Technical Report

TR-477

Cloud CO Enhancement - Access Node Functional Disaggregation

Issue: 1

Issue Date: January 2024

Notice

The Broadband Forum is a non-profit corporation organized to create guidelines for broadband network system development and deployment. This Technical Report is a draft, is subject to change, and has not been approved by members of the Forum. This Technical Report is owned and copyrighted by the Broadband Forum, and portions of this Technical Report may be owned and/or copyrighted by Broadband Forum members. This Technical Report is only available to Broadband Forum Members and Observers.

Intellectual Property

Recipients of this document 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 if it were to be adopted as a 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	Issue Date	Issue Editor	Changes
1	January 2024	Bruno Cornaglia	Original

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

Editors: Bruno Cornaglia – Vodafone
Francisco de Carvalho – Radisys

Work Area Director(s): Mengmeng Li – China Mobile
Bruno Cornaglia – Vodafone

Project Stream Leader(s): Haomian Zheng – Huawei

Acknowledgements

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

Technical Report: Andre Brizido – Altice Labs
Antonio Marchetta – Reply
Antonio Marsico – Reply
Huseyn Ahmet Aydin – Vodafone
Jeff Hartley – CommScope
Ludwig Pauwels – Nokia
Mauro Tilocca – TIM
Robert Peschi – Nokia
Tim Carey – Nokia
Yu Xie – ZTE

YANG Data Models & Protobuf Files Andre Brizido – Altice Labs
Antonio Marsico – Reply
Jeff Hartley – CommScope
Ludwig Pauwels – Nokia
Robert Peschi – Nokia
Tim Carey – Nokia
Yu Xie – ZTE

Table of Contents

Table of Figures

Table of Tables

Executive Summary

This Technical Report specifies the technical aspects to support disaggregated, multi-vendor interoperable OLT solutions within the CloudCO architectural framework.

It is important for BBF to promote CloudCO standardization (i.e., architectures, Network Function (NF) definitions, interfaces, protocols and data models) in collaboration with relevant open source organizations to facilitate industry adoption of the CloudCO architecture and compliance of open source implementations with the related CloudCO standards.

The requirements, NF definitions, interfaces, protocols and data models defined in this Technical Report provide enablers to be used for industry interoperability testing and certification of NFs and the components that implement these specifications in the Fixed Access Domain.

1 Purpose and Scope

1.1 Purpose

This document specifies the necessary architecture, NF definitions, requirements, interfaces and protocols associated with the Disaggregated OLT (D-OLT) software component enabling the deployment of OLTs with disaggregated functionalities into a CloudCO architecture.

This Technical Report identifies the NFs that more usefully can be disaggregated from traditional OLTs and embedded in the D-OLT. Requirements on the interfaces and protocols between pOLT and the NFs hosted in the D-OLT help to ensure interoperability between different vendors and open source implementations. Use cases and deployment models are also captured in order to identify and define the D-OLT capabilities and associated NFs. Although this Technical Report defines and specifies NFs that can be disaggregated from OLTs, an important goal of this Technical Report is to ensure traditional broadband service offerings and deployments are maintained. In addition, new functionalities can be identified as D-OLT NFs, simplifying the operation and administration of the new NFs as not to be integrated into the physical OLT.

1.2 Scope

The scope of this Technical Report is to:

- Define the NFs that can be disaggregated from traditional OLTs in order to be supported by the D-OLT
- Define interfaces and associated protocol(s) to be used between D-OLT and the pOLT, to include the concept of D-OLT and pOLT in the CloudCO framework described in TR-384 [1], TR-411 [2] and WT-413i2 [20]
- Define interfaces and associated protocol(s) to be used between the D-OLT and the SDN M&C functions in the CloudCO domain
- Identify existing applicable data models (BBF, IETF, other SDOs) for the D-OLT and NFs and further develop existing and/or additional models not sufficiently covered by existing YANG models
- Make recommendations and compliance requirements in order to facilitate open source implementations support for CloudCO. Noting where new work is needed in the BBF, open source communities and other relevant bodies

The scope of the work described above is enabled by:

- Performing a gap analysis between CloudCO and open source solution(s) in order to help define NF candidates for disaggregation, use cases and deployment architecture
- Reviewing the CloudCO architecture in light of requirements for integration of disaggregated NFs and update existing specifications where required

2 References and Terminology

2.1 Conventions

In this Technical Report, several words are used to signify the requirements of the specification. These words are always capitalized. More information can be found be in RFC 2119 [26].

MUST	This word, or the term “REQUIRED”, means that the definition is an absolute requirement of the specification.
MUST NOT	This phrase means that the definition is an absolute prohibition of the specification.
SHOULD	This word, or the term “RECOMMENDED”, means that there could exist valid reasons in particular circumstances to ignore this item, but the full implications need to be understood and carefully weighed before choosing a different course.
SHOULD NOT	This phrase, or the phrase "NOT RECOMMENDED" means that there could exist valid reasons in particular circumstances when the particular behavior is acceptable or even useful, but the full implications need to be understood and the case carefully weighed before implementing any behavior described with this label.
MAY	This word, or the term “OPTIONAL”, means that this item is one of an allowed set of alternatives. An implementation that does not include this option MUST be prepared to inter-operate with another implementation that does include the option.

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 www.broadband-forum.org.

Document	Title	Source	Year
[1] *TR-384*	Cloud Central Office Reference Architectural Framework	BBF	2018
[2] *TR-411*	Definition of interfaces between CloudCO Functional Modules	BBF	2021
[3] *TR-413*	SDN Management and Control Interfaces for CloudCO Network Functions	BBF	2018
[4] *TR-402*	Functional Model for PON Abstraction Interface	BBF	2018

[5]	*TR-403*	PON Abstraction Interface for Time-Critical Applications	BBF	2018
[6]	*TR-370*	Fixed Access Network Sharing - Architecture and Nodal Requirements	BBF	2020
[7]	*TR-386*	Fixed Access Network Sharing - Access Network Sharing Interfaces	BBF	2019
[8]	*TR-101*	Migration to Ethernet-Based Broadband Aggregation	BBF	2011
[9]	*TR-156*	Using GPON Access in the context of TR-101	BBF	2017
[10]	*WT-474*	Subscriber Session Steering	BBF	
[11]	G.984.3	Gigabit-capable passive optical networks (G-PON): Transmission convergence layer specification	ITU-T	2014
[12]	G.987.3	10-Gigabit-capable passive optical networks (XG-PON): Transmission convergence (TC) layer specification	ITU-T	2014
[13]	G.988	ONU management and control interface (OMCI) specification	ITU-T	2017
[14]	RFC6241	Network Configuration Protocol (NETCONF)	IETF	2011
[15]	*TR-435*	NETCONF Requirements for Access Nodes and Broadband Access Abstraction	BBF	2020
[16]	*RFC7950*	The YANG 1.1 Data Modeling Language	IETF	2016
[17]	*gRPC*	https://grpc.io/	Linux Foundation	2021
[18]	*GPB*	https://developers.google.com/protocol-buffers/docs/proto3	Google Developers	
[19]	*TR-436*	Access & Home Network O&M Automation/Intelligence	BBF	2021
[20]	*WT-413i2*	SDN Management and Control Interfaces for CloudCO Network Functions	BBF	2021
[21]	*RFC7540*	Hypertext Transfer Protocol Version 2 (HTTP/2)	IETF	2015
[22]	*RFC5246*	The Transport Layer Security (TLS) Protocol Version 1.2	IETF	2008
[23]	*TR-178*	Multi-service Broadband Network Architecture and Nodal Requirements	BBF	2014
[24]	*TR-280*	ITU-T PON in the context of TR-178	BBF	2020
[25]	*TR-451*	vOMCI Specification	BBF	
[26]	*RFC2119*	Key words for use in RFCs to Indicate Requirement Levels	IETF	1997
[27]	*TR-383a5*	Common YANG Modules for Access Networks	BBF	2021
[28]	*TR-385i2*	ITU-T PON YANG Modules	BBF	2020
[29]	*TR-486*	Interfaces for Automated Intelligent Management	BBF	2023
[30]	*RFC7951*	JSON Encoding of Data Modeled with YANG	IETF	2016
[31]	*TR-489*	ONU Authentication and Selection of eOMCI or vOMCI	BBF	2023

[32] *RFC6470*	Network Configuration Protocol (NETCONF) Base Notifications	IETF	2012
[33] *RFC7223*	A YANG Data Model for Interface Management	IETF	2014
[34] *TR-484*	Access Network Abstraction	BBF	2022

2.3 Definitions

The following terminology is used throughout this Technical Report.

Disaggregated OLT	D-OLT represents a logical host for all the functions to be virtualized. It can be located in local servers close to pOLT or centrally in the cloud. It can be also split among multiple locations (local and central).
Physical OLT	pOLT supports the functions related to access lines, L2 user plane functions and specifically all the related functions that are not virtualized.

2.4 Abbreviations

This Technical Report uses the following abbreviations:

AAA	Authentication, Authorization, and Accounting
ACK	Acknowledge
AIM	Automated Intelligent Management
AIMO	AIM Orchestrator
API	Application Programming Interface
BAA	Broadband Access Abstraction
BAL	Broadband Adaptation Layer
BMap	Bandwidth Map
BufOcc	Buffer Occupancy
CHAP	Challenge Handshake Authentication Protocol
CI/CD	Continuous Integration / Continuous Delivery
CNF	Containerized Network Function
CP	Control Plane
CPRi	Control Protocol Redirect Interface
CPU	Central Processing Unit
CSP	Communications Service Provider
DBA	Dynamic Bandwidth Allocation
DE	Decision Element
DHCP	Dynamic Host Configuration Protocol
DHCP RA	DHCP Relay Agent
D-OLT	Disaggregated OLT

EoL	End of Life
EoM	End of Maintenance
E2E SO	End to End Service Orchestrator
FANS	Fixed Access Network Sharing
FCAPS	Fault, Configuration, Accounting, Performance, Security
FFS	For Further Study
GPB	Google Protocol Buffers (Protobufs)
gRPC	Google RPC
HTTP	HyperText Transfer Protocol
HVAC	Heating, Ventilation and Air Conditioning
IEEE	Institute of Electrical and Electronics Engineers
IA	Intermediate Agent
IETF	Internet Engineering Task Force
IP	Internet Protocol
IPCP	Internet Protocol Control Protocol
ITU-T	International Telecommunication Union – Telecommunication Standardization Bureau
JSON	JavaScript Object Notation
L2	Layer 2
L3	Layer 3
LCP	Link Control Protocol
LoID	Logical Identifier
MAC	Medium Access Control
MANO	Management and Orchestration
ME	Managed Entity
MGMD	Multicast Group Membership Discovery
Minf	Management Interface
Mfc	Flow Control Interface
MP	Management Plane
NETCONF	Network Configuration Protocol
NFV	Network Functions Virtualization
NFVI	NFV Infrastructure
NFVO	NFV Orchestrator
NOS	Network Operating System
OAM	Operations, Administration and Management
OCP	Open Compute Project
OF	OpenFlow
OLT	Optical Line Termination
OMCI	ONU Management and Control Interface
ONU	Optical Network Unit
NETCONF	Network Configuration Protocol
P4	Programming Protocol-independent Packet Processors

PADI	PPPoE Active Discovery Initiation
PADO	PPPoE Active Discovery Offer
PADR	PPPoE Active Discovery Request
PADS	PPPoE Active Discovery Session-confirmation
PAP	Password Authentication Protocol
PHY	Physical
PIM	Protocol-Independent Multicast
PNF	Physical Network Function
pOLT	Physical OLT
PON	Passive Optical Network
PPP	Point-to-Point Protocol
PPPoE	PPP over Ethernet
PPPoE IA	PPPoE Intermediate Agent
QoS	Quality of Service
RA	Relay Agent
RegID	Registration Identifier
RFC	Request For Comments
RG	Residential Gateway
ROI	Return Of Investments
RPC	Remote Procedure Call
RU	Rack Unit
SCi	State Control Interface
SDN	Software Defined Networking
SDN M&C	SDN Management and Control
SFP	Small Form Pluggable
SLA	Service Level Agreement
SN ID	Serial Number Identifier
T-CONT	Traffic Container
TCP	Transmission Control Protocol
TLS	Transport Layer Security
ToC	Table of Contents
TR	Technical Report
UC	Use Case
UDP	User Datagram Protocol
UNI	User Network interface
URL	Uniform Resource Locator
vAN	Virtual Access Node
vBMap	Virtual BMap
vDBA	Virtual DBA
VIM	Virtualized Infrastructure Manager
VNF	Virtual Network Function

VNFM	VNF Manager
VNO	Virtual Network Operator
VOLTHA	Virtual OLT Hardware Abstraction
vOLTMF	vOLT Management Function
vOMCI	Virtual OMCI
WA	Work Area
WT	Working Text
XML	Extensible Markup Language
YANG	Yet Another Next Generation
ZTP	Zero Touch Provisioning

3 Technical Report Impact

3.1 Energy Efficiency

Energy Efficiency may be impacted migrating from legacy OLT to Disaggregated OLT. In fact, moving some functions to external servers can increase the energy consumed. On the other hand, optimizing by having a single D-OLT instance managing multiple physical OLTs can reduce the energy consumption and having a better energy efficient solution.

Regulatory differences related to electrical power, Heating, Ventilation and Air Conditioning (HVAC) and fire protection between traditional Central Offices and datacenters are out-of-scope for this document.

3.2 Security

The Disaggregated OLT is subject to the security concerns applicable to the legacy OLT and described in other BBF technical reports, like TR-101 [8], TR-178 [23], TR-156 [9] and TR-280 [24].

Where not explicitly described in directly relevant standards, implementations of Disaggregated OLT includes a description of security considerations, possibly expanding on related referenceable standards.

In addition, any implementation of Disaggregated OLT functionalities includes documentation of special requirements that apply specifically to that implementation – such as measures needed to ensure that only authorized parties can access, or modify, configuration for underlying network infrastructure and that traffic associated with one subscriber/tenant cannot be intercepted by other subscribers/tenants.

Finally, because of the distribution of control and data plane functions defined for Disaggregated OLT, the Mfc protocols include capabilities for providing secure, authorized and authenticated communication between D-OLT and physical OLT.

Operators consider using these capabilities as an important means for preventing attacks intended (for example) to divert or disable data forwarding capabilities through control plane impersonation.

3.3 Privacy

The Disaggregated OLT is subject to the privacy concerns applicable to the legacy OLT and described in other BBF technical reports, like TR-101 [8], TR-178 [23], TR-156 [9], TR-280 [24] and TR-384 [1].

In network protocols, privacy concerns, beyond the protection of potentially private data, focus on two aspects:

1. The potential for tracking of users through exposure of Personal Identifying Information (PII)
2. The potential for correlation of user activity over time through persistent use of network identifiers

Additionally, privacy involves the need to ensure that information to, from and between subscribers/tenants can only be accessed by those who have the right to do so. Further, privacy requirements can vary by regulatory region. In general, two ways to ensure privacy are recognized:

- Preventing data, from being copied to a non-intended destination.
- Encrypting data, so that it cannot be understood even if it is intercepted.

Because of its disaggregated nature, the same (or highly correlated) identifying information may be seen at several points in the network, allowing for identification of a target subscriber or D-OLT component. This increases the number of network elements with potential exposure to privacy violations.

4 Introduction

4.1 Access Disaggregation Principles

Access Network disaggregation refers to the transformation of the Access Network from a set of traditional monolith hardware nodes with tightly coupled software to a set of distributed and cooperating functions that can be hosted in different hosting platforms that include the traditional nodes, general purpose hardware appliances or virtualized as network functions within Cloud platforms. The transformation of the Access Network to these hosting ecosystems requires certain functions provided by the Access network to be disaggregated from the traditional hardware nodes and requires:

- introduction of Disaggregated OLT (D-OLT) systems to more flexibly, nimbly and sustainably evolve and manage the access assets
- reorganization and modernization of the aggregation infrastructure towards bandwidth flexible Leaf/Spine architectures to follow service and traffic needs in an invest-as-you-grow fashion
- migration from traditional Delivery and Assurance chains to unified, multi-vendor/multi-technology, automated SDN platforms to manage and control the underlying assets and service flows

The disaggregation of the Access Network into a set of distributed and cooperating functions provides benefits to Service Providers in terms of:

- overcoming chronic issues, such as vendor interoperability and lock-in, long network upgrade cycles, lack of workflow agility, …
- introducing improvements to current practices for the Access Network and service engineering, deployment and operation practices.

The successful disaggregation in the Access Network relies on the following technology enablers related to Access/Aggregation hardware products and associated software applications that are hosted within the network node or the Cloud platform:

a) Hardware design simplification and modularity: hardware is designed using reference designs that can be used in a variety of applications. For example, OLT reference designs can isolate per-port OLT MACs that fits variety of PON applications.

Expected benefit(s)
- ✓ *Flexibility in network growth and investment and adaptability of newly procured hardware to then-current technology, bandwidth as well as features and service needs.*
- ✓ *Optimisation of expenditure flows for hardware procurement and maintenance*

b) Programmable and Repurposable Hardware: Hardware should support an open installation and execution environment (e.g. OCP's Open Network Install Environment, ONIE), preferably based on an independent Network Operating System (NOS) that enables:

- Clean separation of the user plane forwarding/routing functions from technology-specific aspects of the underlying interfaces.
- Repurposable networking features: broad reprogrammability of forwarding/routing capabilities and control packet interception rules and policies.
- Repurposable access features (if supported by the hardware): ability to reprogram (as CPU and memory allow) supported access technology features and to easily forklift the access interfaces (e.g. via plug-and-use SFP modules and or programmable transceivers).

- Zero-touch reprogramming of the hardware in a least intrusive fashion, purely based on remotely issued software and/or firmware updates and on alignment of the interface(s) to expose those updated capabilities.

Expected benefit(s)
- ✓ *Adapt, to a given extent, deployed devices to new service models and features*
- ✓ *Broader supplier ecosystem and innovation-based competition*
- ✓ *Shorter cycles for technology and service updates*
- ✓ *Improved network infrastructure ROI by postponing the EoM/EoL of deployed devices*

c) Software modularity and openness: Software is designed with the ability to migrate network functions from hardware nodes into distributed Cloud platforms; this aspect complements the principles of Hardware design simplification, modularity and programmability.

- Device software and firmware functions transition from monolithic release packages towards modular architectures that enable shorter lifecycles for testing and introducing network and service updates.
- Service Providers can select software functions from different parties, potentially inspired by Open Source distributions and compliant with carrier grade architecture specifications and programmable and extensible standard interfaces.
- Evolving to SDN Management and Control that is designed with open interfaces and modularity to achieve:
  - unified multi-vendor/multi-technology management and control
  - short feature update cycles
  - interoperability with third party applications
  - management and control interfaces that are programmable by the Service Provider and can easily adapt to network and service evolutions

Expected benefit(s)
- ✓ *Simplified network validation and engineering of highly interoperable components*
- ✓ *Coexistence of legacy and new nodes under unified management chains and workflows*
- ✓ *Streamlined network deployment and upgrades*
- ✓ *Streamlined new services introduction and features improvement to deployed services*
- ✓ *Improved operations with less error-prone and automated OAM*
- ✓ *Features evolution leveraging on interchangeable modules from a more open ecosystem*

d) Cloud-based environment: software modules and applications are hosted either on the network nodes or Cloud platforms and are conceived as part of a cloud native environment.

Expected benefit(s)
- ✓ *Exploit CI/CD and Distributed Applications practices in the lifecycle of "network services"*
- ✓ *Introduce Data Center-like node designs, scale and procedures*
- ✓ *Blending the best of Networking, IT departments and key stakeholders in improving operations*

4.1.1 Disaggregated OLT Functional Architecture

The Disaggregated OLT (D-OLT) provides coordination of applications used to manage and control broadband services for one or more subscribers in the context of the capabilities provided by an OLT. Targets of the broadband services can be tenants of the service provider or end-user subscribers. The scope of the D-OLT instance, includes the management and control of one or more NFs that support a set of broadband services for one or more tenant/subscribers. The tenants/subscribers are attached to one or more OLTs.

4.1.2 Deployment of the D-OLT in CloudCO

When deployed within the context of the CloudCO, the D-OLT is designated as a function within the Access SDN Management and Control layers, and it implements the requirements and capabilities of the D-OLT as well as the interfaces toward the pOLT. If NFs are deployed as separate functions, the D-OLT implements the interfaces toward the NFs and other Access SDN Management entities for management purposes of the D-OLT and its related NFs. The decision whether to host the D-OLT within the centralized Access SDN Management and Control functions, in an OLT device, the BAA layer or a separate computing platform is based on the non-functional requirements of the NF (event latency, application scalability) needed for the implementation of its capabilities.

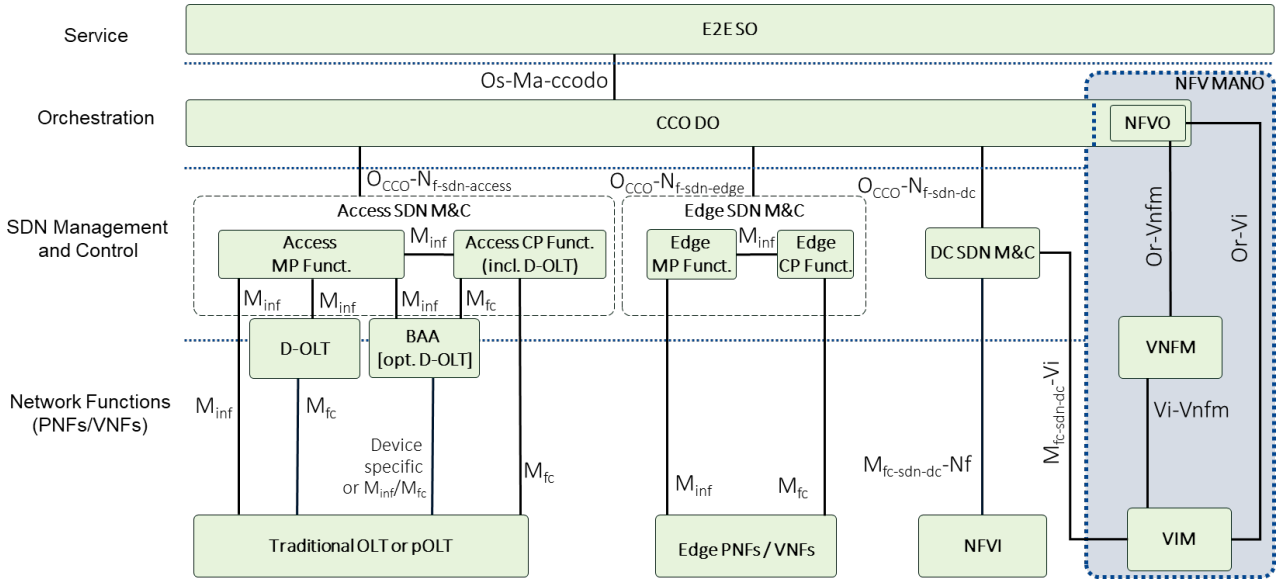

Figure 1 – Disaggregated OLT in CloudCO

When deployed as a function within the Access SDN Management and Control, the D-OLT can be hosted with the centralized Access SDN Management functions providing functions of the OLT that are centralized or the D-OLT can be hosted with Access SDN Management functions on Edge locations, as described above, but without any required adaptation.

There is not necessarily "just one D-OLT" -- D-OLT functions can be widely dispersed, with some in the SDN M&C, some in a detached BAA layer (like a "D-OLT platform with adapters for specific OLTs"), and/or independently-scaled standalone containers. The data models are designed to be likewise flexible, allowing for usage within and without bbf-device-aggregation.

4.1.3 Deployment option in case of FANS

As indicated in the Figure 3, option (c) of this Technical Report, multiple D-OLTs could manage the same pOLT. This is a network sharing/slicing scenario where the virtualized functions refer to the same physical entity (e.g. FANS - “Fixed Access Network Sharing”).

FANS functionality enables VNOs to share the same pOLT by using the implementation and interfaces described in TR-370i2 [6] and TR-386 [7].

In the case of Disaggregated OLT (D-OLT), two scenarios are possible:

- vAN-based FANS, where a D-OLT instance is assigned to a single VNO
- SDN-based FANS, where a D-OLT instance is shared across all the VNOs

The D-OLT provides coordination of applications used to manage and control broadband services for one or more subscribers in the context of the capabilities provided by a pOLT and described in section 4.5.

For both scenarios, the solution provides access to the VNOs for their configurations on the assigned pOLT resources to guarantee that the FANS business framework is fulfilled.

4.2 High level architecture of the Disaggregated OLT

TR-384, [1], describes in section 5.2.6 a possible decomposition of both Access Node (Figure 6 of TR-384, [1]) and OLT (Figure 7 of TR-384, [1]). Section 5.2.6.2 of TR-384, [1], also gives a list of possible candidate functions that can be virtualized, like for example:

- DHCP Relay Agent
- PPPoE Intermediate Agent
- Multicast Snooping

Since TR-384 [1] does not specify which NFs can be virtualized and neither the interfaces between the virtualized NFs and the physical Access Node, this Technical Report[1] specifies the NFs that can be virtualized together with the interface (protocol, data models) between those NFs and the physical Access Node. These virtualized NFs are allocated into an entity called Disaggregated OLT (D-OLT).

The objective of this Technical Report is twofold:

- to define and standardize the functions that can be virtualized
- to define the interface (protocol, data models) between those functions allocated into a logical Disaggregated OLT (D-OLT) and the physical OLT (pOLT).

Future releases of the document can extend the number of virtualized functions.

Table 1 – Functions that can be virtualized in a D-OLT

#	Function	Type	BBF TR Reference	Description
1	DHCP Relay Agent	UP/CP	TR-101 [8] TR-156 [9]	DHCP RA is a function that receives DHCP packets from the pOLT and processes them locally. It may append new options and forward the packets to the DHCP server either via control plane or data plane.
2	PPPoE Intermediate Agent	UP/CP	TR-101 [8] TR-156 [9]	PPPoE IA is a function that receives PPPoE packets from the pOLT and interacts with the PPPoE server. It may append new tags and forward the packets to the PPPoE server either via control plane or data plane.
3	Multicast Proxy / Snooping	UP/CP	TR-101 [8] TR-156 [9]	The MGMD D-OLT NF receives copies of multicast messages (join, leave, etc.) to offload some of the reporting and statistics functions from the pOLT.
4	ONU Authentication	UP/CP	ITU-T G.984.3 [11] ITU-T G.987.3 [12] ITU-T G.988 [13]	i. The ONU Authentication function moves the OLT's authentication of ONTs to the D-OLT, to prevent unauthorized ONTs from coming online, [31].
5	vDBA	UP/CP	TR-402 [4] TR-403 [5]	In traditional PONs, a single DBA scheme is implemented in the OLT hardware. The vDBA operates within a D-OLT. The vDBA allows each tenant to control upstream capacity, latency and jitter for the T-CONTs allocated to the tenant.
6	vOMCI	MP and UP/CP	TR-451	Virtualized OMCI (vOMCI) solution moves the OMCI functionality that is traditionally embedded within Optical Line Termination (OLT) network elements into the Operator's cloud network.

4.2.1 Virtualization Support

Disaggregated OLT can support a wide range of functions and Table 1 shows the list of them.

4.2.2 Architecture Diagram

Figure 2 shows the architecture of the Disaggregated OLT (D-OLT), including the interfaces Minf and Mfc between Access SDN M&C and D-OLT and physical OLT. Note that in order to maintain broad compatibility between the wide variety of D-OLT deployment options, the "Device Aggregation" YANG model may optionally be used with the Minf, for compatibility with BAA (TR-484) deployments. Thus the pOLT may be mounted to the D-OLT. This provides a standardized data model location for the name, type, protocol endpoint-establishment, and other data model libraries of each function. This increases D-OLT normalization with existing TR-385 Separated-NE Mode D-OLTs, BAAs, VOLTHAs and other adapters that use the Common YANG "Device Aggregation" model and methodology.

This Technical Report defines the interface Mfc between Disaggregated OLT (D-OLT) and physical OLT. The Mfc interface is composed of three different sub-interfaces that perform actions according to the function to be virtualized:

- **Management Interface (Mfc_Minf)**: The D-OLT function uses the Mfc_Minf to manage pOLT configuration, operational data, RPCs, and notifications, specific to only the disaggregated functions.
- **Control Packet Redirect Interface (Mfc_CPRi)**: This interface is required to forward and tunnel control packets such as DHCP and PPPoE between D-OLT and physical OLTs. (Note: the term "Packet" in the context of CPRi may also include "frames" in the case of certain Control protocols, such as a DHCP DISCOVER frame or OMCI frame. Thus the word "packet" is simply a documentation convenience, not a constraint.)
- **State Control Interface (Mfc_SCi)**: This interface is used to dynamically program rules both to change user plane traffic behavior and to redirect packets between user plane and control plane. After the traffic rules are programmed onto the physical OLT, the physical OLT will forward the subscriber control and/or data traffic according to the rules.

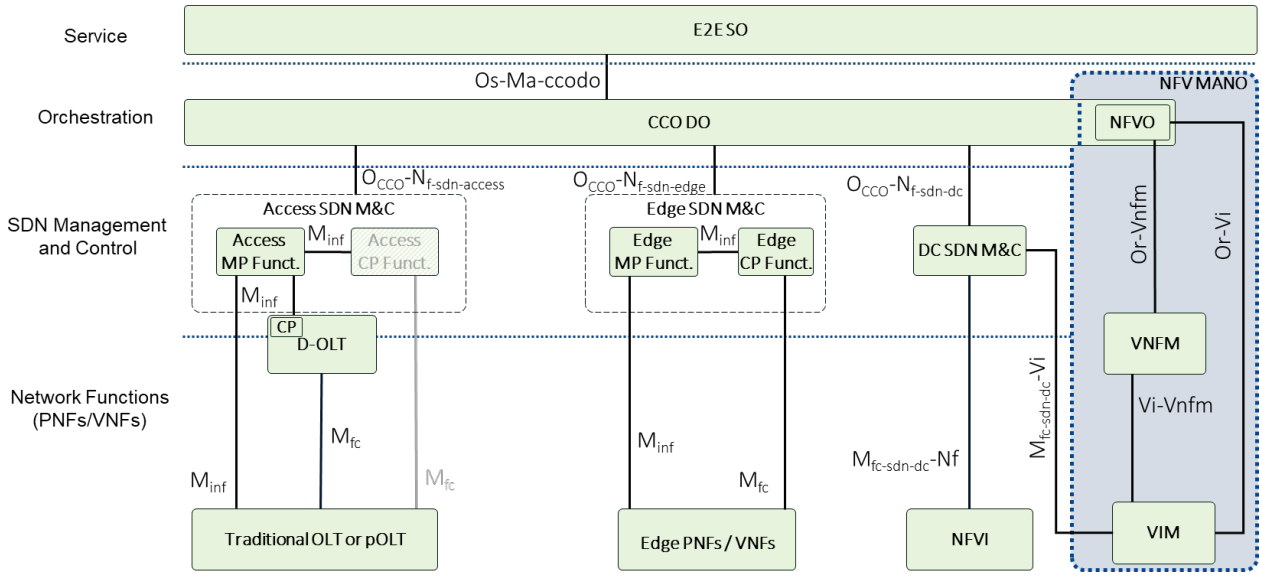

Figure 2 – Disaggregated OLT Architecture

Figure 3 describes three possible deployment scenarios for the D-OLT:

a. Direct relationship between D-OLT and physical OLT. This represents a D-OLT that only manages one physical OLT.
b. A single D-OLT managing multiple physical OLTs. This is relevant, for example, when certain functions that need to be virtualized are centralized.

c. Multiple D-OLTs managing the same physical OLT. This is relevant, for example, in case of network sharing or slicing when multiple virtualized functions refer to same physical entity.

Note that one logical D-OLT depicted in Figure 4 may be practically implemented as a cluster of multiple instances for redundancy purposes, however that does not change the functional deployment scenarios described, nor the interfaces.

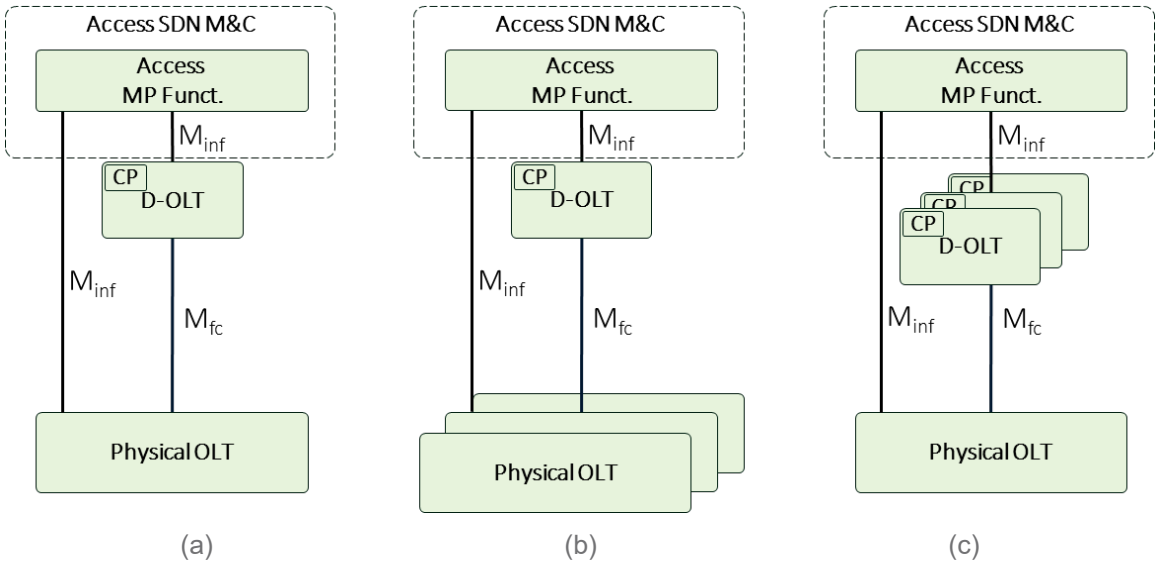

Figure 3 – Deployment scenarios for D-OLT

4.2.3 Interfaces & Definitions

As reported in Figure 2 the interfaces impacted by Disaggregated OLT are the following:

- The **Minf** reference point provides management plane FCAPS functionality, and the associated interface protocol is typically based on NETCONF/YANG, [3].
- The **Mfc** reference point provides control plane functionality and can utilize a number of control plane protocols for the various sub-interfaces of the reference point. Table 2 reports the protocols and data models used for the 3 sub-interfaces of the Mfc interface.

Table 2 – Protocols and data models for the 3 sub-interfaces of Mfc interface

Sub-interface (Mfc)	Functional Category	Protocol	Encoding	Data Model
Mfc_Minf	Management	NETCONF [15] [16]	XML	YANG [17]
Mfc_SCi	Flow Control	gRPC [18]	GPB [19]	YANG [17]
Mfc_CPRi	Flow Control	gRPC [18]	GPB [19]	N/A

4.2.4 Telemetry Considerations

This Working Text does not replace/modify requirements nor design of telemetry requirements described by TR-436 [19], TR-486 [29] and TR-413 [3], but a D-OLT may serve as a convenient location for certain telemetry components described in those documents.

Any function virtualized to a D-OLT is expected also to be compliant to its (original) pOLT requirements for counters/statistics.

4.3 pOLT Functional Architecture

The pOLT functional architecture and related interfaces are described in Figure 4. The pOLT supports the following functions:

- functions related to access lines
- L2 user plane functions
- all the related functions that are not virtualized

It shall also support the following interfaces:

- Minf towards the Access SDN M&C for FCAPS functionalities related to all the functions that are not virtualized according to WT-413i2, [20].
- Mfc interface towards the D-OLT for all the functions that are virtualized according to Table 1 of section 4.2.1.

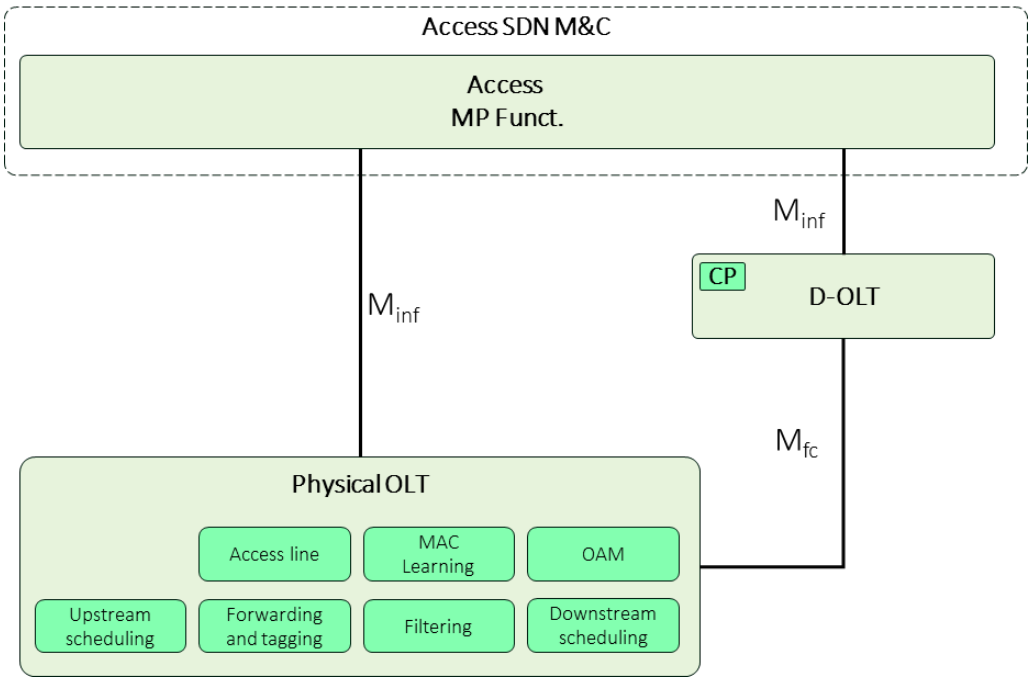

Figure 4 – pOLT Functional Architecture

4.4 D-OLT Functional Architecture

D-OLT functional architecture is reported in Figure 5. D-OLT represents a logical host for all the functions to be virtualized. It can be located in local servers close to pOLT, centrally in a cloud, or split between two locations.

All the functions that can be virtualized are reported in Table 1 of section 4.2.1.

The D-OLT supports the following interfaces for all the NFs hosted by the D-OLT:

- Minf towards the Access SDN M&C for FCAPS functionalities
- Mfc towards pOLT

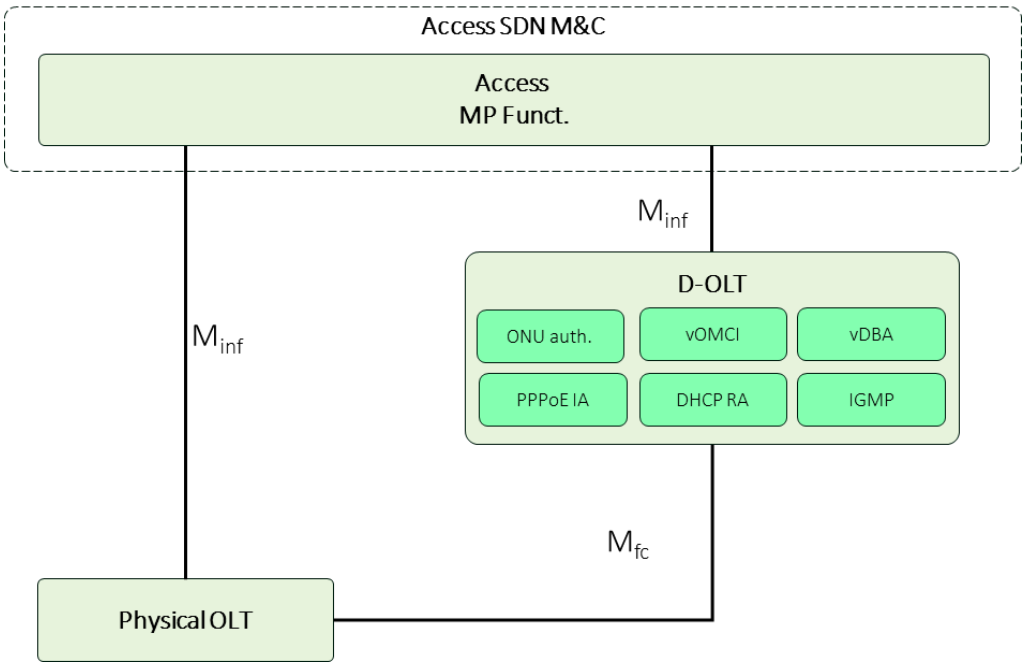

Figure 5 – D-OLT Functional Architecture

4.5 Call Flows between D-OLT and pOLT

This section describes the call flows for the functions listed in the Table 1 of section 4.2.1.

4.5.1 DHCP RA call flow

The pOLT and D-OLT provide the capabilities to relay DHCP traffic between a subscriber device and the service provider's DHCP server. Prior to providing the DHCP relay capabilities, the pOLT and D-OLT need an initialization procedure to enable the functionality and setup the communication channels. The initialization procedure also involves configuration applied whenever the subscriber service is pre-provisioned. Subscriber services can be configured on a per subscriber basis. One aspect of the DHCP relay functionality is that the subscriber's broadband service is considered pending until the DHCP procedure is completed. In some cases, this DHCP procedure is suspended until the subscriber has been identified and authenticated by the system.

1. The Access MP Function configures the D-OLT with profile configuration to the DHCP RA NF.

   Note: The Access MP function may configure the DHCP RA parameters for one or more subscribers and can perform this step at any time. Typically, a single configuration is performed when all the subscriber's profiles can be pre-provisioned at the same time.

2. The Access MP Function configures parameters to initialize the communication between pOLT and D-OLT (e.g., endpoints IP addresses, ports, protocol) via the Minf interface.

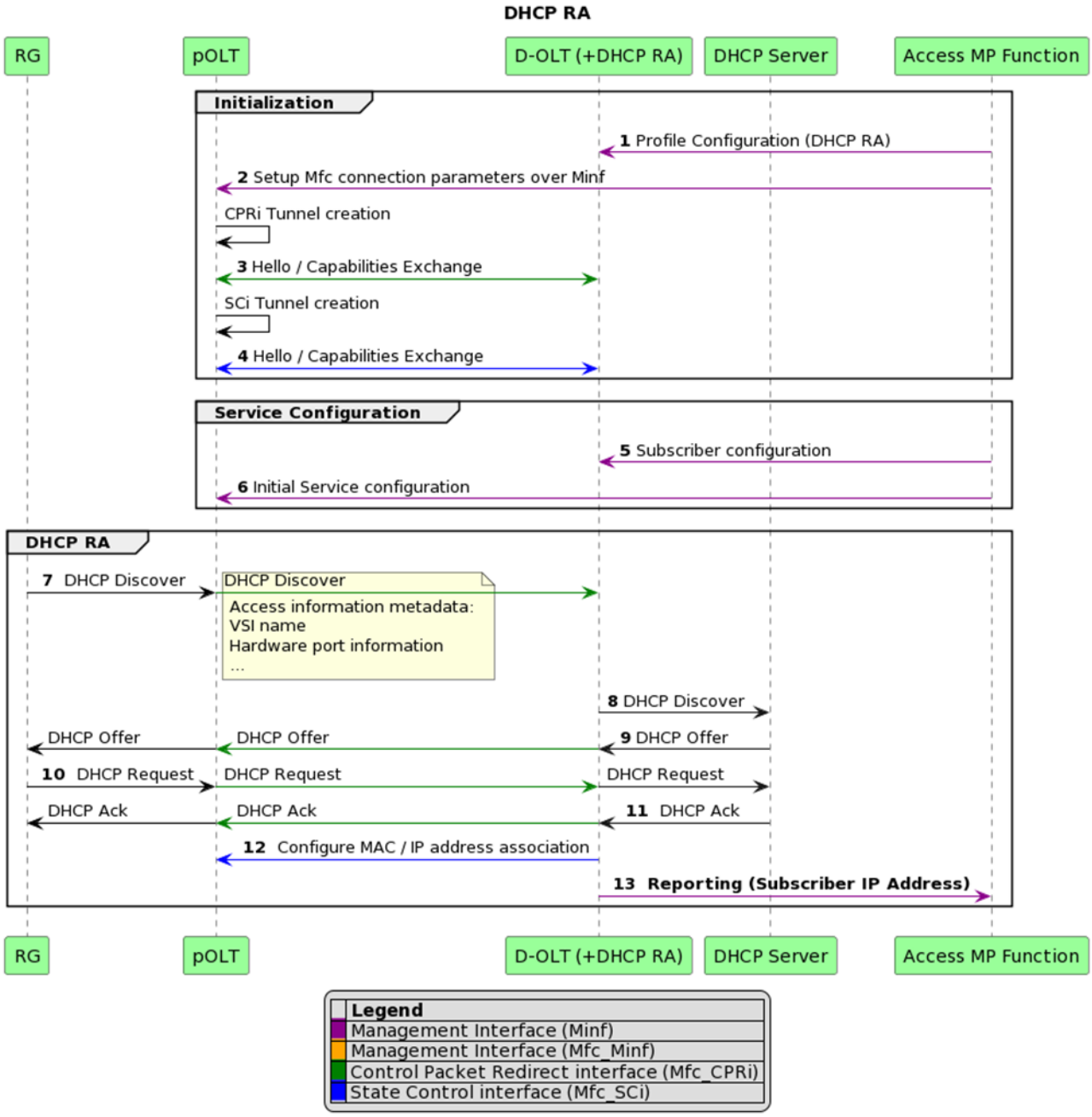

Figure 6 – DHCP RA call flow

3. After the Mfc_CPRI tunnel creation the pOLT and D-OLT exchange Hello messages that contain capabilities information.
4. After the Mfc_SCi tunnel creation the pOLT and D-OLT exchange Hello messages that contain capabilities information.
5. Using the Minf interface, the Access MP Function configures subscriber information in the DHCP RA function of the D-OLT (if not already done in step 1). This information is used by the DHCP RA function to correlate the DHCP packet to a subscriber and append the appropriate DHCP options such as option 82 for DHCPv4 and options 18 and 37 for DHCPv6.
6. Using the Minf interface, the Access MP Function configures the service in the pOLT so that packets can flow from the RG to the pOLT. This includes VLAN, T-CONT and GEM port information. The ONU side configurations can be pushed directly by the pOLT using an embedded OMCI stack or with the help of a disaggregated vOMCI function. The requests sent to the pOLT can be sent directly or it can be mediated by an adaptation layer if such is present and required (see section 4.1.2). Also, at this point, packet redirection is enabled on the service and the appropriate Mfc_CPRi endpoint is specified.
7. Subscriber devices (e.g., RG) that connect to the network send a “DHCP Discover” control message packet. The pOLT detects these control messages and tunnels them over the Mfc_CPRi interface.

   The pOLT can provide additional access interface information as metadata to the D-OLT such as the OLT UNI port and the VLAN Sub-Interface (VSI).
8. The DHCP RA within the D-OLT locally processes the incoming “DHCP Discover” to check its validity against the parameters defined in steps 1 and 5, and forwards the message to the DHCP Server after changing it (e.g., setting the DHCP Server unicast IP address, appending new options). This forward can happen via:
   a. Control Plane, using a unicast IP: in this case, the L3 DHCP RA unicasts the DISCOVER with the DHCP Server (as shown in Figure 6)
   b. Data Plane, using the forwarding context defined in step 6: in this case, the L2 DHCP RA modifies and forwards the DISCOVER packet back to the pOLT's upstream path via the Mfc-CPRi (not shown in Figure 6)
9. Upon successful authentication and authorization, the DHCP Server successfully replies to the D-OLT with a “DHCP Offer” with the IP address allocated to the RG. The “DHCP Offer” from the D-OLT is sent back to the RG through the pOLT utilizing the CPR Interface. This forward can happen via:
   a. Control Plane, using a unicast IP: in this case, the L3 DHCP RA directly receives the OFFER from the DHCP Server (as shown in Figure 6)
   b. Data Plane, using the forwarding context defined in step 6: in this case, the L2 DHCP RA modifies and forwards the OFFER packet back to the pOLT's downstream path via the Mfc-CPRi (not shown in Figure 6)
10. The “DHCP Request” is sent from the RG to the D-OLT through the pOLT, utilizing the dedicated Mfc_CPRi tunnel. This forward can happen via:
    a. Control Plane, using a unicast IP: in this case, the L3 DHCP RA unicasts the DISCOVER with the DHCP Server (as shown in Figure 6).
    b. Data Plane, using the forwarding context defined in step 6: in this case, the L2 DHCP RA modifies and forwards the REQUEST packet back to the pOLT's upstream path via the Mfc-CPRi (not shown in Figure 6).
11. The DHCP process is completed by sending the DHCP acknowledgement. The D-OLT forwards the “DHCP Ack” message through the dedicated Mfc_CPRi tunnel back to the RG though the pOLT. This forward can happen via:
    a. Control Plane, using a unicast IP: in this case, the L3 DHCP RA directly receives the ACK from the DHCP Server (as shown in Figure 6).

   b. Data Plane, using the forwarding context defined in step 6: in this case, the L2 DHCP RA modifies and forwards the ACK packet back to the pOLT's downstream path via the Mfc-CPRi (not shown in Figure 6).

12. Using the Mfc_SCi interface, the DHCP RA informs the pOLT of the IP address / MAC address association. This information can be used by the pOLT to enforce anti-spoofing functionality.
13. This IP address assigned to this subscriber is reported to the Access MP. Note: While not depicted in this message sequence, the DHCP Server or Edge SDN M&C can also report additional information such as the whether the subscriber is authenticated.

The RG has now been allocated an IP address, and user plane traffic associated to this HSIA service can now flow.

4.5.2 PPPoE IA call flow

The PPPoE call flow is made of four major processes:

- **PPPoE Discovery** – is the procedure started by the RG to obtain the MAC address used by the PPPoE NF and obtain a Session ID from it
- **PPP LCP Configuration** – is the procedure for determining the basic configuration for PPPoE connection
- **PPP Authentication** – is the procedure to perform user authentication for PPPoE connection. The method of user authentication is determined by process of LCP Configuration Protocol. For this workflow PAP and CHAP are used as Authentication Protocols
- **PPP IPCP Configuration** – is the procedure to obtain IP, gateway and DNS address for IP protocol

The pOLT and D-OLT provide the capabilities to relay PPPoE traffic between a subscriber device and the service providers' PPPoE server. Prior to providing the PPPoE IA capabilities procedure, the pOLT and D-OLT need an initialization procedure to the enable the functionality. This is required whenever the subscriber service is pre-provisioned. Subscriber services can be configured on a per subscriber basis. One aspect of the PPPoE IA functionality is that the subscriber's broadband service is considered pending until the PPPoE procedure is completed. In some cases, this PPPoE procedure is suspended until the subscriber has been identified and authenticated by the system.

1. The Access MP Function sends profile configuration to the PPPoE IA. In detail, the Access MP indicates to the D-OLT.

   Note; The Access MP function may configure the PPPoE IA parameters for one or more subscribers and can perform this step at any time. Typically, a single configuration is performed when all the subscriber's profiles can be pre-provisioned at the same time.

2. The Access MP Function configures parameters to initialize the communication between pOLT and D-OLT (e.g., endpoints IP addresses, ports, protocol) via the Minf interface.
3. After the Mfc_CPRI tunnel creation the pOLT and D-OLT exchange Hello messages that contain capabilities information.
4. Using the Minf interface, the Access MP Function configures subscriber information in the PPPoE IA function of the D-OLT (if not already done in step 1). This information is used by the PPPoE IA function to correlate the PPPoE IA packets to a subscriber.

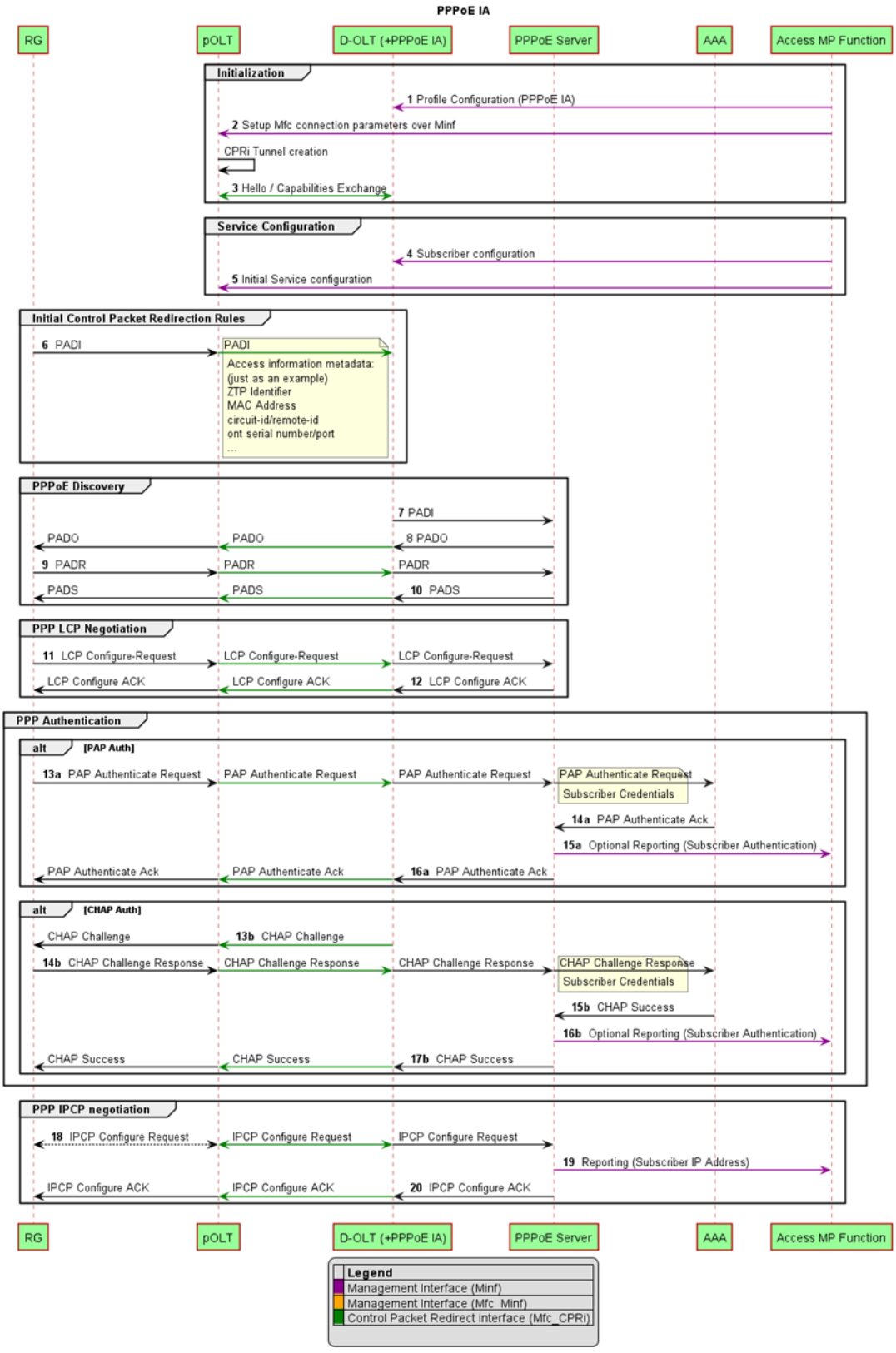

Figure 7 – PPPoE IA call flow

5. Using the Minf interface, the Access MP Function configures the service in the pOLT so that packets can flow from the RG to the pOLT. This includes VLAN, T-CONT and GEM port information. The ONU side configurations can be pushed directly by the pOLT using an embedded OMCI stack or with the help of a disaggregated vOMCI function. The requests sent to the pOLT can be sent directly or it can be mediated by an adaptation layer if such is present and required (see section 4.1.2). Also, at this point, packet redirection is enabled on the service and the appropriate Mfc_CPRi endpoint is specified.
6. Subscriber devices (e.g., RG) that connect to the network send a “PADI” control message packet. The pOLT detects these messages as new control ones that are unrelated to any known sessions and tunnels. Then they are transmitted over the Mfc_CPRi interface.

   The pOLT can provide additional access interface information as metadata (OLT UNI port information) to the D-OLT when the information is not present in the PPPoE packet.
7. The PPPoE IA within the D-OLT locally process the incoming “PADI” to check its validity against the parameters in step 1c and forward the message to the PPPoE Server after changing it (e.g., setting the PPPoE Server unicast IP address). This forward can happen via:
   - control plane, using an unicast IP
   - data plane, using a well-known port on pOLT (which is already configured in step 1)
8. The PPPoE Server successfully replies to the D-OLT with a “PPPoE Active Discover Offer (PADO)”. Using the Mfc_CPRi interface, the “PADO” message from the D-OLT is sent back to the RG through the pOLT.
9. Using the Mfc_CPRi interface, the “PPPoE Active Discovery Request (PADR)” message produced by the RG is sent from the RG to the D-OLT through the pOLT. The D-OLT then forwards the “PADR” message to the PPPoE Server.
10. The PPPoE discovery is completed by sending the “PPPoE Active Discovery Session-confirmation (PADS)”. The D-OLT sends the “PADS” back to the RG through the pOLT utilizing the Mfc_CPRi interface.
11. Using the Mfc_CPRi interface, the “LCP Configure-Request” is sent from the RG through the pOLT. The message indicates either a PAP or CHAP authentication challenge is used. The message is then forwarded by the D-OLT to the PPPoE Server.
12. Using the Mfc_CPRi interface, the D-OLT sends the “LCP Configure-Ack” from the PPPoE Server back to the RG through the pOLT.
13. After LCP negotiation is completed, the PPP authentication stage starts.
    a. If the RG chooses PAP, the RG sends a “PAP request” to the D-OLT through the pOLT utilizing the CPR Interface (13a). The credentials are included in the “PAP Authenticate Request” to the AAA server. The AAA successfully authenticates the RG and replies to the RG with “PAP Authenticate Ack” (14a/16a). As an optional step, the D-OLT could report the result of the subscriber authentication to the Access MP (15a).
    b. If CHAP is instead required, the D-OLT initiates a “CHAP Challenge” to the RG though the pOLT utilizing the CPR Interface (13b). The RG responds back to the challenge to the D-OLT. The challenge response is then sent to the AAA server (14b). The AAA successfully authenticates the RG and replies to the RG with a “CHAP Success” message (15b/17b). As an optional step, the D-OLT could report the result of the subscriber authentication to the Access MP (16b).
18. G sends via D-OLT “IPCP Configure-Request” to PPPoE server for parameter negotiation (e.g., RG IP address, Gateway address and mask), utilizing a dedicated session control packet redirect tunnel.
19. If the RG is allowed to obtain an IP address from the PPPoE Server, this IP address assigned to this subscriber is reported to the Access MP. Note: While not depicted in this message sequence, the PPPoE Server or Edge SDN M&C can also report additional information such as whether the subscriber is authenticated (optional steps 15a and 16b).

20. The PPPoE process is completed by sending the “IPCP Configure Ack”. The D-OLT forwards the “IPCP Configure Ack” through the dedicated Mfc_CPRi tunnel back to the RG through the pOLT.

IPCP enters the Opened state only after the “Configure Ack” messages are sent and received on both RG and D-OLT. At this step, the RG IP is assigned and data packets could be sent immediately.

4.5.3 Multicast call flow

The pOLT and D-OLT need an initialization procedure to enable the functionality. This initialization procedure requires setup of the configuration in the case of the subscriber service is pre-provisioned. Subscriber services can be configured on a per subscriber basis.

The D-OLT applies multicast policies to the pOLT (per subscriber basis) at the subscriber authentication event or during the subscriber session.

1. The Access MP Function sends profile configuration to the Multicast function of the D-OLT.

   Note: The Access MP function may configure the Multicast parameters for one or more subscribers and can perform this step at any time. Typically, a single configuration is performed when all the subscriber's profiles can be pre-provisioned at the same time.

2. The Access MP Function configures parameters to initialize the communication between pOLT and D-OLT (e.g., endpoints IP addresses, ports, protocol) via the Minf interface.
3. After the Mfc_CPRI tunnel creation the pOLT and D-OLT exchange Hello messages that contain capabilities information.
4. Using the Minf interface, the Access MP Function configures subscriber information in the Multicast function of the D-OLT (if not already done in step 1).
5. Using the Minf interface, the Access MP Function configures the service in the pOLT so that packets can flow from the RG to the pOLT. This includes VLAN, T-CONT and GEM port information. The ONU side configurations can be pushed directly by the pOLT using an embedded OMCI stack or with the help of a disaggregated vOMCI function. The requests sent to the pOLT can be sent directly or it can be mediated by an adaptation layer if such is present and required (see section 4.1.2). Also, at this point, packet redirection is enabled on the service and the appropriate Mfc_CPRi endpoint is specified.

MGMD Join:

6. To join a multicast stream after going online, an RG sends an “MGMD Report” message that results in an immediate “PIM Join” message to the pOLT for the multicast source.
7. The pOLT locally process the incoming “PIM Join” message to check its validity against the parameters in step 5. It identifies the subscriber and the multicast stream that the subscriber wants to join and creates a multicast forwarding entry for the RG.
8. At this time, the RG successfully joins the multicast group.
9. Using the Mfc_CPRi interface, the pOLT notifies the stream joined to the D-OLT, by sending a copy of the “MGMD Report” packet received from the RG. The D-OLT stores the list of streams joined by the subscriber for counting, charging, reporting, logging, etc.
10. The pOLT periodically sends an “MGMD Query” message to the RG.
11. Upon receipt of the MGMD Query message, the RG responds with an “MGMD Report” message to keep the multicast program active.

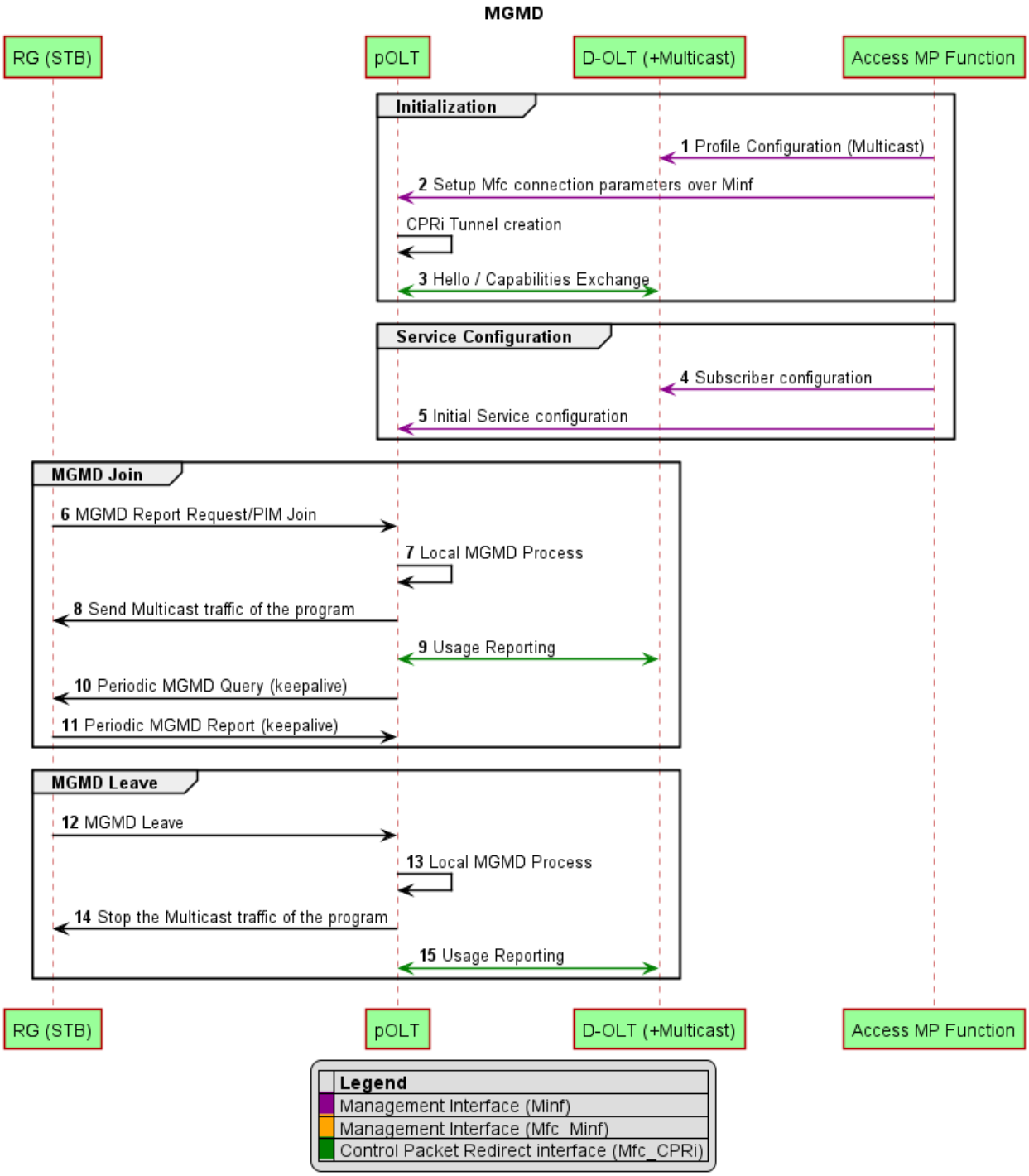

Figure 8 –Multicast call flow

MGMD Leave:

12. To leave a multicast stream, an RG sends an “MGMD Leave” message.
13. Upon receipt of the message, the pOLT identifies the user and the multicast stream that the user wants to leave and delete the multicast forwarding entry for the RG.

14. At this time, the pOLT stops sending to the RG the multicast traffic of the corresponding multicast group it joined.
15. Using the Mfc_CPRi interface, the pOLT notifies the D-OLT on the leaving of multicast stream by sending a copy of the MGMD Message received from the RG.

4.5.4 ONU Authentication call flow

Before the pOLT can provide services to the subscriber, the D-OLT authenticates the validity and identity of each ONU. The case reported below extends the Scenario 4 described in TR-489 [31], by adding the ONU Authentication function, which keeps direct authority for the authentication of ONU, built as a microservice of the D-OLT, as shown in Figure 9. ONU authentication will prevent access from an unauthorized ONU and it will allow the pOLT to retrieve the right configuration to be pushed to the ONU via OMCI (ONU binding).

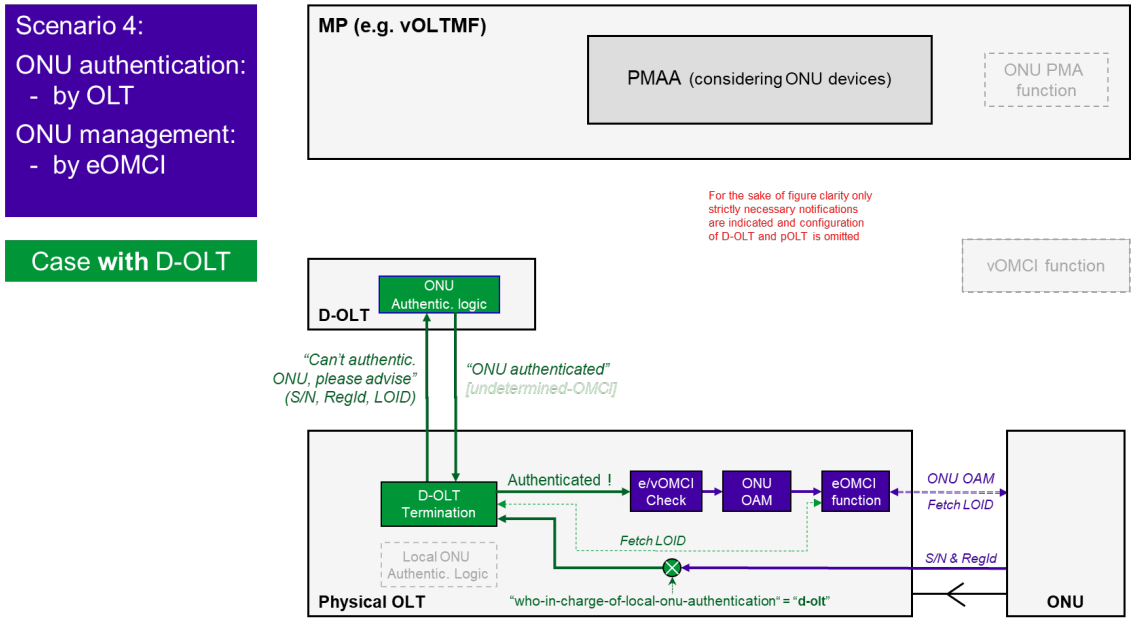

Figure 9 – ONU Authentication by OLT, ONU Management by eOMCI, case with D-OLT

Before an ONU goes online, it is authenticated through an expected authentication method. There are different authentication methods that depend on operational procedures at the discretion of the operator and the possible use case (e.g., in case of ONU failure with a replacement). The main methods (supported also by TR-385 [28] / TR-489 [31]) are:

- ONU Serial Number based
- ONU Registration-ID based
- Logical ONU ID (LOID) based

The ONU is authenticated by comparing its credentials at the activation time with the pre-configured ones for the same ONU device in the ONU Authentication function. Another option to consider is when the ONU goes online not pre-configured. Anyway, the subscriber's broadband service is considered pending until the ONU Authentication and the subsequent registration procedure are completed.

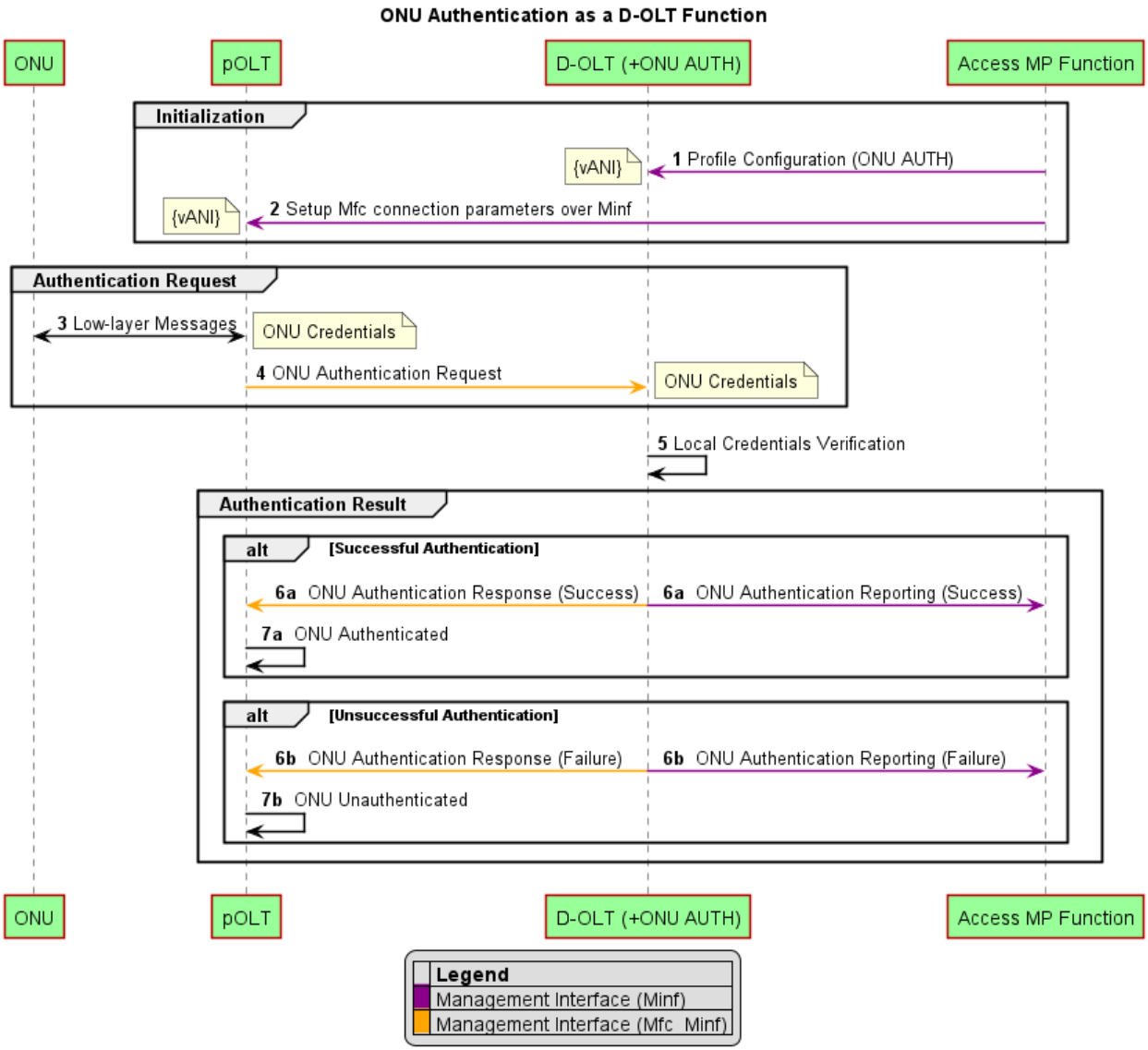

Figure 10 – ONU Authentication as a D-OLT function

1. The Access MP Function sends profile configuration to the D-OLT (+ ONU AUTH). In detail, the Access MP indicates to the D-OLT:
   a. which pOLT to control
   b. that ONU Authentication shall be enabled on the D-OLT
   c. The list of ONUs that the D-OLT must authenticate on behalf of pOLT. This list takes the form of the list of {vANIs} as known in pOLT, each containing, per regular TR-385 [28], the credentials required to authenticate the corresponding ONU.
2. The access MP function configures the pOLT with a list of {vANIs}. It also configures the pOLT that ONU authentication will be performed by D-OLT, together with parameters to initialize the communication between pOLT and D-OLT (e.g., endpoints IP addresses, ports, protocol) via the Minf interface.
3. When an ONU is plugged in, a low-layer exchange of messages is performed between pOLT and ONU. During these negotiations the ONU also reports its credentials to the pOLT.

4. The pOLT has the responsibility of passing the ONU authentication requests (credentials) towards the D-OLT, via the Mfc_Minf sub-interface, as per TR-385 [28] definitions. The authentication request message received from the pOLT includes the ONU credentials.
5. The D-OLT locally verifies the received credentials, according to the information it has stored.

The two possible results of the process are:

- **The ONU is successfully authenticated by the D-OLT**

  6a. The D-OLT reports the authentication status, as per TR-385 [28] definitions, to both the Access MP (via Minf) and the pOLT (via Mfc_Minf).

  7a. The pOLT sets the state of the ONU to “Authenticated”.

- **The pOLT is not able to authenticate the ONU, or it does not expect to authenticate it**

  6b. The D-OLT reports the authentication failure, as per TR-385 [28] definitions, to both the Access MP (via Minf) and the pOLT (via Mfc_Minf).

  7b. The pOLT sets the state of the ONU to "Unauthenticated".

4.5.5 Virtual DBA call flow

As described in Figure 8-1 of TR-402 [4], the DBA function implemented in hardware (physical DBA, PHY-DBA) is disaggregated into one or multiple vDBA functions that run as software components and an engine located in the pOLT, which is assumed to mainly process data at Data Plane level.

Each virtual DBA (vDBA) algorithm computes the virtual bandwidth map (vBMap) for the associated slice of the pOLT and delivers the calculated vBMap to the engine on the pOLT, which is responsible of allocating transmission slots to the ONUs. Multiple vDBA functions are expected to operate on the same PON port of the pOLT, as per TR-402 [4].

The vBMap indicates the desired position for each slot allocation within a slice, while the PHY-BMap is the allowed upstream capacity in the pOLT.

The vDBA algorithms operate as xNFs within the D-OLT to fulfill the requirements for T-CONT types with upstream capacity, strict latency and jitter limits. The pOLT instead takes care of framing and other data plane functions.

The pOLT and D-OLT need an initialization procedure to enable the operation of vDBA functions. The following descriptive steps consider two vDBA functions (vDBA1 and vDBA2), associated with two ONU devices (ONU1 and ONU2) connected to the same PON tree/OLT’s PON port. The reported example with 2 ONUs and 2 vDBA functions can be expanded to the generic case with n ONUs and m vDBA functions.

1. The Access MP Function sends profile configuration to the D-OLT (+ vDBA1&vDBA2). In details, the Access MP indicates to the D-OLT:
   a. which pOLT to control
   b. the DBA Profiles, PHY-BMap frame size, etc. associated to vDBA functions
2. The Access MP Function configures parameters to initialize the communication between pOLT and D-OLT (e.g., endpoints IP addresses, ports, protocol) via the Minf interface.
3. After the Mfc_CPRI tunnel creation the pOLT and D-OLT exchange Hello messages that contain capabilities information.
4. After the Mfc_SCi tunnel creation the pOLT and D-OLT exchange Hello messages that contain capabilities information.
5. Using the Minf interface, the Access MP Function configures subscribers information in the vDBA functions of the D-OLT (if not already done in step 1).

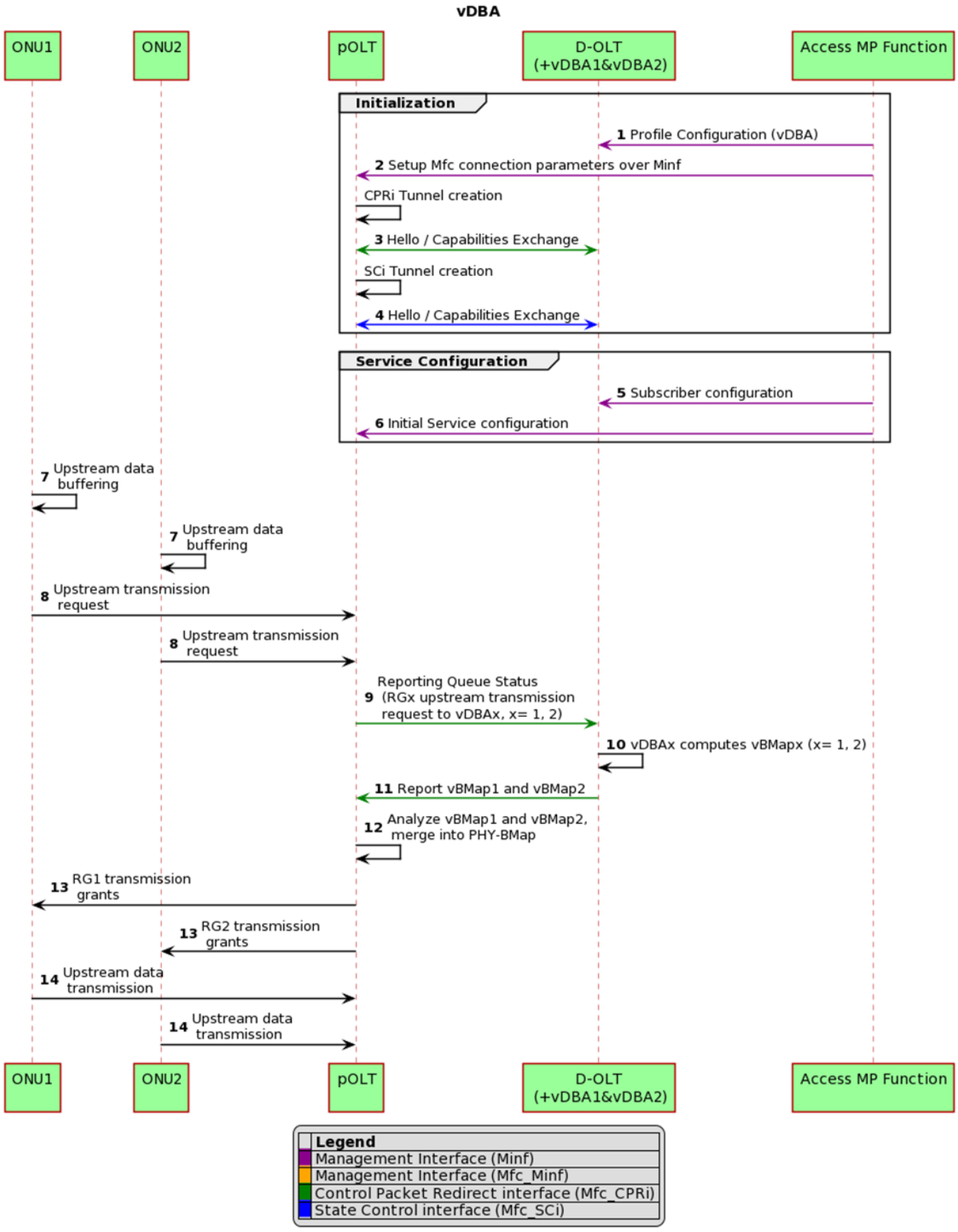

Figure 11 – Virtual DBA call flow

6. Using the Minf interface, the Access MP Function configures the service in the pOLT so that packets can flow from the ONUs to the pOLT. This includes VLANs, T-CONTs and GEM ports information. The ONUs side configurations can be pushed directly by the pOLT using an embedded OMCI stack or with the help of a disaggregated vOMCI function. The requests sent to the pOLT can be sent directly or it can be mediated by an adaptation layer if such is present and required (see section 4.1.2). Also, at this point, packet redirection is enabled on the service and the appropriate Mfc_CPRi endpoint is specified.
7. The ONUs perform buffering of upstream data while new transmission grants are awaited.
8. The ONUs report their queuing sizes (BufOcc as per TR-402 [4] or report frames from specific T-CONTs) to the pOLT as requests of upstream transmission grants.
9. The queue status report messages coming from the ONUs are relayed by the pOLT to the D-OLT.
10. The vDBA functions compute the virtual Bandwidth Maps (vBMaps) for the ONUs.
11. The vBMaps are delivered to the pOLT.
12. The vBmaps are analyzed and merged into a physical bandwidth map (PHY-BMap) applicable to RGs/ONUs at each frame interval.
13. The pOLT sends transmission grants to the RGs/ONUs as PHY-Bmap. This transmission happens at every frame interval (i.e., every 125μs).
14. The ONUs transmit the buffered upstream data to the pOLT per the allocated transmission grants.

4.5.6 vOMCI

The vOMCI solution defined in TR-451 (Figure 2), [25], is comprised by different functionalities (e.g., vOLTMF, vOMCI Function, etc.). When deployed in the context of the D-OLT, the vOMCI Function can be included within the D-OLT component, while the vOLTMF (vOLT Management Function) can be a feature of the Access SDN M&C, a BAA Layer, or an external function.

The vOMCI Function is here mainly responsible for:

- Receiving from the vOLTMF commands issued towards the target ONU(s).
- Translating the received management commands into OMCI management entities (ME) and formatting them into OMCI messages compliant with ITU-T G.988.
- Sending the management data and information generated from the received OMCI messages to the vOLTMF.

The vOMCI communicates with the pOLT via the MvOMCI-OLT interface (2) as specified in section 5.8 of TR-451, while management interface is via the MvOLTMF-vOMCI interface (1) as specified in section 5.7 of TR-451.

The vOMCI solution described in this Technical Report does not use a vOMCI Proxy nor ONU Management Proxy. In this case the D-OLT shall implement the interfaces described in Annex A:.

For additional information refer to TR-451, [25], section 5.7, section 5.8 and Appendix I.2.

5 Technical Requirements

5.1 pOLT functional requirements

[R-1] When combined with a D-OLT the pOLT MUST support the disaggregation of at least one or more functions defined in Table 1 of section 4.2.1.
[R-2] When combined with a D-OLT the pOLT MUST support the implementation options specified in section 4.2.2.
[R-3] When combined with a D-OLT the pOLT MUST interface with functions of at least one D-OLT.
[R-4] When combined with a D-OLT the pOLT MUST be capable of interfacing with functions of multiple D-OLTs.
[R-5] When combined with a D-OLT and for functions that have been disaggregated, the pOLT MUST support the call flows defined in section 4.5 for the disaggregated functions.

5.2 D-OLT functional requirements

[R-6] The D-OLT MUST support the disaggregation of at least one or more functions and services defined in Table 1 of section 4.2.1.
[R-7] The D-OLT MUST support the implementation options specified in section 4.2.2.
[R-8] The D-OLT MUST be capable of hosting disaggregated functions for multiple pOLTs.
[R-9] The D-OLT MUST support the sub-interfaces and protocols defined in Table 2 of section 4.2.3.
[R-10] For the disaggregated functions that are supported, the D-OLT MUST support the call flows defined in section 4.5.
[R-11] The D-OLT MUST support the Telemetry data pipeline requirements defined in TR-436 [19], Automated Intelligent Management, including:
- [R-9] The collector function and [R-77] pipeline orchestration
- Bulk, synchronous (request/response) and asynchronous (subscribe/notify) transfer modes listed in section 4.2.

5.3 Interfaces requirements

The section lists the requirements for the interfaces reported in Table 2.

[R-12] The pOLT MUST support the Minf interface towards the Access SDN M&C for FCAPS functionalities related to all the not-disaggregated functions, as specified in WT-413i2, [20].
[R-13] The pOLT MUST support the Mfc_Minf sub-interface towards the D-OLT for the disaggregated functions, as specified in WT-413i2, [20].
[R-14] The pOLT MUST support the Mfc_SCi sub-interface towards the D-OLT to send notifications of applied forwarding configurations and notifications and statistics of flow control events.
[R-15] The pOLT MUST implement the gRPC protocol for the Mfc_SCi sub-interface towards the D-OLT, as specified in section 6.2 of this Technical Report.
[R-16] The pOLT MUST support the Mfc_CPRi sub-interface towards the D-OLT to relay packets/information to an application endpoint in the D-OLT.
[R-17] The pOLT MUST implement the gRPC protocol for the Mfc_CPRi sub-interface towards the D-OLT, as specified in section 6.2 of this Technical Report.

[R-18] The pOLT MUST transfer information that identifies user interfaces (e.g., port, virtual-port, vlan-sub-interface, on which the control traffic is received) together with the tunneled control packet to the D-OLT.

[R-19] The D-OLT MUST support the Minf interface towards the Access SDN M&C entities for management purposes as specified in WT-413i2, [20], including the bbf-device-aggregation.yang model to ensure compatibility across implementations.

[R-20] The D-OLT MUST support the Mfc_Minf sub-interface towards the pOLT for pushing configurations and retrieving operational state and status to and from the pOLT, as specified in WT-413i2, [20].

[R-21] The D-OLT MUST support the Mfc_SCi sub-interface towards the pOLT to program default rules and policies for the control packet redirection between the control plane of D-OLT and user plane of the pOLT.

[R-22] The D-OLT MUST support the Mfc_CPRi sub-interface towards the pOLT, as specified in section 6.1.3 of this Technical Report, to forward and tunnel control packets between D-OLT and pOLT.

[R-23] The D-OLT MUST pass additional metadata to the tunneled control packets towards the pOLT in order to identify the origin of the relayed Control packet, as defined in the CpriMsg, section 6.1.3.2.1.

[R-24] The D-OLT MUST configure the pOLT to redirect only specific control packets per match criteria (e.g., subscribers connecting to the internet using PPPoE, multicast services, etc.) in order to filter out unnecessary control packets.

[R-25] The D-OLT SHOULD configure on the pOLT the priority of specific control messages per match criteria (e.g., control packets of authorized subscribers versus subscribers not authorized yet) in order to prioritize them.

[R-26] In the scenario where the D-OLT is hosted within a BAA Layer, the pOLT MAY be mounted via the Device Aggregation methodology and YANG model via the BAA Layer rather than directly to the Access SDN M&C. This is covered in section 4.1 of TR-484 [34].

6 Protocol specification

As reported in the section 4.2.3, the Mfc reference point provides control plane functionality and utilizes a number of control plane protocols for the various sub-interfaces of the reference point, reported in the following Table 2 in section 4.2.3.

6.1 Message Types

This section describes the type of messages exchanged by pOLT and D-OLT using the sub-interfaces of Table 2.

6.1.1 Mfc_Minf

To provide the capabilities described in section 4.5, the pOLT and D-OLT need an initialization procedure to enable these functionalities.

Using the Mfc_Minf sub-interface, the D-OLT makes the pOLT aware of:

- the D-OLT is the controller to use for the functionalities
- the information needed for their interaction and message exchange (e.g., IP address, port, protocol)

The Mfc_Minf shall support transactional configuration from the D-OLT to the pOLT based on YANG data model.

NOTE: The expectation here is to make use of the already implemented YANG model from BBF and other organizations.

6.1.1.1 Information Messages

The Mfc_Minf shall support transactional configuration from the D-OLT to the pOLT based on YANG data model, encoded as XML and transmitted via NETCONF, [14].

6.1.2 Mfc_SCi

The Mfc_SCi sub-interface transfers YANG based messages between the D-OLT and the pOLT to disaggregated functions associated with the D-OLT to control the pOLT. Similarly, the Mfc_SCi sub-interface is used to receive notifications and operational data from the pOLT that is needed by the disaggregated functions associated with the D-OLT.

The YANG based messages that are conveyed across the Mfc_SCi sub-interface are serialized in Google Protocol Buffers (GPB) and then transferred using the Google RPC (gRPC) message transfer protocol. The YANG models are encoded in JSON using RFC 7951 [30] encoding.

6.1.2.1 Information Model and Messages

The information conveyed on the Mfc_SCi sub-interface includes data model messages that are targeted to the pOLT from the pOLT.

The messages used on the interface are comprised of a header and a body. The information in the header is common to all messages and each message within the body has its own set of attributes that comprise the request, response, or notification.

```
message Msg {
  Header header = 1;
  Body body = 2;
}
```

Figure 12 – Mfc_SCi Message

6.1.2.1.1 Message Header

The message header is comprised of 3 fields which are optional in the protocol.

- The *msg_id* field is an optional used by the D-OLT to correlate message requests with responses. Likewise, the pOLT can use the msg_id field to identify notifications sent by these entities.
- The *sender_name* field is an optional field that provides the unique name of the entity that originated the request, response, or notification (i.e., D-OLT, pOLT).
- The *recipient_name* field is an optional field that provides the unique name of the entity that is to receive the request, response or notification (i.e., D-OLT, pOLT). The *recipient_name* field is used in certain message transfer protocols like Kafka to verify the intended recipient of the message.

The *msg_id*, *sender_name* and *recipient_name* fields are useful for logging and troubleshooting purposes as they can help identify interactions between the D-OLT and pOLT.

```
message Header {
  string msg_id = 1;          // Message identifier to
                              // 1) Identify requests and notifications
                              // 2) Correlate requests and response
  string sender_name = 2;     // Unique name of the entity that
                              // originated the message
  string recipient_name = 3; // The name of the entity that is to
                              // receive the request
}
```

Figure 13 – Mfc_SCi Header Message

6.1.2.1.2 Message Body

The body of the SCi message can be a *request*, *response* or *notification* message. *Request* messages originate from the D-OLT while *response* and *notification* messages originate from the pOLT.

```
message Body {
  oneof msg_body {
    Request request = 1;
    Response response = 2;
    Notification notification = 3;
  }
}
```

Figure 14 – Mfc_SCi Message Body

6.1.2.1.2.1 Request and Response Messages

Request messages are originated from the D-OLT are directed to pOLT. *Response* messages are originated from the pOLT are directed to the D-OLT that originated the *request* message.

```
message Request {
  oneof req_type {
    Hello hello = 1;
    GetData get_data = 2;
    UpdateConfig update_config = 3;
    RPC rpc = 4;
    Action action = 5;
  }
}

message Response {
  oneof resp_type {
    HelloResp hello_resp = 1;
    GetDataResp get_resp = 2;
    UpdateConfigResp update_config_resp = 3;
    RPCResp rpc_resp = 4;
    ActionResp action_resp = 5;
  }
}
```

Figure 15 – Mfc_SCi Request and Response Messages

6.1.2.1.2.1.1 Hello Message

This *Hello request* message is originated by the originating entiry (D-OLT, pOLT) when the entity establishes a connection to the peer entity. The *service_endpoint_name* field is used to identify the local service endpoint where the D-OLT can be reached. The *Hello response* message is used by the pOLT to identify the service_endpoint that the D-OLT used to send the *Hello request* message. Additionally, the *Hello response* message contains capabilities of the server.

```
  message Hello {
    string service_endpoint_name = 1; //The service endpoint the client
                                      // used to establish the session
    repeated ClientCapability capabilities = 2; // The capabilities
                                                // supported by the
                                                // client
    enum ClientCapability  {
      NO_CAPABILITY_REPORTED = 0;
    }
  }

  message HelloResp {
    string service_endpoint_name = 1; //The service endpoint the server
                                      //used to listen on the session
    repeated ServerCapability capabilities = 2; // The capabilities
                                                // supported by the
                                                // server
    enum ServerCapability  {
      NO_CAPABILITY_REPORTED = 0;
    }
}
```

Figure 16 – Mfc_SCi Hello Request and Response Messages

6.1.2.1.2.1.2 Get Data Message

The *GetData* message is used to retrieve the configuration and state information for the receiver of the *request* message.

The *filter* field in the *GetData request* message is used to scope the request to specific elements. Each filter instance contains an expression that represents a NETCONF [14] subtree filter.

The *status_response* field in the *GetData response* message is used to convey the status of the message along with any error codes if the request failed for any reason.

```
message GetData {
  repeated bytes filter = 1;
}

message GetDataResp {
  Status status_resp = 1;
  bytes data = 2;
}
```

Figure 17 – Mfc_SCi GetData Request and Response Messages

6.1.2.1.2.1.3 Update Configuration Message

The *UpdateConfig* message is used to update requested part of the configuration of the receiver of the *request* message. The pOLT uses the list of changes for nodes within the pOLT's configuration contained within the *delta-config* field.

The *status_response* field in the *UpdateConfigResp response* message is used to convey the status of the message along with any error codes if the request failed for any reason.

```
message UpdateConfig {
  bytes delta_config = 1;          // List of Node changes with the
                                   // associated operation to apply to
                                   // the node
}

message UpdateConfigResp {
  Status status_resp = 1;
}
```

Figure 18 – Mfc_SCi UpdateConfig Request and Response Messages

6.1.2.1.2.1.4 RPC and Action Messages

The *RPC* and *Action* messages permits operations to executed for the entity identified by the *receiver of the request* message. The *input-data* and *output-data* fields contains the instances of YANG elements that are defined for associated YANG RPC and Action.

The *status_response* field in the *Action and RPC response* messages is used to convey the status of the message along with any error codes if the request failed for any reason.

```
message RPC {
  bytes input_data = 1;
}

message RPCResp {
  Status status_resp = 1;
  bytes output_data = 2;
}

message Action {
  bytes input_data = 1;
}

message ActionResp {
  Status status_resp = 1;
  bytes output_data = 2;
}
```

Figure 19 – Mfc_SCi RPC and Action Request and Response Messages

6.1.2.1.2.1.5 Status and Error Messages

The *Status* message permits pOLT to respond to *request* messages. The *status_code* field in the message indicates of the request was successful or if there was an error. If there was an error with the request, the Status message contains one or more *Error* messages in the *error* field of the *Status* message. The Error message is based on NETCONF rpcErrorType defined in Appendix B of RFC 6241 [14]

```
message Status {
  enum StatusCode  {
    OK = 0;
    ERROR_GENERAL = 1;
  }
  StatusCode status_code = 1;
  repeated Error error = 2; //Optional: Error information
}
```

Figure 20 – Mfc_SCi Status Message

The following JSON fragment is an equivalent representation for the first example shown in the Section 4.3 of RFC 6241 [14] and would be encoded in the error field of the *Status* message.

Note: Instead of the "rpc-error" field, the root element is named "error".

```
"error": {
       "error-type": "rpc",
       "error-tag": "missing-attribute",
       "error-severity": "error",
       "error-info": {
              "bad-attribute": "message-id",
              "bad-element": "rpc"
       }
}
```

Figure 21 – Mfc_SCi Error Response Message

6.1.2.1.2.1.6 Notification Messages

The *Notification* message permits pOLT to send notification events to the D-OLT. The YANG notification is contained in the *data* field of the message.

```
message Notification {
  bytes data = 1;
}
```

Figure 22 – Mfc_SCi Notification Message

6.1.2.2 D-OLT Message Service

The Mfc_SCi sub-interface uses a message service (*SciService*) which provides the RPCs needed to exchange GPB encoded Mfc_SCi messages between the pOLT and D-OLT using gRPC to transfer the messages.

Since pOLT and D-OLT entities require a gRPC stream in order to send Mfc_SCi messages, one of the instantiates the stream by sending the *ListenForSciRx* message to the peer entity, which the peer entity then uses to send any messages as response on the stream.

```
service SciMessage {
    rpc ListenForSciRx (google.protobuf.Empty) returns (stream
tr477_sci_message.v1.Msg);
    rpc SciTx (tr477_sci_message.v1.Msg) returns (google.protobuf.Empty);
}
```

Figure 23 – Mfc_SCi Message Service

6.1.2.3 Requirements

[R-27] A pOLT and D-OLT implementation using Google protocol buffers encoding to encode messages across the Mfc_Sci sub-interface MUST conform to the schema defined in section 6.1.2.1.

[R-28] A pOLT and D-OLT implementation that uses gRPC to transfer Google protocol buffers encoded messages MUST conform to the Google Protobuf services defined in section 6.1.2.2.

6.1.3 Mfc_CPRi

The Mfc_CPRi sub-interface is used as a separate interface (tunnel) to forward control packets (e.g., DHCP, PPPoE, etc.) through the pOLT to the D-OLT (Figure 1).

6.1.3.1 Messages Encoding

The control packets related to a subscriber session are serialized via Google Protocol Buffers (GPB) and conveyed in the Mfc_CPRi.

The Mfc_CPRi is thus used as a prioritized tunnel for control packet exchange between the pOLT and the D-OLT.

6.1.3.2 Messages

6.1.3.2.1 Mfc_CPRI Message

Control packets are forwarded from the pOLT to the D-OLT using the Mfc_CPRi Message (CpriMsg) from the protobuf definition.

```
message CpriMsgHeader {
  string msg_id = 1;  // Message identifier to
                      // 1) Identify requests and notifications
}

message GenericMetadata {
  string device_name = 1;       //The name of the device where the packet was
                                //intercepted/is destined to
  string device_interface = 2;  //The name of the interface where the packet
                                //was intercepted/is destined to
  string originating_rule = 3;  //Optional identification of the rule that originated
the
                                //capture of the packet
}

message OnuMetadata {
  string  channel_termination = 1; //The local name of the channel termination
                                   //in the OLT where the ONU is attached
  string  onu_id = 2;              //ITU-T Transmission Convergence (TC)
                                   //layer ONU-ID
}

message DhcpMetadata {

}

message PppoeMetadata {

}


message CpriMetaData {
  GenericMetadata generic = 1;  //Common generic metadata
  OnuMetadata onu = 2;          //Optional information about the ONU that
                                //originated the packet or is destination
                                //for the packet
  oneof specific_metadata {
    DhcpMetadata dhcp = 3;
```

```
    PppoeMetadata pppoe = 4;
  }
}

message CpriMsg {
  CpriMsgHeader header = 1;
  CpriMetaData meta_data = 2;
  bytes packet = 3;            //Unmodified contents of the packet
}
```

Figure 24 – Mfc_CPRi Message

In Figure 24 the *CpriMsg* contains the raw Control packet bytes with additional metadata that helps the D-OLT or SDN M&C to identify the origin of the relayed Control packet.

The Mfc_CPRi channel is bidirectional. When the *CpriMsg* is sent towards the D-OLT or SDN M&C, the *device_name* field of *GenericMetadata* contains the identification of the pOLT that originated the packet and the device_interface field contains the identification of the interface where the packet was captured.

When the *CpriMsg* is sent towards the pOLT, the *device_name* contains the identification of the destination pOLT and the *device_interface* contains the identification of the interface where the packet is to be transmitted.

The *originating_rule* can be used by the D-OLT or SDN M&C to correlate the Control packet to a particular data flow.

6.1.3.3 Message Service

The Mfc_CPRi sub-interface uses a message service (*CpriMessage*) which provides the RPCs needed to exchange GPB encoded Mfc_CPRi messages between the pOLT and D-OLT using gRPC to transfer the messages.

Since pOLT and D-OLT entities require a gRPC stream in order to send Mfc_CPRi messages, one of them instantiates the stream by sending the *ListenForCpriRx* message to the peer entity, which the peer entity then uses to send any messages as response on the stream.

```
service CpriMessage {
    rpc ListenForCpriRx (google.protobuf.Empty) returns (stream
tr477_cpri_message.v1.CpriMsg);
    rpc CpriTx (tr477_cpri_message.v1.CpriMsg) returns (google.protobuf.Empty);
}
```

Figure 25 – Mfc_CPRi Message Service

6.2 Use of gRPC to Exchange GPB Encapsulated Mfc_SCi and Mfc_CPRi Messages

Mfc_SCi and Mfc_CPRi messages that have been encoded using Google Protobufs as described in section 6.1.2 and section 6.1.3 of this Technical Report are transferred between the pOLT and D-OLT using Google Remote Procedure Call (gRPC).

6.2.1 gRPC Channel Initiation

gRPC channels between the pOLT and D-OLT can be initiated by any of the pOLT or D-OLT entities, depending on capabilities (i.e., gRPC client, gRPC server) provided by the pOLT and D-OLT. pOLT, D-OLT entities that initiate the gRPC channel supports the gRPC client interface while entities that support the reception of the initiation of the gRPC channel supports the gRPC server interface. pOLT, and D-OLT entities can simultaneously support both the gRPC client and gRPC server interfaces. Regardless of which entity initiates the gRPC channel, since a pOLT and D-OLT can initiate messages, the asynchronous, bi-directional streaming RPC method is used to exchange messages. When either the pOLT or D-OLT initiates the gRPC channel, the initiating entity has to send the Hello and Listen RPCs.

Note: In the scenario where both pOLT and D-OLT function entities implements the gRPC client and gRPC server interfaces, the resulting solution will establish two (2) gRPC channel in each direction resulting in two (2) HTTP/2 and associated TCP connections.

6.2.2 Remote Entity Contact Information

When either the pOLT or D-OLT requests to establish a gRPC channel with its peer, the initiating entity has to have the information necessary to contact the peer entity. This information includes:

- URL of the peer entity (e.g., host name, port)
- Security credentials necessary to establish the identity of the initiating entity
- Security credentials necessary to verify the identity of the peer entity

6.2.2.1 OLT, D-OLT Identification and Configuration

Based on the capabilities of pOLT and D-OLT, the pOLT or D-OLT function can establish gRPC channels with multiple instances of peer entities. The determination of peer entity instance(s) that an initiating entity establishes a channel is configured through the provisioning interface of the initiating entity.

6.2.3 gRPC Channel Maintenance

gRPC channels between the pOLT and D-OLT are considered persistent connections where if the initiation attempt fails or connectivity of the established gRPC channel fails, the gRPC client attempts to reconnect using a reconnection strategy. Likewise, to ensure that a connection is healthy the gRPC client periodically initiates ping requests to ensure the health of the gRPC channel. Likewise, the interval for the HTTP/2 Ping frames that the sent from the gRPC client to server is configurable.

6.2.4 Using HTTP/2 as the gRPC Wire Protocol

The most common wire protocol used by gRPC is HTTP/2 where many of the attributes of the persistent connection uses underlying features provided by HTTP/2 as defined in RFC 7540 [21] (e.g., keep-alive). The gRPC to HTTP/2 bindings include the values for the request and response headers along with trailers for the HTTP/2 response and are defined by the gRPC specification.

For gRPC requests implementations are able to define and adjust the request timeouts (i.e., grpc-timeout) and if compression is used (grpc-encoding).

When using HTTP/2 as the gRPC wire protocol, HTTP/2 servers can send a GOAWAY frame to the HTTP/2 client indicating that HTTP/2 server will no longer accept connections. When this occurs the entity, acting as the gRPC and HTTP/2 client will not attempt to re-connect to the peer entity and will provide a notification or log entry that the initiating entity cannot establish communications to the peer entity.

6.2.5 Securing the gRPC Channel

The gRPC channel between the peer entities is secured using TLS version 1.2 [22] or higher as required by the gRPC specification. This Technical Report uses the TLS session to provide confidentiality and integrity protection of the gRPC channel along with authentication of the entities. The determination of the authenticity of the entities is provided through the exchange and validation of X.509 certificates.

6.2.6 Requirements

[R-29] When using gRPC to transfer messages, a pOLT and D-OLT SHOULD provide the capability to initiate the gRPC channel establishment procedure acting as a gRPC client.

[R-30] When using gRPC to transfer messages, a pOLT and D-OLT SHOULD provide gRPC server capabilities.

[R-31] When establishing a gRPC channel between the pOLT and D-OLT, the Bi-directional streaming RPC method MUST be used to exchange messages.

[R-32] When establishing a gRPC channel between the pOLT and D-OLT, the initiating entity MUST send the Hello RPC in order to exchange host names and endpoint information needed to exchange messages.

[R-33] When establishing a gRPC channel between the pOLT and D-OLT, the initiating entity MUST send the ListenForSciRx and ListenForCpriRx RPC in order to allow the peer entity to send GPB messages.

[R-34] When the pOLT and D-OLT acts as a gRPC client, the entity MUST provide the capability to establish gRPC channels to one (1) or more remote entities using the peer entity's contact information defined in section 6.2.2 of this Technical Report.

[R-35] The gRPC channel established between the pOLT and D-OLT MUST be a persistent connection where if the initiation attempt fails or the connectivity of the established gRPC fails, the gRPC client attempts to reconnect using the reconnection strategy defined by the gRPC specification.

[R-36] When the pOLT and D-OLT cannot establish a gRPC channel with a remote entity, the initiating entity MUST provide a notification that communication with the remote entity cannot be established along with a possible reason (e.g., HTTP/2 GOAWAY frame).

[R-37] For established gRPC channels, the initiating pOLT and D-OLT MUST ping the remote entity on a periodic interval. The ping interval MUST be configurable and default value MUST be 5 minutes.

[R-38] When sending gRPC requests, the requesting pOLT and D-OLT MUST provide a timeout for the request and indicate if compression is used for encoding. The gRPC request timeout and compression

values MUST be configurable. The default value for the grpc-timeout MUST be 30 seconds and the default value for the grpc-encoding MUST NOT include compression.

[R-39] Implementations MUST ensure the confidentiality, integrity and authenticity of the gRPC channels when exchanging messages between the pOLT and D-OLT.

[R-40] The pOLT and D-OLT endpoints MUST implement TLS 1.2 as described in RFC 5246.

[R-41] When an initiating pOLT and D-OLT, acting as the gRPC client, receives the remote entities' X.509 certificate while establishing TLS session, the initiating entity MUST identify the remote entity according to section 6 of RFC 6125.

[R-42] The pOLT and D-OLT, acting as a gRPC server, MUST authenticate the initiating entity using the X.509 certificate presented by the initiating entity according to section 6 of RFC 6125.

[R-43] The Mfc_Minf MUST support transactional configuration, as defined in TR-435 [15] from the D-OLT to the pOLT.

Annex A: D-OLT YANG Modules

This section describes the YANG modules to be used in the context of a D-OLT deployment for the purposes of this Technical Report.

A.1 Dependencies on related Yang modules and standards

The YANG modules in this Technical Report are based on YANG 1.1 (RFC 7950 [16]) and are used for:

- Establish gRPC endpoints for the pOLT and the D-OLT functionalities
- Configuration and state information for the pOLT and D-OLT to identify the remote endpoints toward the pOLT and the D-OLT functionalities
  - Maintain the metadata for the configuration (e.g., ONU, protocols, CPRi, etc.)
  - Notifications:
    - for the discovered ONUs
    - for the results of ONU authentications (Authenticated/Unauthenticated)
  - Configuration and state for the association policy between pOLT and D-OLT functions instances

This Technical Report uses the YANG modules defined in the following sections.

A.1.1 Minf Interface

The relationships between the functional components described in this Technical Report and the YANG modules related to the Minf Interface are included in the following Table 3.

Table 3 – pOLT / D-OLT YANG Modules for Minf interface

Function	YANG Module(s)	Source	Integrations/Gaps
pOLT	$M_{inf\text{-}OLT}$ yang modules	WT-413i2	
	bbf-olt-vomci.yang	TR-451	
	bbf.olt.vomci.state.yang	TR-451	
	bbf-olt-d-olt.yang	TR-451	
D-OLT (Management Functions)	bbf-device-aggregation.yang	TR-477	
	bbf-d-olt-network-function-types.yang	TR-477	
	bbf-vomci-function.yang	TR-451	
	ietf-netconf-notifications.yang	RFC 6470	
	bbf-d-olt-dhcpra.yang	TR-477	
	bbf-d-olt-pppoe-ia.yang	TR-477	
	bbf-mgmd-mrd.yang	TR-383	
	bbf-xpon-onu-authentication.yang	TR-385	

Table 4 – pOLT / D-OLT YANG Modules for Mfc_Minf interface

Function	YANG Module(s)	Source	Integrations/Gaps
pOLT	bbf-grpc-client.yang	TR-451	
	bbf-xponvani.yang	TR-385	
	bbf-sub-interfaces.yang	TR-383	
	ietf-interfaces.yang	RFC 7223	
	ietf-netconf-notifications.yang	RFC 6470	
	bbf-l2-forwarding*.yang	TR-383	
	bbf-subscriber-profiles.yang	TR-383	
	bbf-mgmd-types.yang	TR-383	
	bbf-mgmd-mrd.yang	TR-383	
	bbf-mgmd-configuration-*.yang	TR-383	
	bbf-mgmd-operational-interface-*.yang	TR-383	
D-OLT (Management Functions)	bbf-grpc-client.yang	TR-451	
	bbf-network-function-client.yang	TR-451	
	bbf-network-function-server.yang	TR-451	
	bbf-d-olt-network-function-types.yang	TR-477	
	bbf-olt-vomci.yang**Error! Bookmark not defined.**	TR-451	
	ietf-netconf-notifications.yang	RFC 6470	
D-OLT (DHCP Relay)	bbf-d-olt-dhcpra-client-server.yang	TR-477	
	bbf-d-olt-dhcpra-client-server-tcp.yang	TR-477	
D-OLT (PPPoE IA)	bbf-d-olt-pppoe-ia-client-server.yang	TR-477	
	bbf-d-olt-pppoe-ia-client-server-tcp.yang	TR-477	
D-OLT (Multicast)	bbf-mgmd-*.yang	TR-383	
D-OLT (ONU Auth)	bbf-xpon-onu-authentication.yang	TR-385	
D-OLT (vDBA)	bbf-d-olt-vdba.yang	TR-403	
D-OLT (vOMCI)	vOMCI YANG modules	TR-451	

A.1.2 Mfc_Minf Interface

The relationships between the functional components described in this Technical Report and the YANG modules related to the Mfc_Minf Interface are included in the following Table 4.

A.1.3 Mfc_SCi Interface

The relationships between the functional components described in this Technical Report and the YANG modules related to the Mfc_SCi Interface are included in the following Table 5.

Table 5 – pOLT / D-OLT YANG Modules for Mfc_SCi interface

Function	YANG Module(s)	Source	Integrations/Gaps
pOLT	bbf-network-function-state.yang	TR-383	
	bbf-network-function-capabilities.yang	TR-383	
D-OLT (Management Function)	bbf-network-function-client.yang	TR-451	
	bbf-network-function-server.yang	TR-451	
D-OLT (DHCP Relay)	bbf-l2-dhcpv4-relay.yang	TR-383	
D-OLT (PPPoE IA)	bbf-pppoe-intermediate-agent.yang	TR-383	
D-OLT (Multicast)	bbf-mgmd-*.yang	TR-383	
D-OLT (vDBA)	bbf-vdba.yang	TR-403	
D-OLT (vOMCI)	vOMCI YANG modules	TR-451	

Annex B: Protobuf Files

This section reports the Protobuf files defined for Mfc_SCi and Mfc_CPRi interfaces in section 6 that are used to establish the gRPC channel between the D-OLT and pOLT. Table 6 – reports the Protobuf files for the Mfc_SCi interface, while Table 7 – the Protobuf files for Mfc_CPRi interface.

Table 6 – pOLT / D-OLT Protobuf files for Mfc_SCi interface

Interface	Protobuf File(s)	Source	Integrations/Gaps
Mfc_SCi	tr477_sci_message.proto	TR-477	
	tr477_sci_service.proto	TR-477	

Table 7 – pOLT / D-OLT Protobuf Files for Mfc_CPRi interface

Interface	Protobuf File(s)	Source	Integrations/Gaps
Mfc_CPRi	tr477_cpri_message.proto	TR-477	
	tr477_cpri_service.proto	TR-477	

Appendix I. Whitebox OLT

The concept of white-box design applies originally to switches and routers.

White-boxes and the SDN paradigm shift moved their first footsteps from campus applications and naturally spread out for boosting performances in Data Centers and branch office applications, as well as greenfield deployments.

Market volumes have been pushed by widespread white-box switch/router deployments in hyperscalers' Data Centers. Others that have deployed white-box switches with branded NOS software include large Communications Service Providers (CSPs) and Tier 2 cloud providers.

In the access domain all (or most) of those concepts have been applied to, for example, what is called a white box OLT, in that this is an Access Node with switching capabilities.

As a matter of fact, for more than two decades of fixed broadband deployments worldwide, ANs with only L2 features have been adopted.

The only difference is that the tributary interfaces that connect to the access network are technology specific, so to control a white box OLT as a commodity switch, driven by a pure forwarding protocol, requires an abstraction of the technology specific features.

Aggregation switches connected to the 'commodity switch' OLT itself should also be implemented as open, modular white box switch layers to fully meet the needs of white box flexibility.

The following characteristics have been identified for a white-box OLT, as system design:

- Based on merchant chipsets and/or isolated, per-port OLT MACs and Reference Designs which can fit increasing variety of environments, including: 1RU CO slot, pizza box, modular type (e.g. SFP+ form factor), outdoor/hardened environments, cell site backhaul, and others.
- Supporting general purpose processors, open installation and execution environments, independent Network Operating System (NOS)
- Clear separation of Ethernet switch and PON domains, both open to control-plane abstraction
- L2 and L3 capabilities that can be broadly repurposed, as on board computing and memory allow
- Highly programmable (e.g. via OF or P4 protocols) forwarding plane and control packets relay strategies
- PON MAC chipset interface exposed northbound to directly manage the PHY layer on a per-port basis or with native commands and a lightweight local protocol (gRPC)
- All management and control functionalities that are not latency sensitive are moved out-of-the-box to a Network Controller

Though not strictly defining a white-box OLT, the related engineering and deployment strategies rely on these architectural aspects to meet efficiency and operational objectives:

- Distributed and flexibly scalable aggregation and switching infrastructure (e.g. a Leaf Spine or other switch fabric architecture)
- Cross-domain Network Controller governing flow switching and routing across the access, aggregation and edge domains

End of Broadband Forum Technical Report TR-477