software-specific.

Each revision is defined by the following metadata:

id	A unique identifier for the software revision.
alias	If supported by the specific software, an optional unique name for the software revision that may be assigned by the client when the software revision is downloaded.
version	A string that identifies the published version of the revision.
is-committed	If supported by the specific software, a Boolean that if 'true' indicates that the revision will be loaded and made active upon a reboot of the device.
is-active	If supported by the specific software, a Boolean that if 'true' indicates that the revision is currently loaded and active.
is-valid	If supported by the specific software, a Boolean that if 'true' indicates that the revision's contents have been verified to be a valid revision that may be activated and committed.
product-code	An optional string to indicate vendor-specific product code information.
hash	An optional hash of the revision

At any given time, at most one revision may be active and at most one revision may be committed.

Figure 6 illustrates the relationship between hardware components and software within the YANG data model.

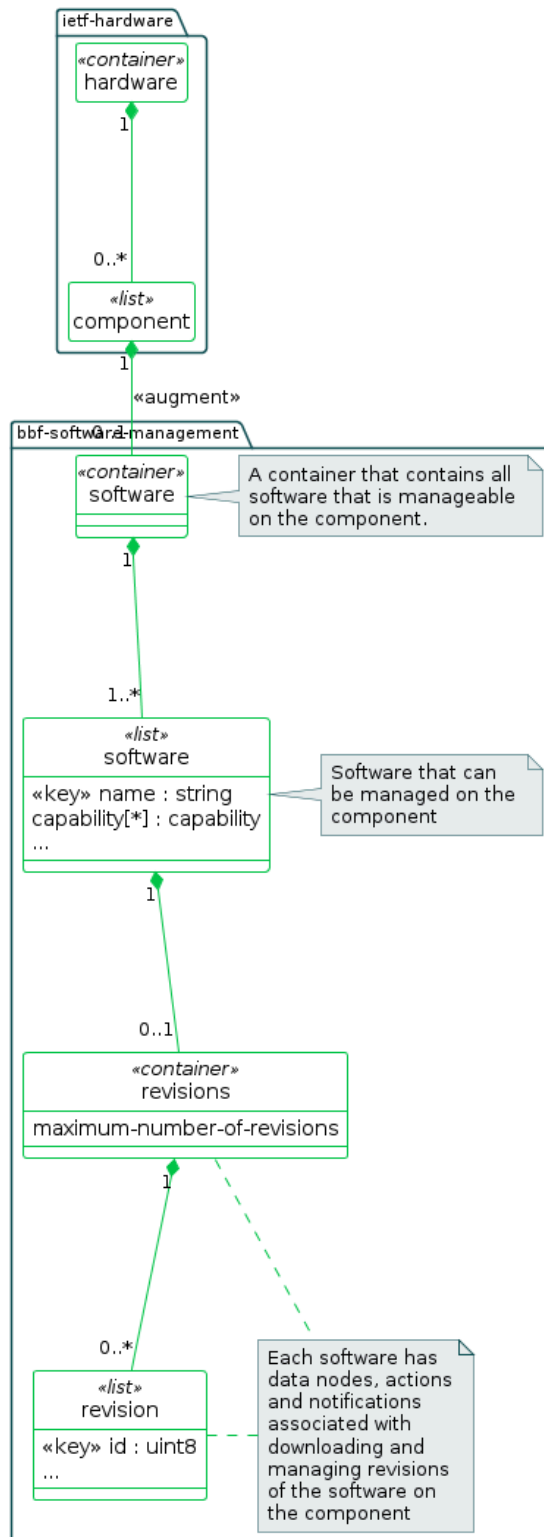


Figure 6 – UML Diagram Showing the Relationship Between Software and Hardware Components

11.2 Management Capabilities

As discussed above, the Software Management YANG data model does not make any assumptions about the software. Consequently, the management capabilities for a software as implemented by a device may vary dependent on the specific type of software or specific implementation. For example, in some cases revisions may be implemented as fixed entities in the device, which may neither be created nor deleted, but only replaced, such as Network Termination (NT) software images managed according to ITU-T G.997.2 Annex S [129]. For some types of software, activating and committing the software may not apply. To enable clients to determine the management capabilities supported by a specific software, the data model enables an implementation to advertise the management capabilities in the leaf 'capability' for each software.

A device may advertise the following capabilities.

activate

The software supports activating software revisions. Whether a revision is currently activated is reported in the leaf 'is-active'. See [Section 11.3.2](#).

alias

The software supports the association of a unique alias to software revisions.

commit

The software supports committing software revisions. Whether a revision is currently committed is reported in the leaf 'is-committed'. See [Section 11.3.3](#).

conditional-activation

The software supports the conditional activation of software revisions applicable to voice calls; this requires the support of the module `bbf-software-management-voice`. See [Section 11.4](#).

delete

The software supports deleting software revisions. See [Section 11.3.4](#).

download-target-selection-by-system

The software supports automatic selection of the revision to be used as the target of a download by the system, see [Section 11.3.1](#).

download-target-selection-by-user

The software supports the client selecting a specific revision to be used as the target of a download, see [Section 11.3.1](#).

validate

The device supports an explicit validation of the software during download and reports this in the leaf 'is-valid'.

NOTE: To support downloading of revisions, at least one of the capabilities 'download-target-selection-by-system' or 'download-target-selection-by-user' must be supported.

Depending on the specific capabilities advertised by a software, specific actions and notifications may or may not be available for that software. See [Section 11.3](#) for more details.

11.3 Managing Revisions

Depending on the management capabilities of a specific software, a client manages revisions of that software by downloading, activating, and committing the software, which moves the revisions through a series of states as shown in [Figure 7](#).

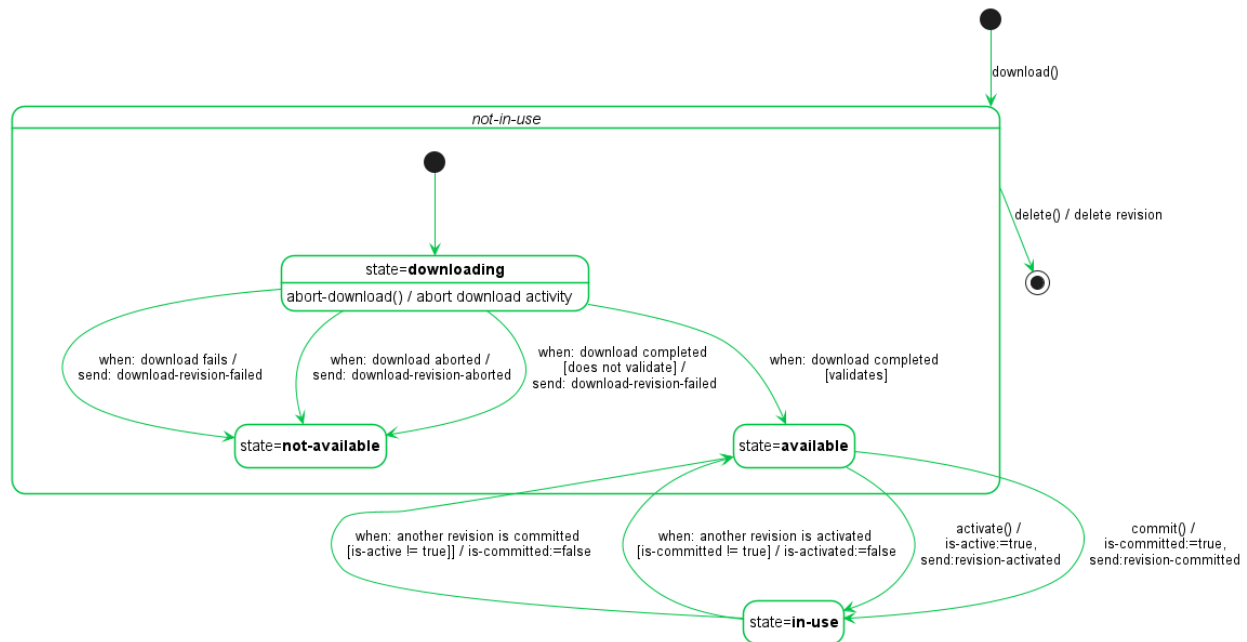


Figure 7 – State Machine of a Revision

11.3.1 Download

If a software supports downloading revisions, it must advertise at least one of the capabilities 'download-target-selection-by-system' or 'download-target-selection-by-user'. If a software does not support download, it may support a fixed set of revisions that may be managed, but this is not within the scope of this Technical Report.

The download of revisions is managed within the container 'download' within the container 'revisions', which is a child node of the list 'software'. Only a single download activity per software is supported at any one time: the container 'download' will not be available in the operational state datastore if a download is in progress.

To initiate an activity to download a revision of a software to a component, the client invokes the action 'download'.

The choice 'source' supports a single case: 'url'. However, vendors may augment this choice to support additional download methods, but this is beyond the scope of this Technical Report.

The choice 'target' gives the client the opportunity to specify how an entry in the list 'revision' is to be selected to store the downloaded revision. Depending on the management capabilities of the specific software, the client may select a specific target revision for the download or let the system select the target revision. Depending on the selection, a new entry in the list 'revision' may be created or an existing entry replaced.

Important to the concept of the Software Management YANG data model is that a successful invocation of the action 'download' must always result in the device creating or replacing an entry in the list 'revision' with the value of the leaf 'id' of that revision as specified in the output element of the RPC, regardless of whether the activity triggered by the action is successful or not. If, for example, the revision cannot be downloaded, is corrupt or not compatible with the software, the revision must enter the state 'not-available'.

An entry must not be created or replaced in the list 'revision' if the invocation of the action is unsuccessful.

If an implementation of the data model limits the maximum number of entries in the list 'revision', the known maximum number of supported entries should be reported in the leaf 'maximum-number-or-revisions' within the container 'revisions'.

A client can abort an on-going download by invoking the action 'abort-download' within the container 'abort-download' of an entry in the list 'revision'.

The client is notified of the results of the activities initiated by the actions discussed above through the notifications 'revision-downloaded', 'download-revision-failed', and 'download-revision-aborted'.

The leaf 'state' of a revision indicates whether the download activity is ongoing or, if it has completed, the outcome of the download as shown in [Figure 7](#).

11.3.2 Activate

When this action is applied to a software revision, execution of the software revision that is currently active will be suspended and the device will load and execute the software represented by the given revision, whereby any steps necessary to achieve this are performed implicitly by the device; such steps are not within the scope of this Technical Report.

If a software supports explicitly activating revisions, it must advertise the capability 'activate'. If a software does not support 'activate', loading/execution of the software would be implementation-specific and is not within the scope of this Technical Report.

The activation of revisions is managed within the container 'activate' of an entry in the list 'revision' and is only available in the operational state datastore for software that supports the capability 'activate' and for revisions of that software that are in the state 'available' or 'in-use'.

To initiate an activity to activate a revision of a software on a component, the client invokes the action 'activate'.

If a restart of the component or device is necessary to activate a specific revision of a software, this must be performed implicitly by the device as part of the activation activity.

The client is notified of the results of the activities initiated by the action 'activate' through the notifications 'revision-activated' and 'activate-failed'.

11.3.3 Commit

When this action is applied to a software revision, the software represented by this revision will be loaded and executed by the device upon subsequent restarts of the device or component.

If a software supports explicitly committing revisions, it must advertise the capability 'commit'. If a software does not support 'commit', whether a specific software revision is loaded and executed at the startup of the device or component would be implementation-specific and is not within the scope of this Technical Report.

Committing revisions is managed within the container 'commit' of an entry in the list 'revision' and is only available in the operational state datastore for software that supports the capability 'commit' and for revisions of that software that are in the state 'available' or 'in-use'.

To initiate an activity to commit a revision of a software on a component, the client invokes the action 'commit'.

The client is notified of the results of the activities initiated by the action 'commit' through the notifications 'revision-committed' and 'commit-failed'.

11.3.4 Delete

When this action is applied to a software revision, the revision will be deleted and no longer available on the component.

If a software supports explicitly deleting revisions, it must advertise the capability 'delete'. If a software supports downloading revisions but not deleting revisions, then revisions would only be able to be replaced when downloading a revision.

Deleting revisions is managed within the container 'delete' of an entry in the list 'revision' and is only available in the operational state datastore for software that supports the capability 'delete' and for revisions of that software that are not in the state 'in-use'.

To initiate an activity to delete a revision of a software on a component, the client invokes the action 'delete'.

The client is notified of the completion of the deletion through the notifications 'revision-deleted' and 'delete-failed'.

11.4 Supporting the Management of Software Upgrade Processes of FastDSL NTs

To support the management of the software upgrade processes in FastDSL Network Terminations (NT) as defined in ITU-T G.997.2. Annex S [129], a device populates a 'remote' hardware component within the operational state datastore of the IETF A YANG Data Model for Hardware Management (RFC 8348) [108] for each NT that it detects as described below:

When an NT is detected on an interface, the device performs the following in the operational state datastore:

- instantiates a component of class 'bbf-hwt:fastdsl-nt';
- on the interface on which the NT was detected, instantiates a reference to this component as defined in the module bbf-interfaces-remote-hardware;
- if the NT supports the management of its software images as supported by the device,
 - instantiates an entry in the software list of the component for each software on the remote device that can be managed by the device;
 - for each software,
 - populates the leaf-list 'capability' with the software management capabilities supported for that software; see Table 2 for the set of capabilities required to support the management of NT software images according to ITU-T G.997.2 Annex S [129]
 - instantiates an entry in the revision list for each revision that it detects on the NT;
- populates the leafs 'mfg-name', 'model-name', 'serial-num' of the fastdsl-nt component from the inventory data provided by the NT;
- optionally populates other optional nodes of the component as supported by the vendor.

When an NT disconnects, the device:

- deletes the component representing the NT in the IETF A YANG Data Model for Hardware Management (RFC 8348) [108].

Table 2 – Capabilities Required to Support the Management of Fastdsl NT Software Images According to ITU-T G.997.2 Annex S

Capability	Remarks
activate	Mandatory as this is a requirement of the software upgrade processes defined in the ITU-T recommendation.
alias	Not required as the software images defined in the ITU-T recommendation do not support an alias.
commit	Mandatory as this is a requirement of the software upgrade processes defined in the ITU-T recommendation.
conditional-activation	Optional for the software upgrade process defined in ITU-T G. 997.2 Annex S. See NOTE 1 .
delete	Cannot be supported, because 2 software images are always automatically instantiated as a requirement of the ITU-T recommendation.
download-target-selection-by-system	Recommended to relieve the client of the need to track which 'id' is available for a download and leave this to the server. See NOTE 2 .
download-target-selection-by-user	Optional. See NOTE 2 .
validate	Mandatory as this is a requirement of the software upgrade processes defined in the ITU-T recommendation.

NOTE 1: To support the requirement of G.997.2 Annex S clause S.7.1.2.1 [129], the module bbf-software-management-voice will need to be supported by the server. This module augments additional data nodes into the input element of the 'activate' action of a revision to enable the client to specify the conditions under which the activation request for a software image shall be executed dependent on whether voice calls are currently in progress.

NOTE 2: At least one of the capabilities 'download-target-selection-by-system' and 'download-target-selection-by-user' must be supported.

12 IPFIX

The intent of this section is to provide some general information regarding the use of the BBF IPFIX data model. A BBF Access Node can be configured to act only as an IPFIX exporter to stream data to an IPFIX collector (e.g., statistics, Inter-Channel-Termination Protocol (ICTP)[8] IPFIX, and other data).

Figure 8 shows the main components of the model that are involved in data export. The data model uses:

- a list of templates
- a list of exporting processes.

Each template refers to one or more exporting processes that will use the template for data export. In a device capable of reporting data through IPFIX, a data template is created and applied to its device resource instance(s).

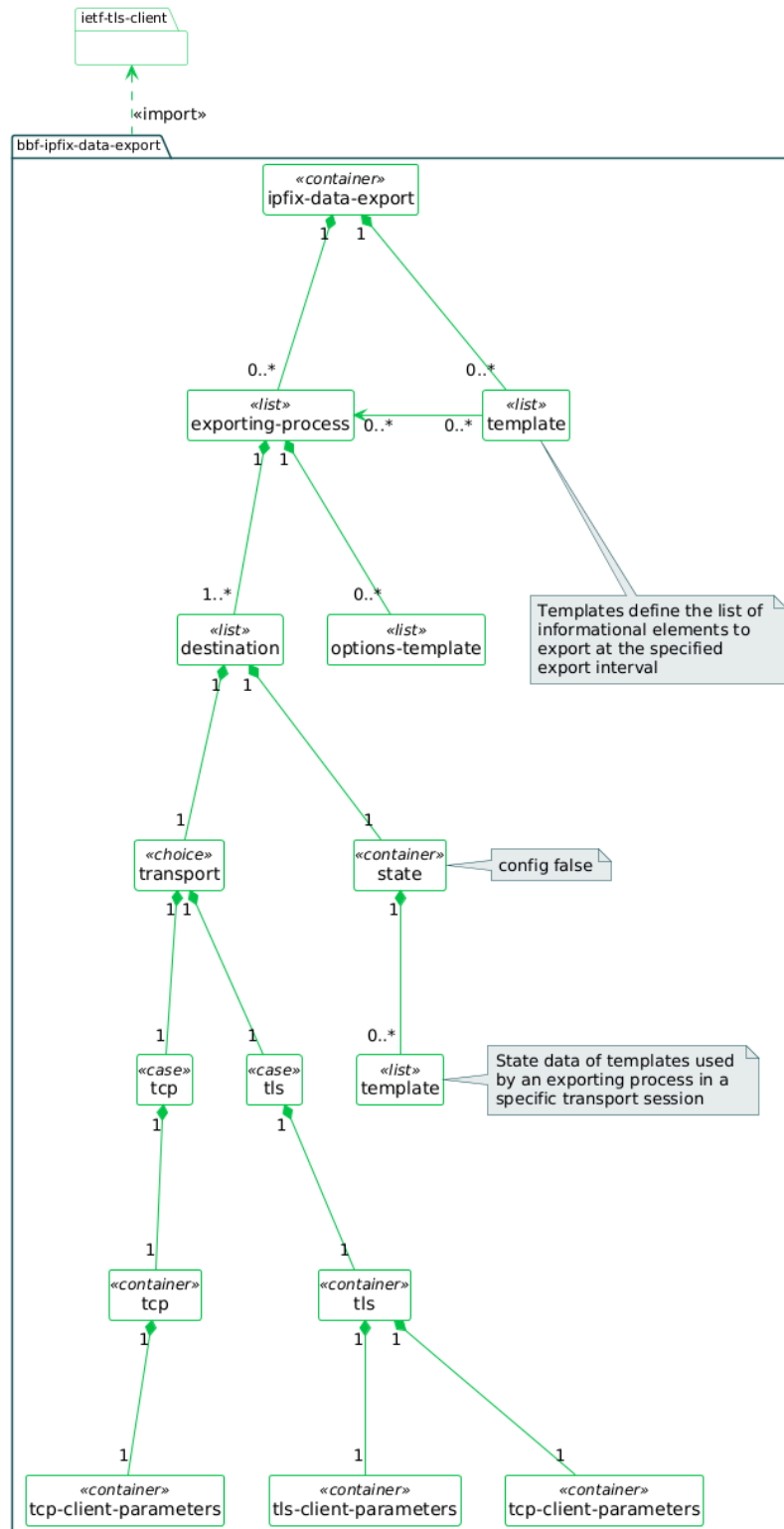


Figure 8 – UML Diagram Showing Relationship Between the Main Components of `bbfx-ipfix-data-export`

12.1 Exporting Process

The list 'exporting-process' specifies destinations to which the data are to be exported. The order in which entries in the list 'destination' are configured has a specific meaning only if the leaf 'export-mode' is set to "fallback".

The list 'exporting-process' also contains the leaf 'exporting-process-id', which corresponds to the Information Element (IE) exportingProcessId [97]. Its purpose is to help associate exporting process reliability statistics exported according to the IPFIX protocol specification [97] with the corresponding Exporting Process [97].

12.2 Exporter

The container 'exporter' within an entry in the list 'destination' contains the configuration for the transport layer to the export destination. In the case of TCP, Transport Layer Security (TLS) should be used unless the data is not sensitive and the data is being transported over a closed network. By configuring the TLS client identity described in RFC 9645[117], Transport Layer Security (TLS) is enabled and configured for this export destination.

12.3 Templates

The list 'template' specifies the data template to be applied to all resources or to a particular set of resources. It defines a list of IE identifiers of resources (via the choice 'resources'), for which Data Records [97] are to be exported and the interval for periodic export of the Data Records in the leaf 'export-interval'. These IE's are defined either in IANA [25] or can be enterprise-specific. Each vendor may define their own list of IE's according to the properties defined under Section 2 of RFC 7012[98]. BBF has its own list of IE's defined in [14].

The leaf-list 'exporting-process' references a list of exporting processes that are to use the template to export data. The list 'template' also provides state information about the template records across all exporting processes.

12.4 Options Templates

The list 'options' enables the selection of additional information to be reported by the Exporting Process to the collector, such as exporting reliability statistics and extended type information. RFC 7011 [97], RFC 5473 [75] and RFC5610 [78] specify several types of reporting information that can be exported through options templates.

12.5 Security

RFC 7011 [97] mandates strong mutual authentication of exporting processes. To prevent on-path-attacks from impostor collecting processes or the export of data to an unauthorized collecting process, strong mutual authentication via asymmetric keys is recommended to be used for TLS.

This model uses the TLS client part of the TLS client server YANG model being defined by RFC 9645[117].

12.6 Transport Session

The container 'transport-session' contains state data of an exporting process and includes the list 'template' that contains the state and statistics about the templates transmitted during the transport session.

Since a given template may be transmitted by more than one Exporting Process, operational state data is maintained separately for each Exporting Process. The list 'field' reports the operational state of individual IEs specified in the template.

The Exporting Process may modify the data being exported to enable a more efficient transmission or storage under the condition that no information is changed or suppressed. For example, the Exporting Process may shorten the length of a field according to the rules of reduced size encoding [97]. The Exporting Process may also export certain fields in a separate data record as described in RFC5473 [75]. Hence the need for some data to be maintained separately in state data though they are part of the configuration data.

13 QoS Policy Data model

This section provides general information regarding the QoS policy data model, the capabilities it provides, and its usage in combination with other YANG data models. It does not intend to cover all possible applications of the model, instead concentrating primarily on the theory of operation.

13.1 QoS Policy Profiles

A QoS Policy profile is a framework used to define QoS actions that are intended to be applied on a frame. It is made up of an ordered list of policies where each policy in turn is an ordered list of classifiers. Each classifier defines a filter and a list of QoS actions to be executed if the frame matches the filter. [Figure 9](#) depicts a pictorial representation of the structure and order of execution of different components in a QoS Policy Profile.

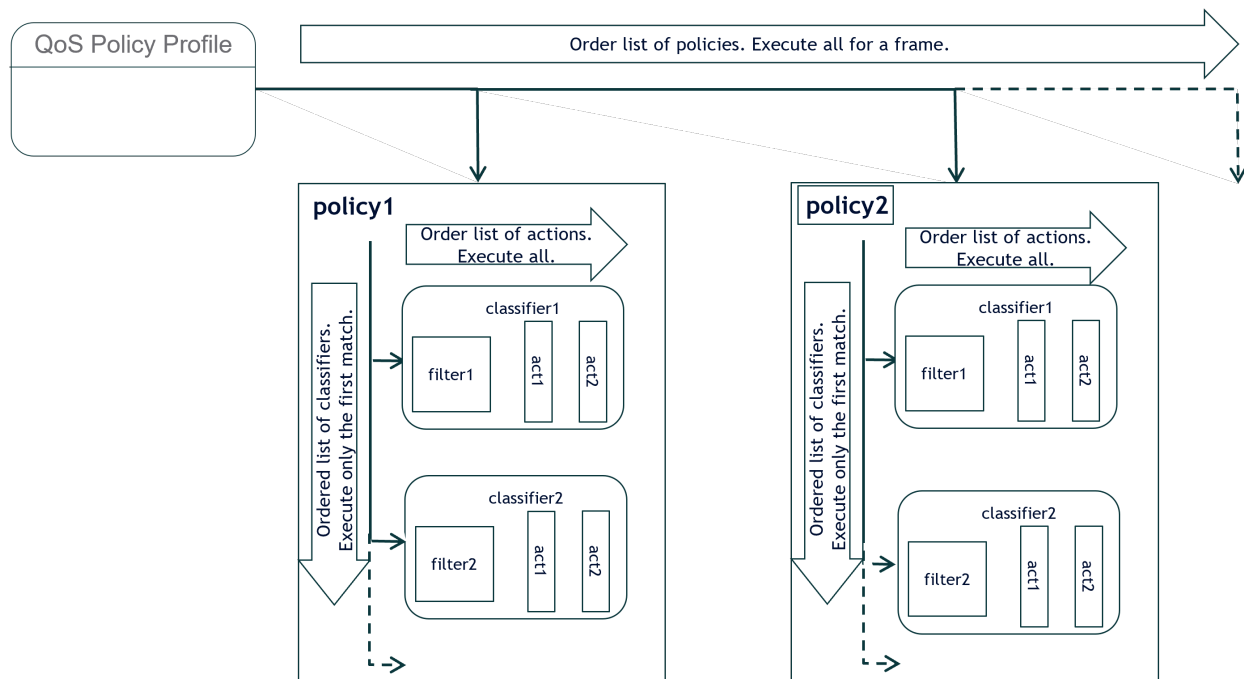


Figure 9 – QoS Policy Profile

The model leverages the possibility of data reuse wherever possible, by keeping the components that forms a QoS Policy Profile in centralized lists directly under the root. This allows:

- Classifier entries to be created at `/bbf-qos-cls:classifiers/bbf-qos-cls:classifier`.
- Policy entries to be created at `/bbf-qos-pol:policies/bbf-qos-pol:policy` with each policy referencing a list of classifier entries from `/bbf-qos-cls:classifiers/bbf-qos-cls:classifier`
- Policy profiles to be created at `/bbf-qos-pol:qos-policy-profiles/bbf-qos-pol:policy-profile` with each policy profile referencing a list of policy entries from `/bbf-qos-pol:policies/bbf-qos-pol:policy`

[Figure 10](#) takes a simple Layer 2 forwarding pipeline defined using TR-383 data models and highlights, in green boxes, the points where a QoS policy profile can be attached in this pipeline.

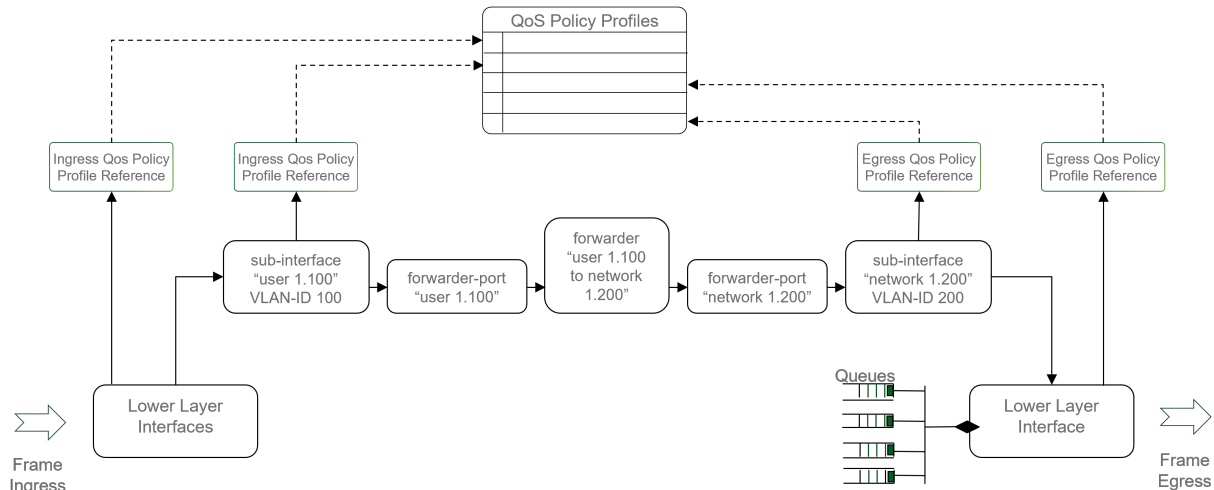


Figure 10 – QoS Profile Association Points

13.1.1 Ingress QoS Policy Profile References

All frames that ingress through a lower-layer interface will be subjected to an ingress QoS policy profile, if referenced on that interface via the leaf 'ingress-qos-policy-profile'. Subsequently, when these frames get classified to one of the VLAN sub-interfaces (VSI) of this lower-layer interface, these frames can get subjected to another level of QoS policy profile, if referenced on that VSI via the leaf 'ingress-qos-policy-profile'.

13.1.2 Egress QoS Policy Profile References

Similar to the ingress QoS policy profiles, all interfaces and VLAN sub-interfaces (VSI) can also reference a QoS policy profile to be applied to all frames that egress through that interface or VSI via the leaf 'egress-qos-policy-profile'. While egressing, as the frame first get processed at the egress VSI and then to its lower-layer interface, the order of execution of the QoS Profile will also be in the same order, i.e., first execute the QoS policy profile referenced at egress VSI and then the one referenced at the its lower-layer interface.

13.2 QoS Classifiers

A classifier is the central part of the QoS policy framework which holds the actual definition of a filter with match criteria and a list of QoS actions to be executed if a frame matches the filter criteria. Note that within a policy, there can be several classifiers but a frame will execute only the first classifier for which it finds a match. And within a classifier, if there are multiple actions defined, the frame will execute all of them. The following subsections look in to the details of filters and actions within a classifier.

13.2.1 QoS Classifier Filters

A filter can be defined either inline within a classifier by selecting the 'bbf-qos-cls:filter-method' choice as 'enhanced-classifier' or as a reference to a common filter definition under /composite-filters/composite-filter by selecting the choice 'composite-filter' for bbf-qos-cls:filter-method. Some of the match criteria that can be used to define a filter are:

- all packets
- source MAC address
- destination MAC address
- p-bit or DEI values present in a frame's VLAN tag

- Ethernet frame type
- IPv4/IPv6 header fields
- Layer 4 Protocol
- Layer 4 Source and Destination port ranges
- p-bit or DEI values stored in the metadata fields (pbit-marking-list/dei-marking-list) of a frame.
- flow color of the frame.

If a filter defines multiple match criteria, then a frame must match all the conditions to satisfy the filter i.e., treated as Logical AND conditions. Note that the above list of match criteria is not a full list but only an abstract to highlight the possibilities.

Note that the leaf 'filter-operation' and bbf-qos-cls:filter-method choice 'inline' are deprecated data nodes as it was causing confusion when the filter-operation was set as "bbf-qos-cls:match-any-filter".

13.2.2 QoS Classifier Action List

Actions for a classifier are defined in-line with in each classifier entry. Each classifier defines an ordered list of actions, each performing a specific action-type. Some of the possible action types are:

- pbit-marking
- dei-marking
- scheduling-traffic-class
- flow-color
- bac-color
- policing
- discard
- permit

Note that 'pbit-marking' and 'dei-marking' actions will not modify the packet header directly, but can only store the remarked p-bit/DEI value in the metadata field 'pbit-marking-list' / 'dei-marking-list'. These QoS metadata fields are described in [Section 13.3](#).

Note that 'policing' is a special action which applies a policer profile. A policer profile combines the configuration of a rate-limiter and also the actions to be performed for each output color from that rate-limiting (e.g. red – discard, yellow – remark DEI, etc.).

13.3 QoS Metadata field associated with frames

The model assumes that each frame has a set of metadata, i.e., data that belongs to the frame but is not a field of the frame itself, associated with it. Once assigned to the frame, the metadata travels with the frame until the frame is transmitted or discarded. This is independent of all frame manipulations that take place prior to the final egress rewrite. The most obvious examples are the traffic class and the Buffer Acceptance Control (BAC) color of the frame which are necessary to do traffic management at the egress queues. Apart from these, there are a few other pieces of information as well, that the TR-383 data model assumes, a frame carries with it as metadata.

13.3.1 P-Bit Marking List

The marking list is a piece of metadata, the content of which is generated via a QoS profile or via the ingress rewrite of a VLAN sub-interface by copying the ingress frame's p-bits to the marking-list. Note that a frame has just one p-bit marking list, i.e., if multiple actions write to the same position in the list, then the last write overwrites the previous. As an example, it is possible to provide a metadata p-bit value to a frame at the

ingress side, this data travels with the frame through the switch, and the metadata is used at the egress side when a VLAN tag is pushed into the frame.

13.3.2 DEI Marking List

A Drop Eligibility Indicator (DEI) marking list is generated via a QoS profile or a QoS profile or via the ingress rewrite of a VLAN sub-interface by copying the ingress frame's DEI value to the marking-list. Note that a frame has just one DEI marking list, i.e., if multiple actions write to the same position in the list, then the last write overwrites the previous. As an example, it is possible to provide a metadata DEI value to a frame at the ingress side, this data travels with the frame through the switch, and the metadata is used at the egress side when a VLAN tag is pushed into the frame.

13.3.3 Flow Color

The Flow Color for a frame can be assigned from a QoS policy, either via an explicit QoS action or implicitly by means of a QoS policer. If there is no QoS policy profile configured or a QoS policy doesn't assign a Flow color for a frame, a system default action can be defined to either assign a fixed frame color or not assign any color but discard if the frame reaches a decision point which needs a flow color.

13.3.4 BAC Color

Buffer Acceptance Control (BAC) Color metadata is used to perform color aware buffer acceptance at the egress queues. A BAC color can be assigned to a frame by an action with in a QoS classifier. If there is no QoS policy profile configured or a QoS policy does not assign a BAC color for a frame, system default actions can be defined to either assign a fixed color or copy the color from the flow color or not assign any color but discard if the frame reaches a decision point which needs a BAC color.

13.3.5 Scheduling Traffic Class.

A traffic class is used, after the forwarding decision has been made and all QoS policies are applied, to identify the queue on the outgoing interface. A traffic class can get explicitly assigned for each frame via a QoS policy action. If it is not explicitly configured, there can be a system default defined via classifiers/ system-default-actions.



End of Broadband Forum Technical Report TR-383