Residential CO Re-Architected Datacenter, Services, platforms, architecture and use cases
MINT 709 – Master in internetworking

Ahmed Milad Elragaiee
University of Alberta

To my family, Eman, Elias, Yamamah, Adam and Maria.

Table of Contents

Section 1: FTTx technologies overview

Chapter 1 Introduction

Majority of Telecom provider's network comprised of a backbone, regional/metro and access network. The backbone (sometimes called core) and regional network's structure are usually alike, on the other hand access networks will take different architecture depends mainly on provided services and customers' applications [1]. Figure 1-1 shows general Telecom network structure.

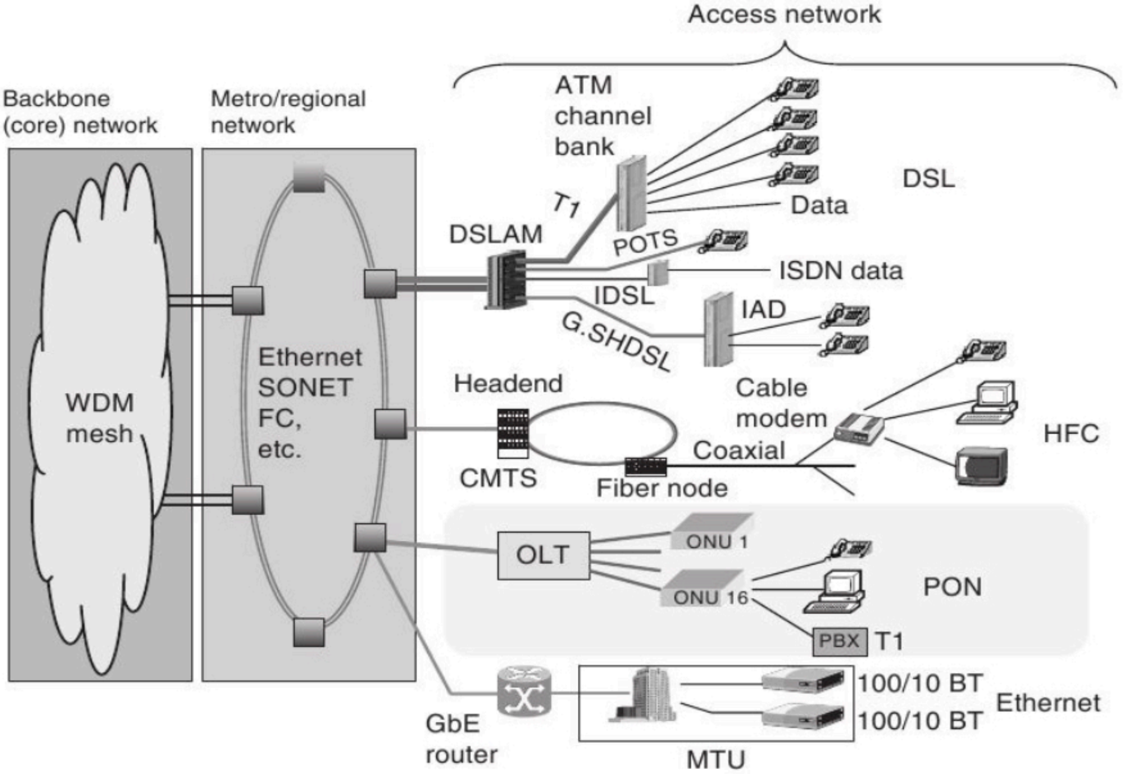


Figure 1-1 [2] General structure of Telecom network

Passive optical network (PON) is the main choice technology amongst service providers, in virtue of its high bandwidth (in the range of THz), it doesn't interfere with electromagnetic (less signal error), longer life span and much less power consumption compared with other copper or wireless based distribution technologies. PON or alternatively called Fiber to the "x" (FTTx), where "x" could refer to Home (FTTH), Premises (FTTP) or curb (FTTC) or other name depending on where the Optical Network Unit (ONU) termination takes place from the end user. [2]

1.1 History and evolution of FTTH

In 2001, the FTTH Council was formed in North America. The same year the Broadband Internet Access Act (BIAA) provided tax incentives to investors to push the development of future broadband technologies. Another important factor that helped the FTTH technologies momentum was the removal of the unbundling requirement on FTTH by the FCC (US Federal communication commission) in 2003, which meant that the operating companies are no longer obligated to share their access network (last mile) with competition, this alone provided tremendous financial motivation towards FTTH network implementation [3]. Figure 1-2 shows different FTTx network types.

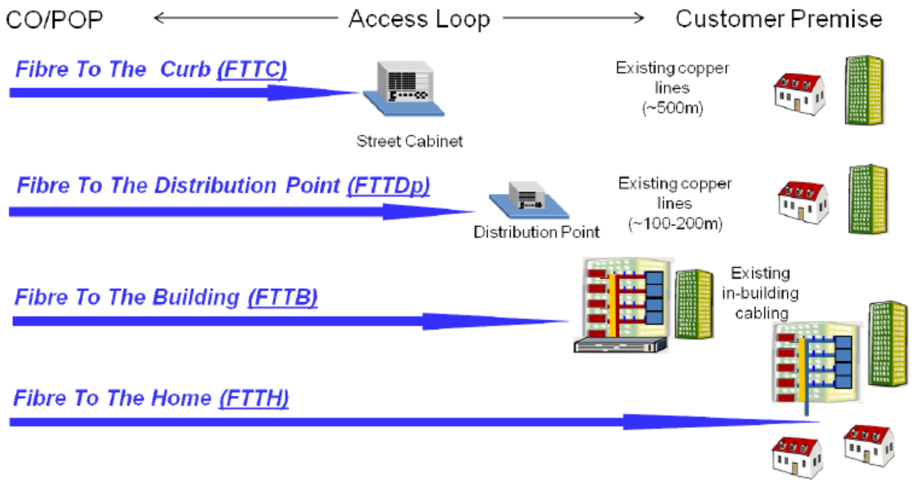

CO/POP
Access Loop
Customer Premise
Fibre To The Curb (FTTC)
Street Cabinet
Existing copper lines (~500m)
Fibre To The Distribution Point (FTTDp)
Distribution Point
Existing copper lines (~100-200m)
Fibre To The Building (FTTB)
Existing in-building cabling
Fibre To The Home (FTTH)

Figure 1-2 [4] different FTTx network types

For a better understanding of PON existing and upcoming standards, it is important to give a brief historical roadmap explanation, Figure 1-3 summarizes historical roadmap for different PON technologies. When it comes to PON standards there are two main paths to follow. In 1990 ITU (International Telecommunication Union) introduced APON (ATM over PON), it almost doesn't exist today, APON considered more experimental than practical at the begging, however APON formed the foundation towards building next PON technologies, it uses ATM (Asynchronous Transfer Mode Data link protocol) in layer 2. The lack of video broadcasting ability in APON, lead to development of BPON (Broadband PON) where a 1550nm wavelength was preserved for broadcast video overlay, another close wavelength (1480-1580nm) dedicated for downstream data for max data rate of 622.08 Mb/s, however the release of G983.1 2005 proposed max downstream data rate of 1244.16 Mb/s with operating wavelength of (1480-1500nm in Single fiber) and (1260-1360nm in Dual fiber), for upstream 155Mbps data rate an operating wavelength of (1260-1360nm) and for 622Mbps max upstream data rate the operating wavelength is (1260-1360 nm, *depends on MLM/SLM laser type, see* [5]*Table 4-f/G.983.1*). [5] [1]

Because of the increasing demand for higher data rate broadband PON and the fact that Ethernet became the dominant layer 2 protocol, in 2004 ITU came up with the Gigabit PON (GPON), also known as G.984 standard, with higher data rate of 2.5 Gbps in the downstream and either 1.2 or 2.5 Gbps in the upstream, with the requirement to support Ethernet IEE802.3, TDM or the previously used ATM as layer 2 protocol, though, after Ethernet almost became the de-facto for data link commercial appliances, the industry choose GPON-Ethernet based option to manufacture network products for mass-market. Afterward, ITU came up with XGPON1 & 2 (ITU-T G.987.x) standard, then the Next Generation NG-PON2 (ITU-T G.989.x) which could support capacity of 40Gbps in both directions, a 40-km to 60-km fiber length and serving 256 (or more) end users. Likewise, The ITU-T is still cooking even higher speed PON standards for the future [1][6]. Figure 1-4 Shows PON technologies evolution.

The other PON path was driven by IEEE 802.3 Ethernet subcommittee, in 2004 they came up with the 802.3ah Ethernet PON (EPON), also known as Gigabit Ethernet PON (GE-PON) or Ethernet in the “First Mile” (EFM). The first EPON 802.3ah offered 1Gbps in both directions, EPON was more flexible standard than the GPON as it covered absolutely fundamental requirement, things such as encryption and system management was left up for developers to decide what works best for their existing systems, this flexibility came with extra cost of compatibility and interoperability issues across different vendor’s equipment. Later in 2009 the standard was updated to 802.3av to offer 10Gbps downstream and upstream. EPON gain its momentum in Asia, conversely the GPON was the choice in North America and Europe, though, and some EPON implementations existed too. In Addition, 802.3ah also proposed a P2P FTTH Ethernet specification, something that GPON standard did not conclude. A new committee sponsored by the IEEE and backed up by other telecom vendors was formed under the name of Service Interoperability in Ethernet PON (SIEPON) which led the IEEE P1904.1 project aims to add missing specifications on the 802.3ah&av standards in order to achieve multivendor interoperability ecosystem, hence making EPON more vigorous and commercially appealing standard. [1][7]

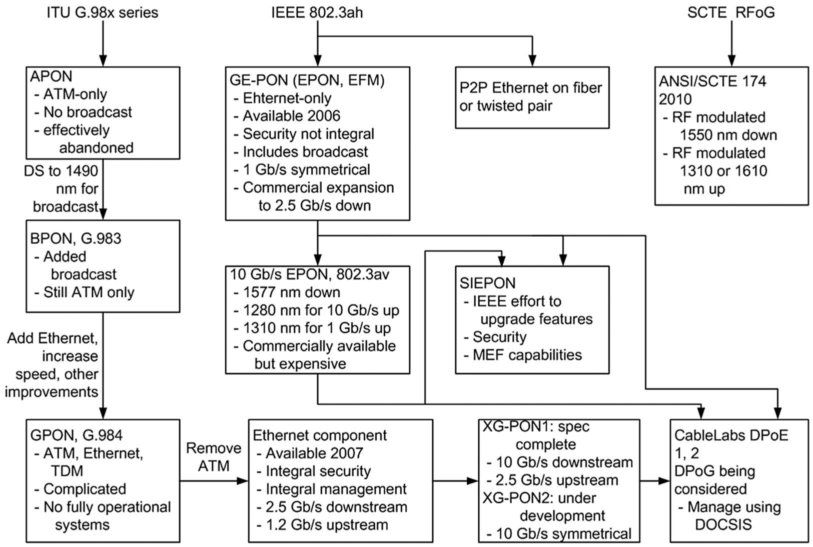


Figure 1-3 [1] PON technologies roadmap

The EPON story in the US was a bit different, since Cable TV providers were keen to upgrade their existing infrastructure to EPON, however this brought tremendous challenges, because of the totally different EPON management protocols that didn't match with existing DOCSIS (Data Over Cable Service Interface Specification), the heavily invested DOCSIS played major role in residential data market, though, it was more economically and technically feasible to modify the EPON standard instead, also to guarantee smooth transition without disrupting ongoing business, the target approach was to utilize existing HFC infrastructure while build a new PON infrastructure in residential greenfield. The efforts to modify EPON standard was driven by Cable Television Laboratories (CableLabs), they have created DOCSIS provisioning of EPON (DPoE) standard which enabled management of EPON through DOSIS. Furthermore, to be able to use existing HFC physical layer IEEE had to append EPON protocol and add PON optical-to-electrical signal converter in the last mile, hence EPON Protocol over Coax (EPoC) (*also known as RF over Glass - RFoG*) standardization was initiated in 2010, the work was completed in Sep 2016 with the approval of IEEE standard 802.3bn-2016. [1][8].

1.2 Fiber to the Home (FTTH)

Amongst all PON access network technologies, FTTH PON, where Optical Network Terminal ONT installed at subscriber's home, has the advantage of serving multiple end users (point to multipoint, or P2MP) on a single fiber sharing same bandwidth thus provides multiple services at low cost where all outside plant elements are passive. Organizations such as ITU-T and IEEE lead the establishment of new PON standards, this opened the doors for massive scale industry for PON technologies on considerably low cost. For that reason, FTTH became the main choice for Green-field residential deployment. It is also good to mention that FTTH has a Point to Point (P2P) architecture too, in P2P a single fiber will extend from the CO until subscriber's home.[3]

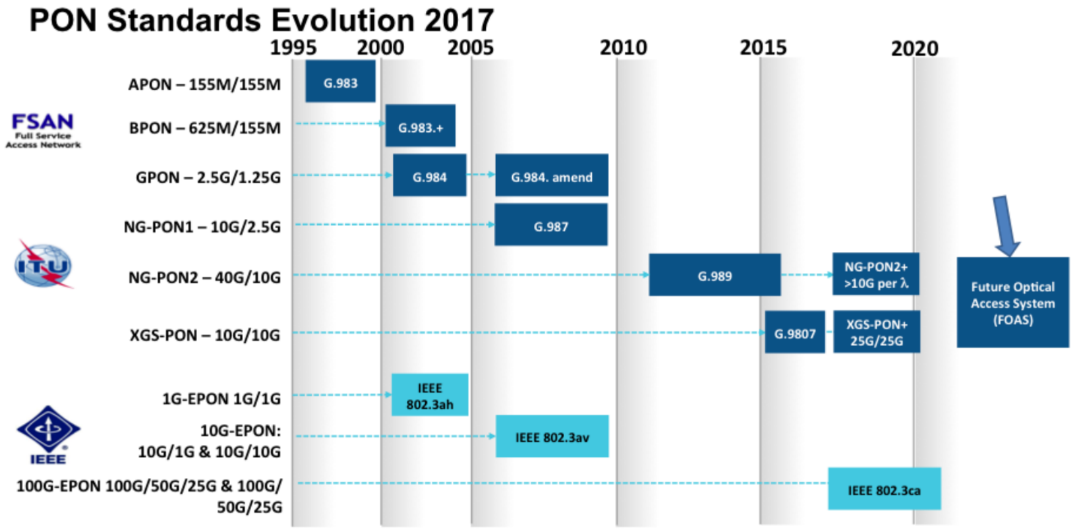


Figure 1-4[4] PON systems evolution

1.3 TDM and WDM comparison

PON utilizes both Time Division Multiplexing TDM and Wave Division Multiplexing (WDM) to share physical fiber between OLT (optical line terminal) at the CO and the Optical Network Unit (ONU) (or sometimes referred to as ONT) at the customers' premises. In TDM-PON same signal is broadcasted (downstream) from the OLT to all ONUs using a power splitter, where all signals from different ONUs are multiplexed in TDM (TDMA), see Figure 1-5a. An embedded address label within each signal is used by the ONU to identify and receive the data packet that belongs to it or discard it if it belongs to other ONU. TDM-PON exists in all PON technologies such as GPON and EPON.[2]

As illustrated in Figure 1-5b, in WDM-PON OLT transmits all different signals for ONUs on a single fiber, where each signal is transmitted on a different wavelength (or channel), later these signals are decoupled using WDM coupler at each ONU [2]. Technically speaking, M2MP solutions are utilizing both TDM and WDM, as WDM coupler is used at the OLT to couple downstream and upstream frequencies and/or analog video overlay (optional), while TDMA is used to combine upstream ONU signals.

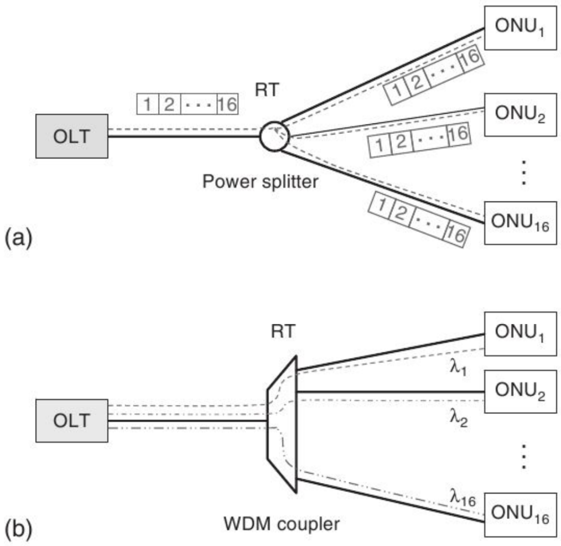


Figure 1-5 [2] Architecture of (a) WDM-PON (b) TDM-PON

Chapter 2 FTTH Architecture

2.1 FTTH network components

FTTH PON architecture generally contains the following components, Figure 2-1 shows general PON architecture:

- Optical Line Terminal at the Central office CO, the OLT is responsible for connecting the access network to the Internet, the Public Telephone Switching Network PTSN and the Video distribution source. Video source could be Analog such as a Direct Broadcast Satellite DBS or Cable TV (CATV) or even digital such as IPTV. Before Analog video signal enters the OLT, it is converted to Optical signal by an optical video transmitter device which converts the analog electrical signal into optical and utilizes Wide Wave division multiplexing WWDM coupler to transmit optical signal as an overlay, later will be explained in more details.[3]
- The feeder fiber from the OLT to the splitter in the case of P2MP-PON (One feeder per splitter) or fiber pair in case of point-to-point PON. [3]
- The splitter, a one splitter per feeder fiber, each splitter will serve 32 up to 64 end users (depends on PON network topology), also there could be more splitters per each fiber branch in a cascade design, see Figure 2-2 (a) and (b). Splitter (s) located at Fiber Hub distributer close from end user premises. [3]
- Distribution fiber to connect each splitter’s branch with the Optical Network Terminal ONT (ONT consists of Optical Network Unit ONU and Optical Network Interface ONI). [3]
- ONT for each end user, typically installed and powered at user’s home, it converts the optical signal to electrical and provides different connection to each service (Video RF port for example, plain old telephone service (POTS) for Telephone service, or Ethernet for internet data e.g. RJ45), where services varies depending on subscription and/or using technology. [3]

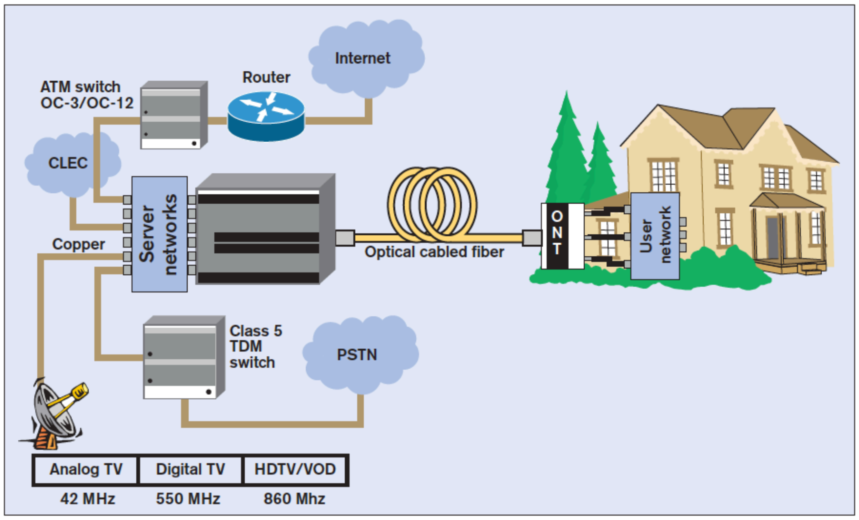


Figure 2-1 [3] General PON architecture.

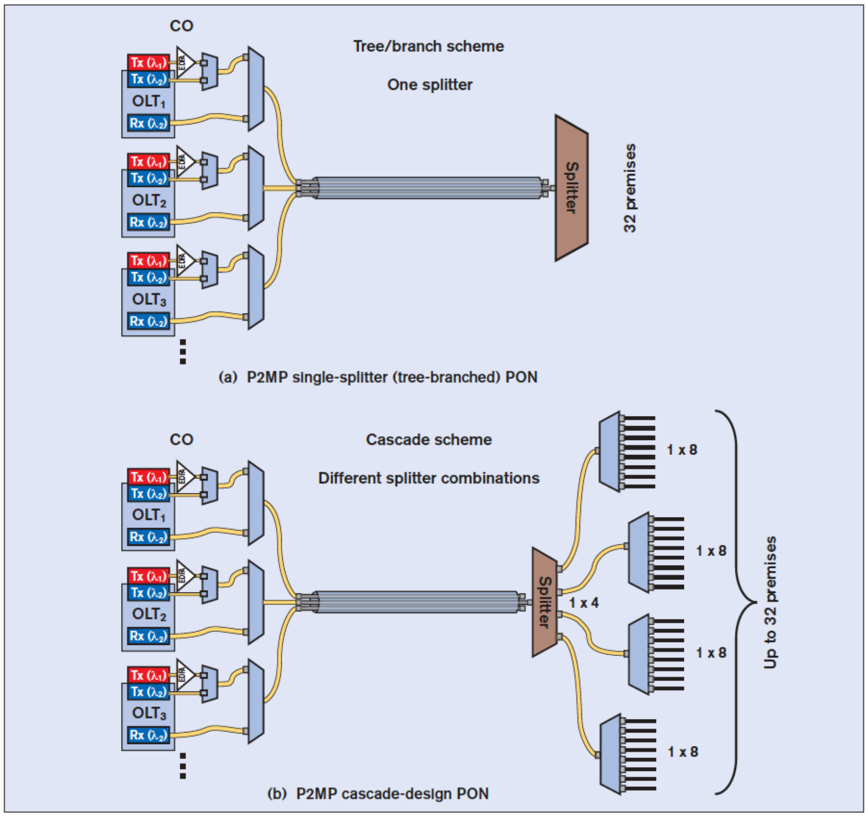


Figure 2-2 [3] Point to Multi-Point Splitter topology options in a PON network.

2.2 FTTH network topology

2.2.1 Point to multi-point (P2MP)

Downstream signals from OLT to ONT (data, voice and video) and upstream signals from ONT to OLT (voice and data) are transmitted on the same fiber feeder bidirectional, the feeder extends from the CO (or OLT) to the branching point (Optical Splitter in case of GPON) and from there to every end user where a single fiber per branch is deployed, in GPON data will be encoded with special tag for each user as it will be explained later. Instead, EPON uses Ethernet Switches installed in the last mile to logically split the traffic per user (e.g. define a VLAN per user).[4]

As mentioned in Chapter 1, frequency bands are precisely specified by different PON standard, ITU-I specifies bands for G98x series (APON, BPON, GPON and later technologies), where IEEE specifies bands for EPON technologies series. A WDM coupler at the CO is responsible for coupling these frequencies as we will see later when we speak about each different technology in more details[1]. See Figure 2-4.

2.2.2 Point to point (P2P)

P2P architecture is when a fiber run from OLT directly to serve a single ONT, symmetrical downstream and upstream data is transmitted over the same fiber on a different optical wavelength as specified by the IEEE Ethernet. In P2P the ONU equipment is cheaper as it doesn't have to deal with complicated PON protocols, since it only receives data for a single user, however the burden of correctly route data to its correct destination (ONU) laid on the switch at the CO, also because there will be a single fiber from the CO to the end user this means more fiber is needed. The clear advantage of P2P is more available bandwidth for each user (no sharing), also more privacy as user's data is physically split when it's transmitted on a single separate fiber and no other user can see it[1]. See figure 2-3.

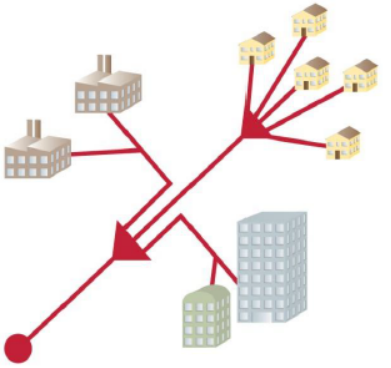


Figure 2-3 [4] P2P topology

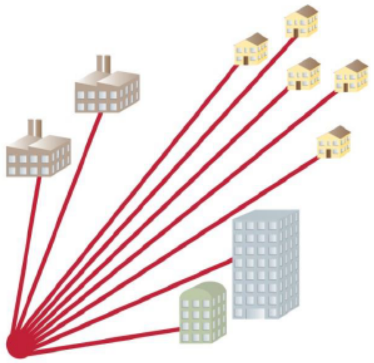


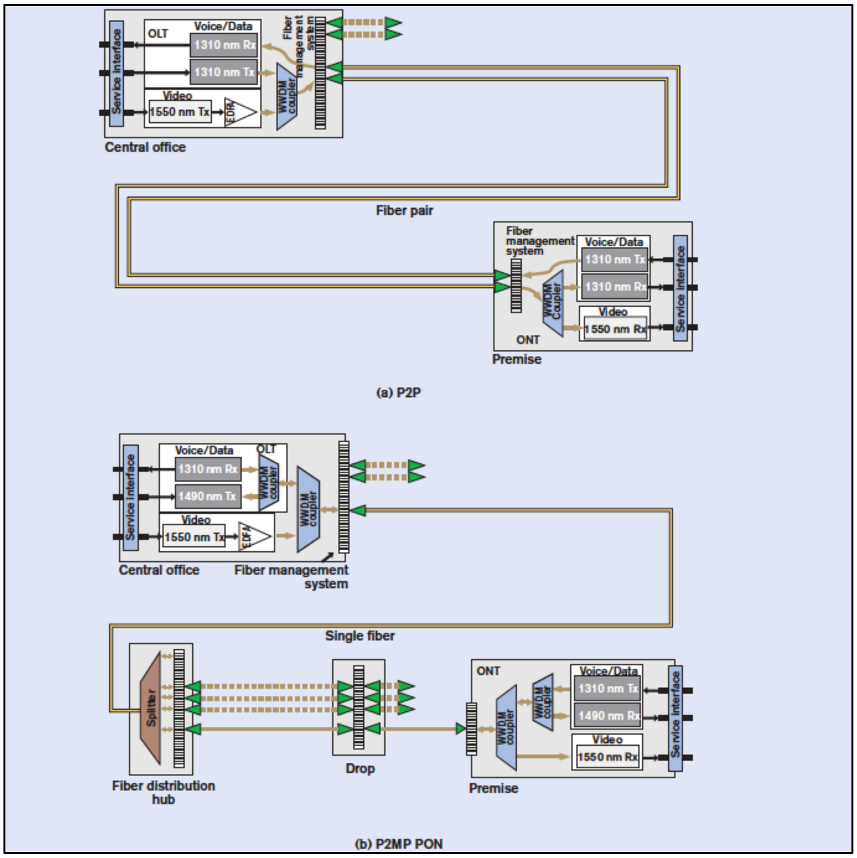
Figure 2-4[4] P2MP topology

Chapter 3 Network design and technologies

3.1 PON technology

As mentioned in 2.2, PON technologies exist in two topologies. In P2P CO, OLT is directly linked with the ONT with a pair of fibers one for downstream data and voice and the other for upstream data and voice, a wavelength of 1310nm is used for down and upstream. A video broadcast could be implemented on 1550 nm wavelength in the downstream fiber as an analog overlay. However, in P2MP the OLT could be connected to multi ONTs (32 up to 64) through a splitter, the downstream and upstream are sharing the same single fiber, same in P2P wavelength of 1550 nm could be used for analog video overlay broadcast service, for downstream data and voice 1490 nm wavelength is used, where 1310 nm is used for upstream data and voice from ONT to OLT direction. To avoid confusion, this report won't differentiate between ONT/ONU, either one will indicate the same component.

To transmit different wavelength on the same fiber a WWDM coupler is used installed in the distribution hub. Figure 3-1 shows the main components of PON technology (P2P and P2MP). [3].



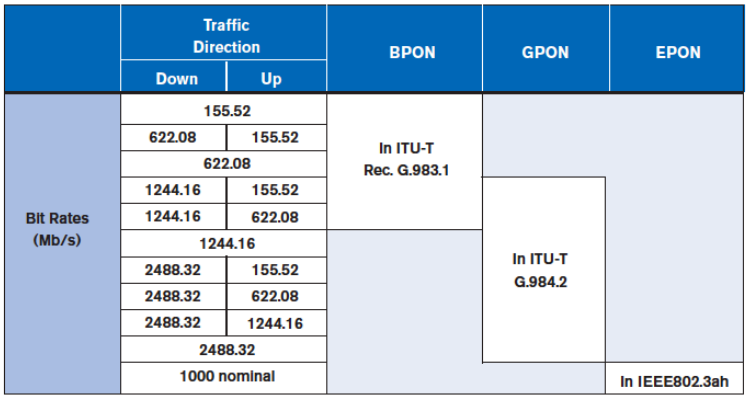
(a) P2P

(b) P2MP PON

Figure 3-1[3] Main components of PON Technologies (P2P and P2MP)

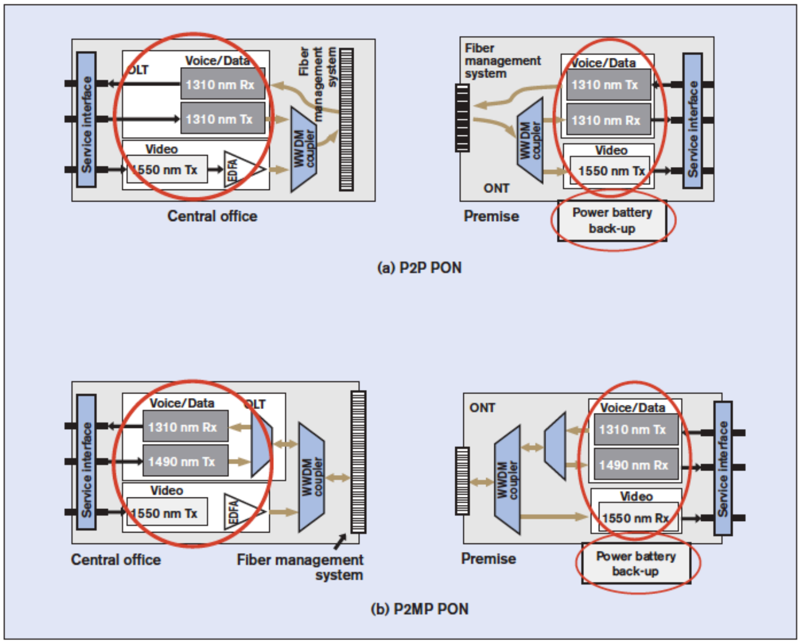
Figure 3-1 (a) shows P2P solution or it also called Active Ethernet (AE) which is part of IEEE 802.3ah standard, in case of digital video (IPTV) instead of analog video overlay, then the 1550 nm could be then used for downstream data (include IPTV as data packets) and voice.[1]

Table 3-1 [3] Supported Bit rates per standard



	Traffic Direction		BPON	GPON	EPON
	Down	Up			
Bit Rates (Mb/s)	155.52		In ITU-T Rec. G.983.1		
	622.08	155.52			
	622.08				
	1244.16	155.52		In ITU-T G.984.2	
	1244.16	622.08			
	1244.16				
	2488.32	155.52			
	2488.32	622.08			
	2488.32	1244.16			
	2488.32				
	1000 nominal				In IEEE802.3ah

Figure 3-1 (b) shows the P2MP solution based on PON standards which could be one of the two-standard path (ITU or IEEE path). Regarding ITU G.984.1 standard (GPON) it supports either asymmetrical bit rate of 2.4 Gbps (to be precise 2488.32 Mbps) downstream and 1.2 Gbps upstream or symmetrical 2.4 Gbps in both directions. On the other hand, IEEE802.3ah-2004 it offers symmetrical 1Gbps bit rate in both directions. Table 3-1 shows supported bit rates options per standard, older ITU BPON standard was mentioned too as some old implementations are still operational. [3]



hh

Figure 3-2 shows active components in (a) P2P (b) P2MP PON

3.1.1 PON network components

Even though, it mainly called passive optical network but it also contains active parts. So, it makes more sense to split the discussion into active and passive components.

3.1.2 Active components

Active components are parts of a network that consumes power, these could be found in different locations along the network grid. As shown in figure 3-2.

- OLT, located at the CO (head end), depends on its brand, OLT could handle many users, on general in P2MP GPON for example it could handle up to 16,384 users per shelf (if 64 users per one connection setup is used), or it can be P2P, where it can support 768 users instead. Note the number of subscribers will depend on the processing capacity of the used equipment and how many boards installed and their type, these numbers are mentioned just to give the reader some feeling on how much one OLT could do. Furthermore, the OLT can support more than PON standard simultaneously, depends on installed cards. For example, it could support multiple standards such as GPON, XGPON, XGSPON, which makes network upgrade process to a new technology is a matter of installing the new card.[4]

  In general, OLT performs the following main functions:

  1. Transmit downstream (data/voice/IPTV), broadcast video overlay (optional).
  2. Receive upstream data from the ONT's

- (Optional) Video terminal equipment, installed at CO, utilized if an analog video broadcast service is provided, the main functions are to transmit and amplify analog video signal. [3]
- ONU, also known as ONT in ITU standards, especially when it comes to single ONU supports a single subscriber. It is installed at end-user premises. There are different types of ONU depends on the type of end user, which decides its install location. It could be Indoor, outdoor, business or Multi-dwelling application, (here we're talking FTTC, FTTB, FTTH and FTTD) only the very last mile will be different,

precisely between ONU and UNI, but the rest of the network more or less is the same. [4]. In general, ONU performs the following functions:

1. transmit upstream (data + voice)
2. receive downstream (data + voice)
3. (Optional) receive analog video broadcast. [3]
4. It provides a different network connection to end user, for example it could provide RF connection for analog video, RJ45 for data access, POTS for telephone service, Wi-Fi connection and so on. See figure 3-3. [4]

- (Optional) Battery installed at ONT, to provide power in case of an electricity outage.

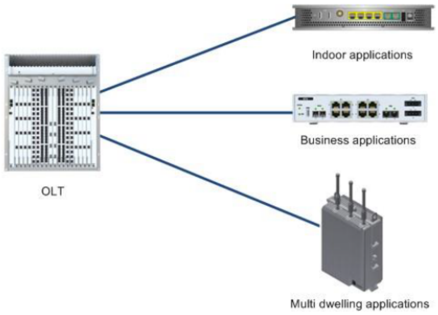


Figure 3-3[4] Different ONU/ONT type per application

3.1.3 Passive components

Passive components are parts of the network that do not require electrical power to perform their function, in other words they do not consume electrical energy (directly). In PON network, they are the WWDM couplers, optical splitters and all fiber cabling including (feeders, connectors, cable drops and distribution). Figure 3-4 shows passive components red circled.

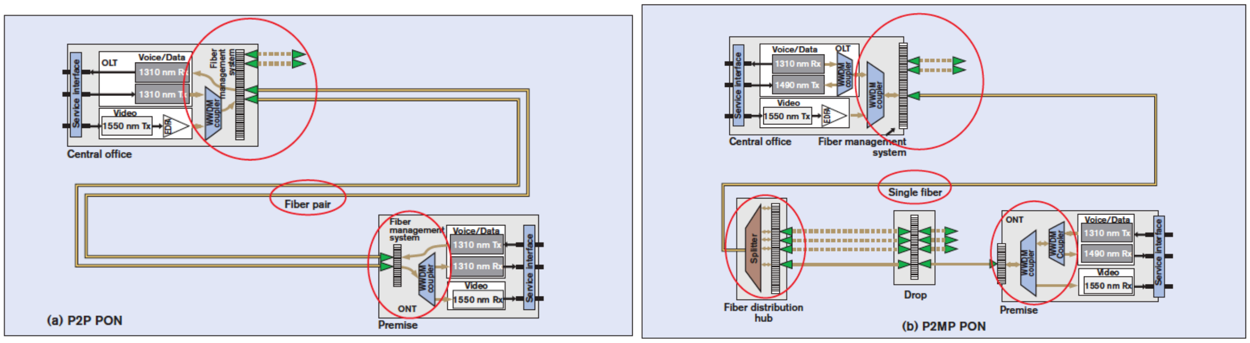


Figure 3-4 [3] Passive components in P2P and P2MP mode

3.1.3.1 WWDM coupler

A bidirectional component exists in P2P and P2MP PON, it is located at OLT and ONT. It performs the following main functions:

- In P2MP mode, WWDM coupler at OLT couples the (voice/data) downstream signal 1490 nm and the upstream signal 1300 nm on the same fiber. It also couples the analog video 1550 nm with the "down/upstream voice/data signal" 1490 & 1300 nm. See figure 3-5.
- In P2MP mode, at the ONT WWDM coupler, couples/decouples the analog video 1550 nm and the upstream/downstream voice/data signals 1490 & 1300 nm. And it also couples voice/data downstream 1490nm and the upstream 1300 nm signals.
- In p2p, at both OLT and ONT it couples/decouples analog video signal 1550 nm and the voice/data downstream signal 1330 nm (only). In case of digital video (e.g. IPTV) then no need for the WDM coupler. [3]

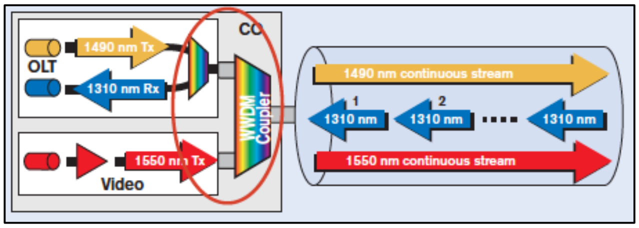


Figure 3-5[3] WWDM Bidirectional coupler P2MP

3.1.3.2 Optical splitter

It only exists in P2MP PON, a bidirectional component located at the distribution hub (outside planet), it has only one input (single fiber) and multiple outputs to allow sharing the bandwidth between N number of subscribers (e.g. 32 or 64). The main functions of the splitter are:

- Splits the "coupled" downstream (voice/data) + Video signal to 32 (or 64) fiber branch (one fiber per branch/ONT).

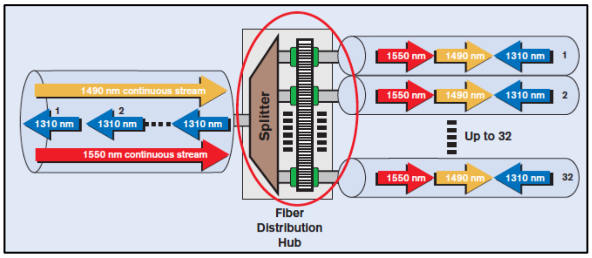


Figure 3-6 [3] Optical Splitter in P2MP mode

- Combines 32 (or 64) ONT (user) upstream (voice/data) signals to a single fiber towards the OLT. See figure 3-6.

There is also other physical characteristics, either optical or mechanical attributes, for passive and active components likewise, these attributes affect device limitation and network implementation such as signal loss and actual reach distance and other limitations, however they are out of scope of this project and we will not discuss them further, instead we will focus on higher layers (layer 2 and above), since nearly same physical layer within access network infrastructure is utilized in R-CORD project, besides any differences between conventional PON network and R-CORD starts from layer 2 and above.

3.2 Bandwidth management

Before speaking about each PON protocol, it's important to explain how bandwidth is managed within any PON solutions. At this point you should be aware that PON solution is utilizing TDM and WWDM to manage wavelength and share bandwidth.

All PON P2MP technologies (e.g. GPON, XGPON, EPON...etc.) are using TDM to assign bandwidth, where every ONT is given a time frame to transmit its data upstream toward OLT which forwards each packet by its network address to the outside network (e.g. Internet). In the opposite direction, the same downstream signal is transmitted from OLT to all ONU, received data at ONU is recognized using a logical tag (e.g. port ID) either sent to a higher layer or discarded by the UNI.

Different wavelength is defined by ITU-T & IEEE for downstream, upstream and analog video broadcast. These frequencies are combined during transmission between OLT and ONT by means of wide wavelength multiplexing WWDM as mentioned in 3.1.3.1.

In conclusion, both TDM and WDM are being used in all PON P2MP solutions, with one exception regarding WDM in P2P mode, where it could be implemented only if analog video broadcast service is provided. [3]

Chapter 4 Gigabit Passive Optical Network - GPON

4.1 Introduction

As discussed in chapter 1, GPON had its momentum in North America and Europe mainly, but not exclusively, it was widely embraced by telephone operators. GPON standard codified under G.984.1, the recommendation saw light in March 2003 narrated by ITU-T Study Group 15, followed by series of recommendations as described in Table 4-1, these standards described a promising optical fiber access network for residential services and businesses that is able to provide downstream speed of 2.4Gbps and two speed options for upstream, either 1.2 or 2.4 Gbps. This chapter will discuss G.984 recommendation series with fair details in certain areas, its characteristics, Transmission conversions layer, ONT management and controls (OMC) interface.

Table 4-1 [9] GPON G.984 series recommendations and release history

Recommendation code	**Release date(s)**	**Description**
G.984.1	March 2003, replaced by March 2008 release, Amd1 Oct 2009, Amd 2 April 2012	Gigabit-capable passive optical networks (GPON): General characteristics
G.984.2	March 2003, Amd 1 Feb 2006, Amd 2 Mar 2008	(G-PON): Physical Media Dependent (PMD) layer specification
G.984.3	April 2004 replaced by Jan 2014, Amd 1 2009, Amd 2 2009, Amd 3 2012	(G-PON): Transmission convergence layer specification
G.984.4	Jun 2004 replaced by Feb 2008, Amd1 2009, Amd2 2009, Amd3 2010, Erratum 2009, Cor1 2010	(G-PON): ONT management and control interface specification.
G.984.5	Sep 2007 replaced by May 2014, Amd1 May 2008.	(G-PON): Enhancement band
G.984.6	Mar 2008, Amd1 2009, Amd2 2012	(GPON): Reach extension
G.984.7	July 2010	(GPON): Long Reach

4.2 GPON main characteristics

4.2.1 Bit rate

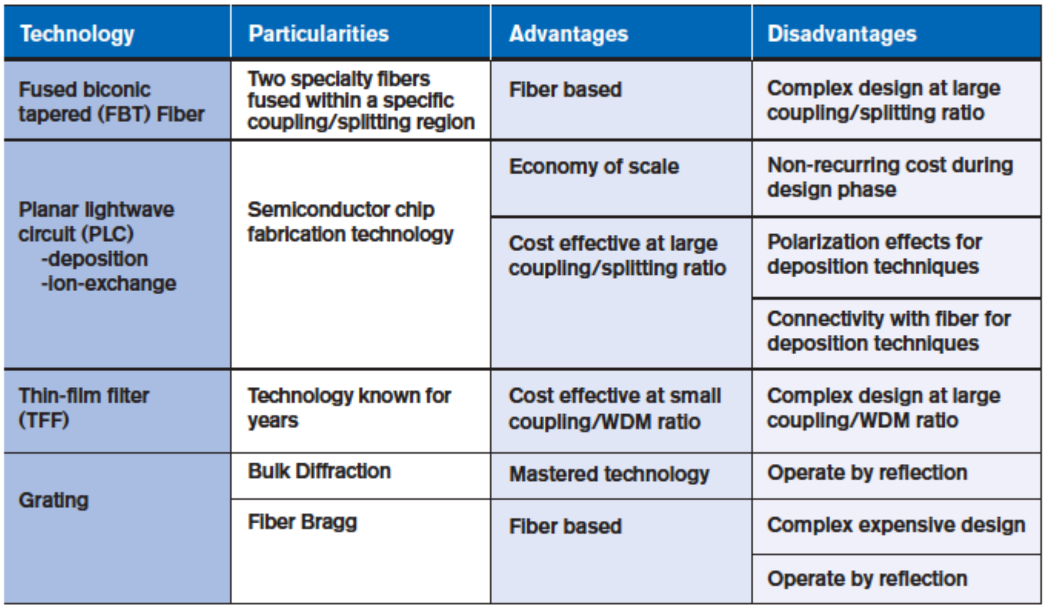
ITU-T recommendation 984.1 [6] defines two options for bit rate as follows:

- 1.24416 Gbps in upstream, 2.48832 Gbps in downstream
- Symmetric option with 2.48832 Gbps in both directions.

4.2.2 Logical and physical reach

Ideally the fiber between OLT and ONT could reach 60 km regardless of component’s technical specifications, however in reality, the physical reach value is between 10 and 20 km, where actual physical reach is governed by the technical specifications of the utilized passive components such as Insertion Loss (IL), coupling/splitting ratio, loss uniformity, Optical Return loss (ORL) and Polarization dependent loss (PDL) as well as the fabrication technologies used to build couplers and splitters, Table 4-2 shows a comparison between some of these technologies. In other words, the maximum distance limitation is governed by the passive components characteristics and/or the intended data rates, for an example, for a data rate of 1.25Gbps and higher, the maximum reach can’t exceed 10km. Another factor that affects the physical reachability is the differential fiber distance, defined as the difference between the nearest and farthest ONTs from the OLT, because a longer variation in a differential distance results in a longer discovery grant, hence more idle time for the upstream transmission will occur, which affects the network performance, as a result ITU G.984.1 standard has limited the differential distance to a maximum of 20 km in order to maintain a practical discovery grant period. [6] [3]

Table 4-2 [3] Comparison between coupler and splitter technologies.



Technology	Particularities	Advantages	Disadvantages
Fused biconic tapered (FBT) Fiber	Two specialty fibers fused within a specific coupling/splitting region	Fiber based	Complex design at large coupling/splitting ratio
Planar lightwave circuit (PLC) -deposition -ion-exchange	Semiconductor chip fabrication technology	Economy of scale	Non-recurring cost during design phase
		Cost effective at large coupling/splitting ratio	Polarization effects for deposition techniques
			Connectivity with fiber for deposition techniques
Thin-film filter (TFF)	Technology known for years	Cost effective at small coupling/WDM ratio	Complex design at large coupling/WDM ratio
Grating	Bulk Diffraction	Mastered technology	Operate by reflection
	Fiber Bragg	Fiber based	Complex expensive design
			Operate by reflection

4.2.3 OLT and ONU functional blocks in GPON

As described in 3.2, the OLT could provide a connection to a switched network (e.g. PSTN), an ISP and maybe to a Cable-TV. It also provides ODN interfaces (last-mile). Figure 4-1 shows a standard OLT functional blocks, they perform three main functions: Service port interface, Core-Connect and Optical distribution (ODN) interface.

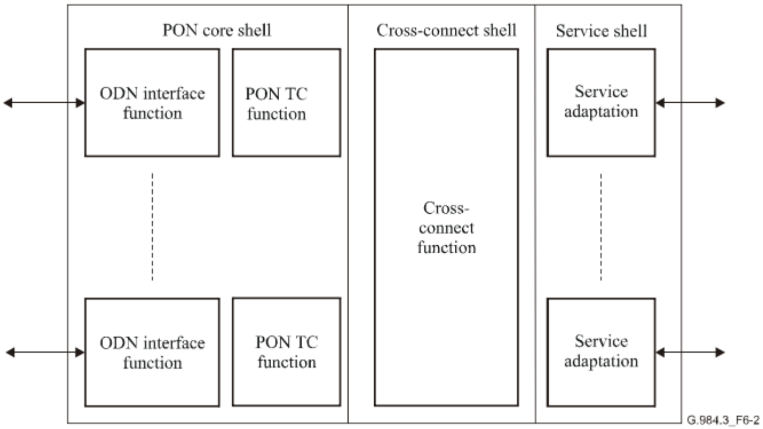


Figure 4-1 OLT functional blocks as per G.984.3

The OLT's main function blocks are as per the following:

- PON Core-shell consists of the following functions (blocks):
  - ODN interface: provides connectivity to an ODN (optical interface).
  - PON TC layer function: performs Media access control, Framing, Dynamic Bandwidth allocation and reporting, Operation Maintenance and Administration (OMA), mapping Ethernet into GEM frames, ONU management, activation, and deactivation …etc.
- Cross-connect shell, provides cross connection between the core-shell and/or the service shell.
- Service shell, provides conversion between the service protocols PDU and the TC layer frame.

The ONU functional blocks are shown in Figure 4-2, they are quite similar to the OLT functional blocks, despite there is no cross-connect-shell function, because the ONU has only one connection and that's with the OLT.

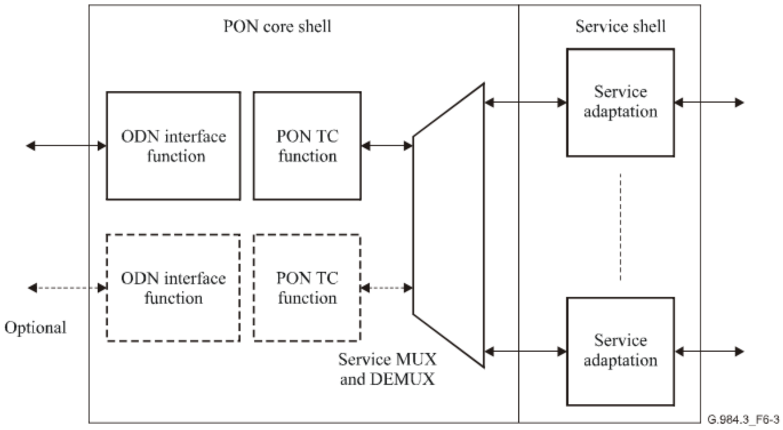


Figure 4-2 ONU functional blocks (reference G.984.3)

4.3 GPON Transmission convergence (GTC) layer

GPON uses TDM to transmit the downstream data, the OLT sends same traffic to all ONUs, and every ONU filters its own traffic based on the port id, as it will be explained later. You may picture the physical layer's downstream traffic as it was a non-stop running train, where every trailer represents a 125µs frame, these frames will carry the transmission convergence layer (GTC) frames.

GTC frames will carry in their own payload a one or more variable size GEM frame, however each GEM frame will carry only one or a fragment of the service data unit (SDU). The SDU, historically, could be any supported protocol packet such as ATM, Ethernet or E1/T1. Nowadays, only Ethernet is supported since the industry is no longer interested with the other two. The Physical frame length is fixed at 125µs, this implies that it may carry a different amount of data at a different bit rate, for 2488.32Mbps it carries 38,880 bytes in downstream, and with 1244.16 Mbps upstream bit rate it carries 19440 bytes.[10][1]

In upstream direction, the story is different. Due to the fact that the upstream data generated by an ONU has a burst nature, therefore an ONU's upstream frame is called a burst. The bandwidth allocation mapping for an ONU is managed by the OLT, where every ONU is allocated its own transmit time, where a transmission time is allocated once the ONU is discovered by the OLT. The term traffic container (T-CONT) is used to identify the allocated time for every ONU, and in TC layer, T-CONTs are mapped to an allocation ID (Alloc-ID) as we will see later when discussing framing. Each ONT may support one or more T-CONTs. A T-CONT might interface with one or more logical ports according to its bandwidth requirements. Each Port represents an end user service and/or equipment. Figure 4-3 shows GPON identification structure. [10][1]

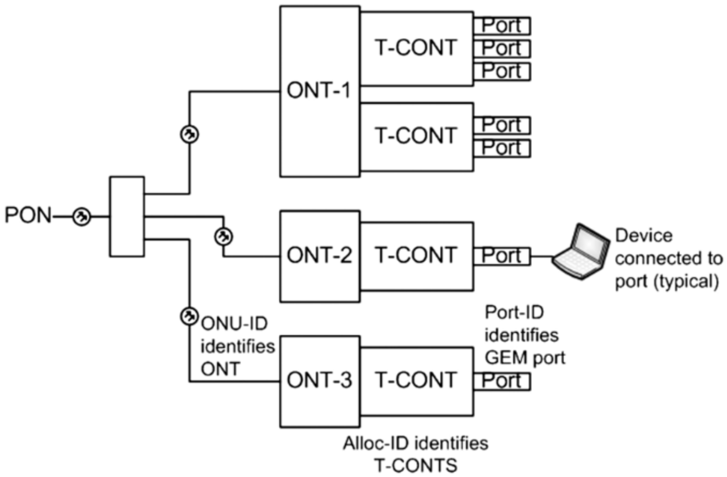


Figure 4-3 [1] GPON identification structure

In conclusion, the two key functions of the GTC layer are, the media access control (MAC) where the OLT allocates a specific time for each ONU so that it could talk separately within an upstream GTC frame without interfere with another ONUs. The other key function is ONU registration and activation, where the OLT gives an ONU an ONU-ID, maps it's TCONT with an Alloc-ID, so ports (PORT-ID) could be designated to its T-CONTs (Alloc-IDs) in order for the ONU to be able to send and receive data.

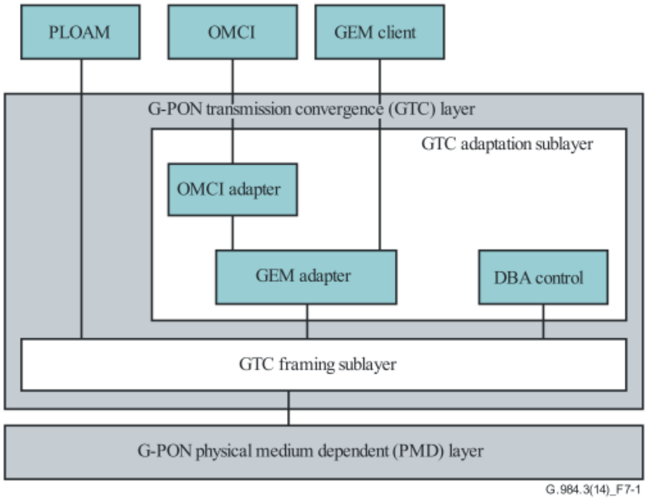


Figure 4-4 [11] GTC protocol stack

4.3.1 GTC protocol stack

Figure 4-4 shows the whole protocol stack for GPON transmission and convergence layer (GTC). It contains two plans, the management and control (M/C) plan and the User data plane, each plan has a different protocol stack. The GTC layer contains two sub layers, the adaptation sub layer and the framing sub layer. We will talk about each one of them in more details.[11]

4.3.1.1 Management and control plane (M/C plane)

Figure 4-5 shows the protocol stack for M/C plane. It contains three components, PLOAM, Embedded OAM and ONU management and control interface (OMCI). The embedded OAM exists inside the GTC frame header where it carries messages between the OLT and the ONU in order to manage time critical communication that controls the Physical Medium depend layer (PMD) and the GTC layer, also it's responsible for the bandwidth allocation information (BWmap), the dynamic bandwidth assignment report (DBR in upstream) and the encryption security key switching (in the Identification field, 30-bit supercenter). [11]

The PLOAM also used to manage the PDM and the GTC layers, but unlike the embedded OAM, the PLOAM exists in a separate field as part of the GTC header (PCBd), it's used for communication between the OLT and the ONU, for an example, during ONU discovery process and activation, as it will be explained in more details.

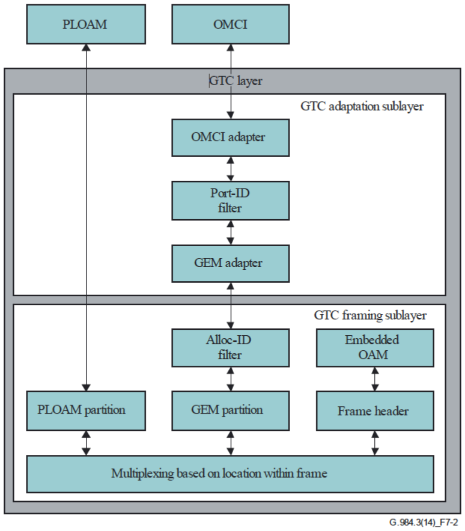


Figure 4-5 [11] C/M plane protocol stack

The OMCI manages and controls the service layers above the GTC layer, technically it's not a part of GTC layer, and however the later should provide the GEM frame interface to carry the OMCI messages.

4.3.1.2 User data plan (U-plane)

As shown in figure 4-6, the protocol stack for this plan is responsible to carry the user data or SDU (In this report we will always consider the SDU as an Ethernet frame) into the GEM frames payload, and adds a port-ID into the GEM header to be used at the receiver side to filter the user data. In the upstream it uses T-CONT for every allocation interval (Alloc-ID) to transmit the user data from the ONU to the OLT and beyond. [11]

In order to understand how the traffic flows through the U-Plane protocol stack, let's break it down into downstream and upstream as per the following:

- Downstream traffic (from OLT to ONUs): at the GTC adaptation sub layer the SDU is encapsulated into GEM payload, then GEM header is added with a port-ID (one or more PORT-ID are already associated with an Alloc-ID which is mapped to a T-CONT). At the GTC framing sub layer Multiple GEM frames combined into a GTC frame payload, a GTC header is added including bandwidth allocation intervals for each T-CONT. Without mentioning further details regarding the PMD layer, When any GTC frame is received at the ONU, it includes GEM frames that belong to all the other ONUs at this point, the ONU GTC framing sub layer will extract all GEM frames from the GTC frame payload, then the GTC adaptation sub layer will look into each GEM frame's header and filter them based on the port-ID, it will either send it to the upper layer or discard it. [11]

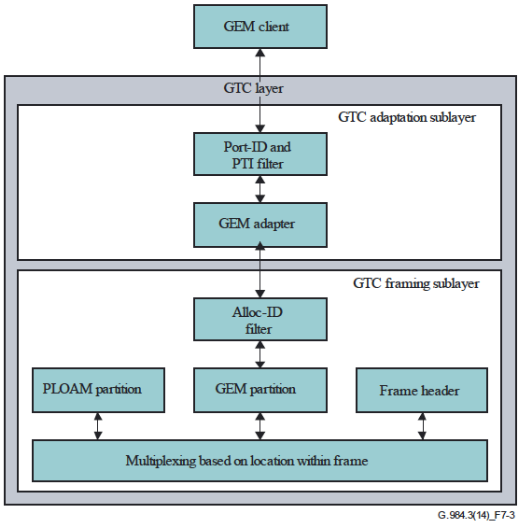


Figure 4-6 [11] U-plane protocol stack

- Upstream (from ONU to OLT): similar to the downstream GTC adaptation sub layer, it will also carry a GEM frame that carries an SDU data in its payload. The GTC framing sub layer will create a GTC burst frame that carries the GEM frame then send it to the OLT in the upstream where multiple bursts will get combined in a 125 μsec frame in the PMD layer. Next, the OLT will receive the GTC bursts, at the GTC framing sub layer it will extract the GEM frames from the bursts payload, the GTC adaptation sub layer will look into each GEM header and forward the SDU data to the GEM client accordingly. [11]

4.3.1.3 GTC framing sub layer

As indicated by the naming, the GTC framing sub layer is responsible for performing framing and de-framing process (or multiplexing/de-multiplexing), It's responsible for multiplexing the GTC payload and the PLOAM messages into the GTC frame (header and payload) when it's sending data in either direction, in the same time it performs extraction of encapsulated data when receiving. For example if sending from OLT to ONU, the framing sub layer creates the header of the GTC frame and adds it to the downstream frame, when the frame is received at the ONU then the framing sub layer decodes the header, execute OAM, and based on allocation interval ID the data is routed from/to the GEM TC adaptation sub layer. [11]

4.3.1.4 GTC adaptation sub layer

As shown in figure 4-4, it has two adapters, GEM Adapter and OMCI adapter. The GEM adapter reads GEM frames coming from the GTC framing sub layer in the upstream, in the downstream it maps the GEM frames into the GTC framing payload. Also, it performs an SDU encapsulation from/to the GEM payload. The OMCI adapter provides a GEM frame to carry OMCI channel's messages by designating a special PORT-ID chosen by the OLT for the OMCI channel communication, in other words, the OMCI adapter encapsulates/de-capsulate OMCI PDU into/from GEM payload. [11]

4.3.2 Dynamic bandwidth assignment (DBA)

DBA control functional block is part of the GTC adaptation sub layer as shown in figure 4-4, is responsible for the DBA process. In GPON, the available upstream bandwidth assignment is a dynamic process based on the ONU's traffic bearing entities (T-CONT) activity and the contracted bandwidth as per service level agreement (SLA). IN DBA, the OLT will do two things, it will request the ONU to send its buffering status update of each T-CONT, then reacts accordingly by assigning necessary bandwidth to each T-CONT within the ONU by using mapped Alloc-ID.

There are two ways of getting the T-CONT activity status, either by a status reporting, where the OLT asks the ONU to send the DBA reports (buffer queue) by setting a DBA flag inside the allocation field, alternatively the OLT will monitor the upstream traffic for any idle GEM frames within allocated intervals, accordingly the OLT will change the bandwidth allocation map in the next downstream PCBd BWmap, where it will reallocate bandwidth from idle ONUs to the more active ones.

Benefits of the DBA mechanism ranging from having more efficient use of the upstream bandwidth, which means more available bandwidth, thus more subscribes might be added, to a better service experience offered for each user according to their changing activity requirements. [11] [10]

4.3.3 GTC framing

Figure 4-7 shows, GTC layer framing in the downstream and upstream. Both are 125 µs, nonetheless, there are a quite differences between the both. Downstream GTC frame as header PCBd and contiguous GEM frames in the payload. Where in upstream consists of ONU bursts occupying the 125 µs GTC frame, each burst has its own leading PLOu header and one or more allocation intervals carries upstream data. [11]

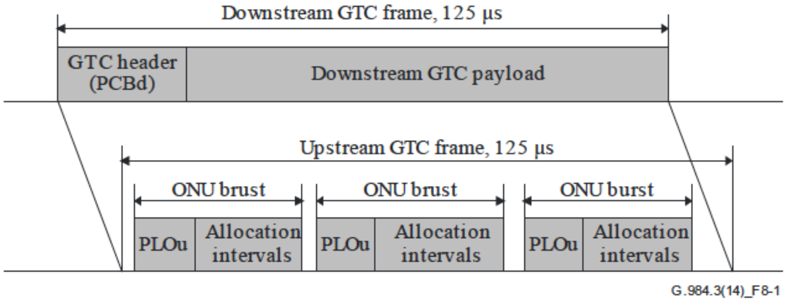


Figure 4-7 [11] GTC layer downstream and upstream frames

4.3.3.1 Downstream Framing

The GTC downstream frame is transmitted continuously within a 125 µs physical layer frame. As shown in Figure 4-8. In the upcoming sub-headers, we will discuss the frame structure in fair details.

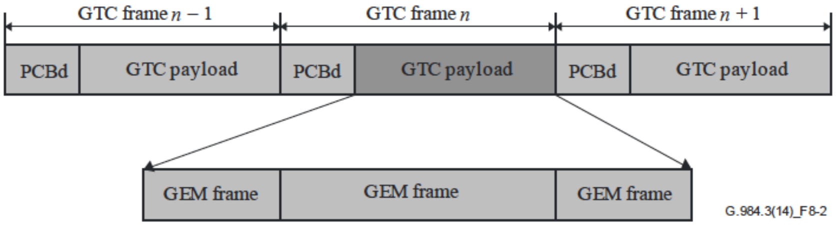


Figure 4-8 [11] GTC downstream frame

4.3.3.1.1 Physical control block downstream(PCBd)

As shown in Figure 4-9, the first field is **physical synchronization (PSync),** a 4 byte long pattern (0XB6AB31E0) to help the UNO receiver to perform byte synchronization, the ONU will be in a “Hunt state” while it examines the bits stream (bit by bit), as soon as it locates a pattern match, it instantly changes its status to a “preSync State” where it waits for a full frame length (125 µs) then re-check for the pattern, if it finds the pattern at the expected point, then it goes to the “Sync State” and starts to receive the frame. [3]

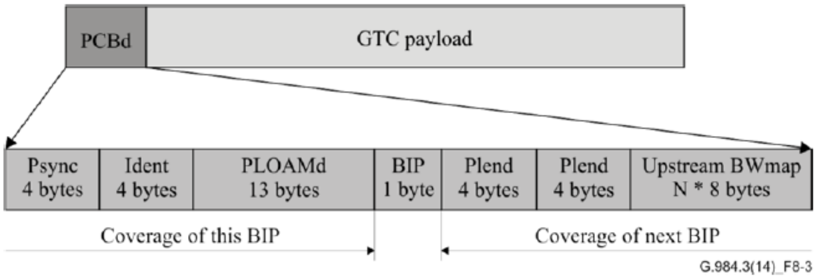


Figure 4-9 [11] Physical Control Block, GTC downstream frame header.

Identification field (Indent), as shown in Figure 4-10, 4 bytes in size, it has a super 30-bit counter which used for a downstream data encryption key switching, where every downstream frame is counted, it also contains one bit for an FEC identification, if it was set, it means that an FEC is used in the downstream frame. Another one bit is reserved for future use. [11]

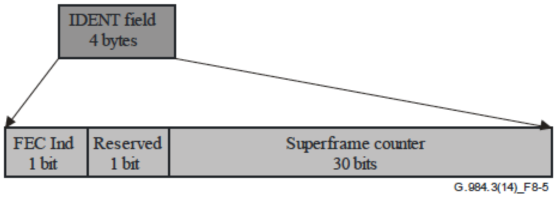


Figure 4-10[11] Identification field

Physical layer OAM field (PLOAM), a 13-byte field used to send the OAM control messages from the OLT to the ONUs, for an example, it is used for the ONU activation process. For more details about PLOAM message types and their use, you may refer to G.984.3 (9.2.1 & 9.2.2)

Bit Interval Parity (BIP), this 8-bit field contains a bit parity of all bytes sent since the last BIP. On the other side, the receiver should computed the BIP for all bytes received since the last BIP occurrence, then compares the computed BIP with the received BIP in order to find how many errors occurred on transmission. [11]

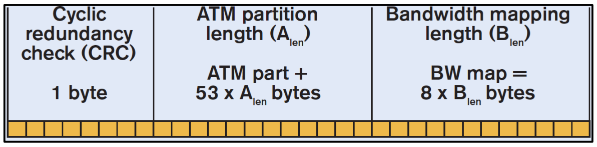

Cyclic redundancy check (CRC)
1 byte
ATM partition length (A_{len})
ATM part + 53 x A_{len} bytes
Bandwidth mapping length (B_{len})
BW map = 8 x B_{len} bytes

Figure 4-11[3] payload length field

Payload length (Plen) field, as shown in Figure 4-11, it contains three subfields. First, the Bandwidth mapping field, it specifies how many “8-byte allocations” will be mapped, please note that each discovered ONT could have one or more granted allocation according to its contracted bandwidth and traffic requirements (e.g. T-CONT). An ATM partition field that is not used any more after ATM opt out. An 8-bit CRC field which is used for error detection and correction for the whole payload length field. In addition, to add extra level of robustness, Plen field was copied twice, to be sure of uncorrupted Bandwidth mapping length. [10] [3]

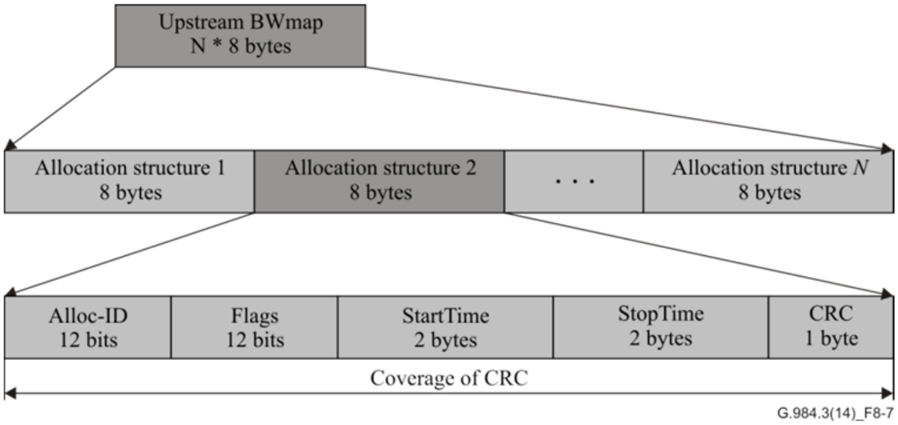


Figure 4-12 [3] Bandwidth mapping in downstream PCB for upstream ONTs

The last field and the most important in PCB is the bandwidth map (BWmap), as shown in Figure 4-12, it’s an 8 byte per upstream grant or (T-CONT), and a total of 5 fields are defined as per the following:

Allocation Identification (Alloc-ID), a 12-bits field, it can define total of 4095 grant allocations. The allocations from 0 to 253 are used to address the ONU directly, they should be used during the ranging process through the ONU activation. The very first Alloc-ID given to the ONU is called “default Alloc-ID” for this ONU. The default Alloc-ID is the same as the ONU-ID value, where the ONU-ID exists in the PLOAM field. The default ONU-ID or (default Alloc-ID) is used to address the PLOAM, the OMCI messages and/or the user traffic. However, if the ONU have an extra T-CONTs and needs more Alloc-IDs to be assigned for its additional T-CONTs, then the new Alloc-ID value will be between 256 to 4094.

Allocation value 254 is reserved to be used during “discovery grant” in the ONU activation process, where any unknown UNO that doesn’t have any allocated grants, maybe because it was just powered up or because it came back from a failure recovery, the ONU during the activation process initially will response with its serial number through a PLOAM message before it gets an Alloc-ID from the OLT. The 255 value is reserved to indicate that the allocation can’t be used by any T-CONT, but only those unknown ONUs are allowed to talk in order to be discovered and get activated by the OLT.

Start Time field, is two bytes in size, that used to define a start time for an ONU upstream bandwidth allocation, this time doesn’t include the start of the PLOu header, it only indicates at which byte the allocation interval for a T-CONT will start. Calling it a start time is a bit confusing, because technically it’s a byte counter, it counts byte order in reference to byte number 0 to byte number 38878 in a 2488.32 Mbps upstream bit rate.

Stop Time field, is two bytes in size that point at the last valid byte that is associated with this allocation interval (T-CONT end).

CRC field, an 8-bit used for an error checking and correction for the BWmap field. If an uncorrectable error was detected, then the whole allocation should be discarded by the ONU. [10] [3]

Flags field, a 12-bit field as shown in Figure 4-13, it contains the following indication flags:

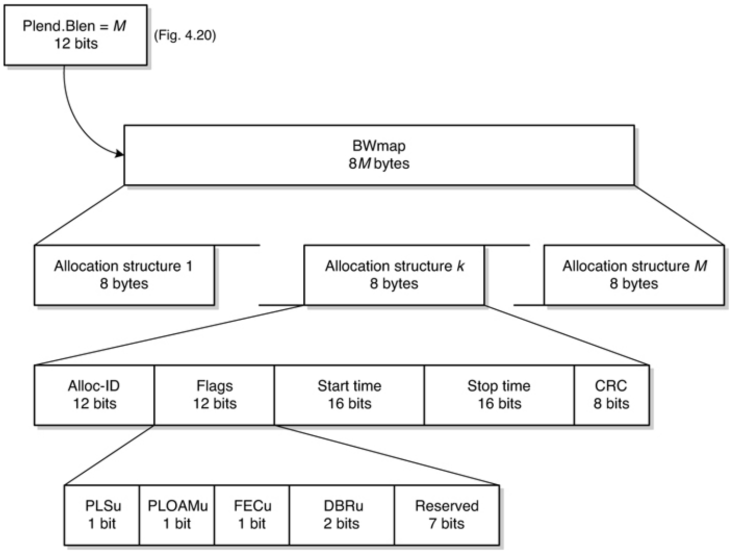


Figure 4-13[10] indication flags

Power leveling (PLSu), this is an obsoleted flag, previously used by the OLT to reduce the power level from an upstream signal sent by the ONU. This bit should always set to zero.

Send PLOAMu flag, is used to request an ONU to send a PLOAM message in corresponding upstream allocation when it is set.

Use FEC flag, if this flag was set, then it will enable the forward error correction (FEC) in the upstream allocation, in order that the ONU includes FEC in corresponding upstream allocation.

Send dynamic bandwidth report DBR flag, it is a 2-bit flag used to inform the UNO to send a DBA report in the next upstream frame, and also specifies the DBR mode. The value “01” means that a mode zero (a two byte) DBRu should is in use, a value “10” means that a mode one (three byte) DBRu should be in use. We will see this in more details during the upstream ONU frames discussion. [10] [3] [11]

4.3.3.1.2 GTC payload

GTC payload contains a contiguous GEM frames, once they received at the ONU, they will get filtered based on their PORT-ID within each GEM frame's header.

4.3.3.1.3 Downstream FEC assembling process

Assume that there are a contiguous GEM frames ready for a downstream transmission toward ONUs. The GTC payload at a 2.4 Gbps downstream bit rate is able to carry up to 38800 bytes including the GTC header. There are two options for a GTC transmission. Option one, if the FEC is not enabled, then the physical layer's 125µs frame will be the same in size as the GTC frame, but only scrambled, in a GPON G.984 it's a requirement to scramble the downstream traffic. However, if the FEC was enabled then 16 parity bytes will be added after each 239 bytes, starting from the header, the new FEC word size equals 255 bytes (16 + 239 = 255 bytes), therefore the new frame could only carry 152 complete FEC words (255 bytes each) and the remaining 120 bytes could carry a 104-byte fragment plus 16-parity bytes, again it will be scrambled then sent toward the ONUs.

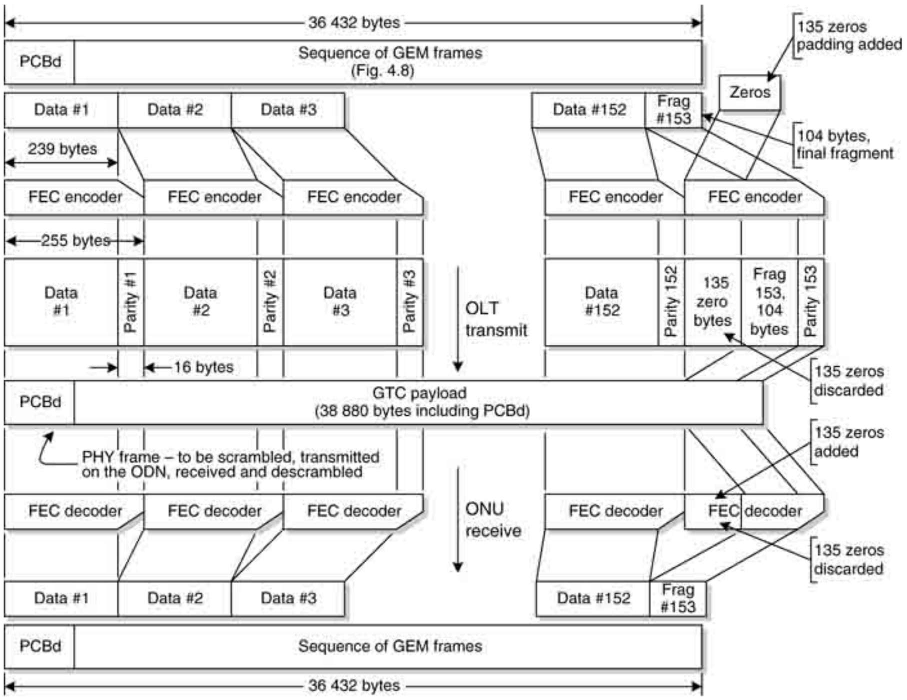


Figure 4-14 [10] GPON Downstream FEC assembling process

When a GTC frame received at the ONU, first it will be descrambled, then it will be checked if the FEC bit is enabled, if yes then it the ONU will perform an FEC check and remove all the 16 parity bytes from the GTC frame, this will reconstruct the same contiguous GEM frames (36432 bytes) without the FEC words that we started with at the OLT. Figure 4-14 illustrates the FEC assembling process of the downstream frame. [10]

4.3.3.2 Upstream framing

Similar to the downstream, the upstream GTC layer frame is a 125 µs in length, see Figure 4-7, thus the frame size varies according to the bit rate, and it could be either 19440 bytes or 38880 bytes at 1244.16 Mbps or 2488.32 Mbps respectively. Figure 4-15 shows the GTC framing structure, each upstream burst has only one physical layer overhead (PLOu), and one or more allocation interval assigned to each T-CONT (Alloc-ID) as specified by the OLT downstream BWmap. Each allocation could have a PLOAMu overhead and/or DBRu overhead, as specified by the OLT downstream flags, if a flag set for either one of them means that the ONU must include the requested message in the following allocation interval's burst, however a PLOAMu must only be requested by the OLT at a BWmap allocation that is addressed to a default Alloc-ID. Nonetheless, a DBRu could exist at any allocation interval as long as the flag set by OLT (not necessary only with default an Alloc-ID).

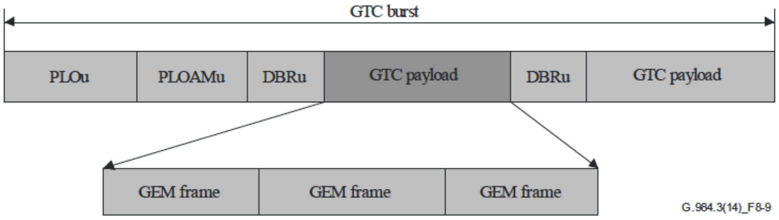


Figure 4-15 Upstream GTC Burst framing structure

Each Burst will begin with a physical layer overhead (PLOu), starts after a specific guard time used for "burst isolation" and delineation from other bursts within the same GTC frame, there will be no gaps between allocation intervals (T-CONTS) belongs to the same ONU, in other words if the first allocation interval "stop time" is pointing at Nth byte, then the start time of the subsequent allocation equals N+1.

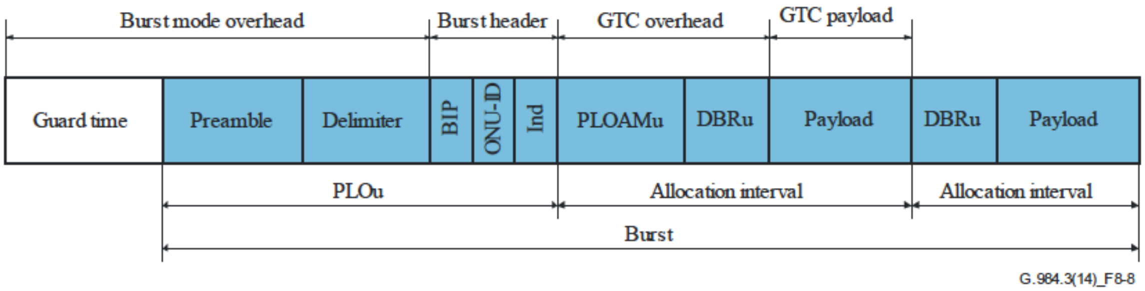
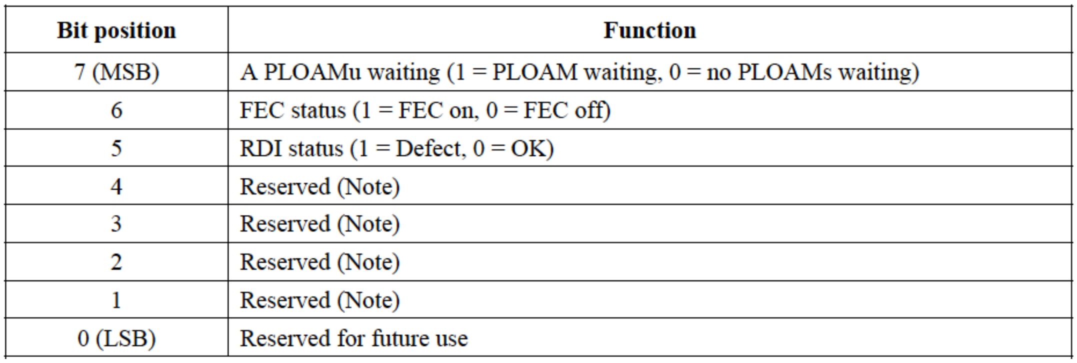


Figure 4-16 upstream burst structure

Figure 4-16 shows the following fields:

- **Preamble and Delimiter**, they're located at the start of the PLOu overhead and before the burst header, they're used for synchronization and byte delineation, the specific format of these two fields is specified by the OLT using a special PLOAMu messages.
- **Bit interleaved parity field (BIP),** this field is used to monitor errors on the link between the OLT and the ONU (the downstream link) after all errors detection/correction is done. It is a 1 byte in size, and it will contain the result of the interleave of all previous bytes belongs to the same ONU from the last occurrence of the BIP field, excluding the BIP field, the preamble and delimiter and any FEC parity bits if implemented. The OLT similarly calculates the BIP field and compares it with the received BIP that was calculated by the ONU, as an indication of number of errors occurred on the link after all error detection and correction is done.
- **ONU-ID field**, is a 1 byte field that contains the sending ONU-ID, where the OLT will check the received ONU-ID field and confirm if the sending ONU is using the correct allocations as dictated by the OLT BWmap.

Table 4-3 Indication bit functions



Bit position	Function
7 (MSB)	A PLOAMu waiting (1 = PLOAM waiting, 0 = no PLOAMs waiting)
6	FEC status (1 = FEC on, 0 = FEC off)
5	RDI status (1 = Defect, 0 = OK)
4	Reserved (Note)
3	Reserved (Note)
2	Reserved (Note)
1	Reserved (Note)
0 (LSB)	Reserved for future use

- **Indication field (Ind),** it's an 8-bit field used by the ONU to set flags that will be seen by the OLT as shown in table 4-3. If a PLOAMu bit was set, then the OLT should enable the PLOAMu for this ONU in the next upstream BWmap allocation by setting the PLOAMu flag. If a remote defect indication (RDI) bit was set, then it will trigger an RDI alarm at the OLT, which indicates that the received downstream frame was invalid (defected or corrupted), the RDI alarm stay ON unless the ONU sets it to "0" when it receives an error-free downstream frame.

- **PLOAMu**, is a physical layer operation, administration and maintenance field, an optional 13-bit field that is used for a PLOAM upstream messages exchange between the ONU and the OLT, only sent by the ONU when enabled by the OLT flag. And it should be always associated with the default Alloc-ID. See Figure 4-17.
- **Power leveling sequence (PLSu),** this is an obsolete field and should be always zero.
- **Dynamic bandwidth report (DBRu),** it contains two subfields as per the following:
  - **DBA report field,** is used to report the buffer status for the traffic bearing entity (T-CONT), the sending decision and the size of this field is determined by the OLT through DBR flag bits, as explained in 4.3.3.1.1.
  - **Cyclic redundancy check field (CRC),** it's a CRC error detection and correction for the DBRu field, it can correct only a one-bit error.

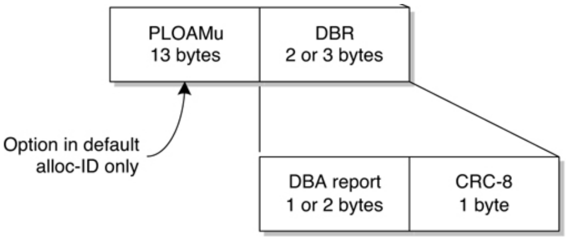


Figure 4-17 upstream GTC overhead per allocation interval

4.3.3.3 GEM Frame

Figure 4-18 illustrates the GEM frame structure as described in ITU-T G.984.3 recommendations.

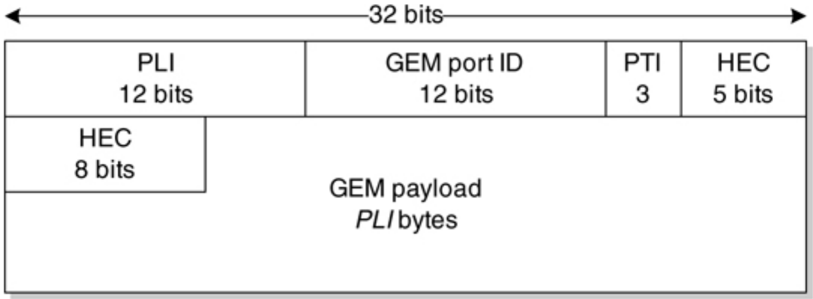


Figure 4-18[10] GEM frame

4.3.3.3.1 GEM header

It is five bytes long and it has the following fields:

Payload length indication (PLI), a 12-bit field specifics the GEM payload size in bytes, thus the GEM payload is limited to a maximum of 4095 bytes, if the encapsulated Ethernet frame was larger in size, then it will be fragmented and carried by multiple GEM frames. The minimum payload the GEM allowed to carry is 1 byte Ethernet fragmented frame. It's also used to indicate the start of the sequent GEM header.

GEM port ID, is a 12-bit field used to identify the final destination port where the traffic will go, the maximum number of ports is 4095. In the upstream, one or more ports could exist in one allocation interval (T-CONT).

Payload Type Indication (PTI), is used to identify the type of a payload, historically this was necessary for the ATM SDU fragmentation, however after obsoleting ATM this field is not really useful. Later it was dropped from XGPON frame. [10]

HEC, the field head error check is s 13-bit field used for error checking and correcting code that covers only the GEM header as the payload is not included. As discussed earlier, the only way to locate the next GEM frame in a contiguous GEM frames inside a GTC frame payload is to look inside each GEM frame header and identify the PLI value in order to determine the beginning of the subsequent GEM frame. See Figure 4-19. If the header found to be corrupted beyond correction (more than 1 bit error), then the byte synchronization will be lost, and there is no way to locate the next GEM frame beginning, unless the receiver successfully recovers the synchronization, for more details regarding synchronization reacquiring process refer to ITU-T G.984.3 8.3.2. [11]

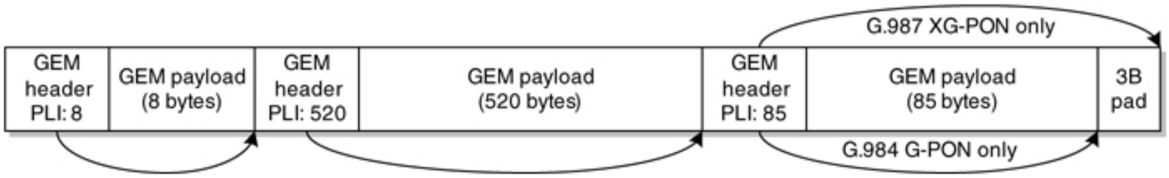


Figure 4-19 [10]Contiguous GEM frames

4.3.3.3.2 GEM Payload

It will carry a payload size equals to the PLI value, however the maximum payload size is 4095 bytes, and the minimum is only one byte. Each GEM payload carries only one complete Ethernet frame or a fragment, though it can't carry more than one Ethernet frame at any time. The Ethernet frame is mapped inside the GEM payload after stripping the preamble and delimiter as shown in Figure 4-20. Also, an MPLS packet (label stack and MPLS Payload) could be mapped inside the GEM payload with possibility of fragmentation if the size exceeds 4095 bytes.

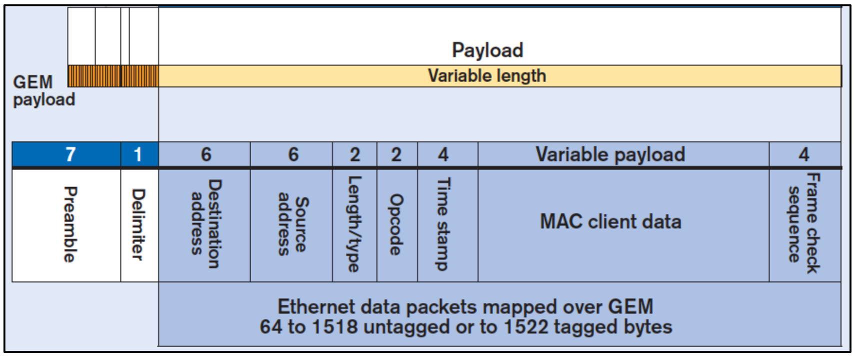


Figure 4-20 [3] Mapping Ethernet frame into GEM payload

Another reason for fragmentation is that in the downstream direction the physical frame is only 125 µs, if a TC layer frame was not fully sent during a 125 µs, then it will be fragmented with an exception that the fragment is a minimum of 1 byte in size, otherwise an empty GEM frame will be sent. In the upstream the rule is similar, with allocated transmission time, if the allocated time was too short to send the complete SDU frame then it will be fragmented with the same exception of a minimum of one byte fragment size, otherwise the SDU layer has to wait for the next allocation interval with the same designated port ID to send the SDU packet.

Empty GEM frame, also called an idle frame, is all zeroes sequence, no heading, nothing, but all zeroes, technically it's only an empty header, a 5-byte header filled with sequence of zeroes, this means that the idle GEM frame can't be part of a contiguous GTC frames, so this means the remainder of the 125 µs GTC frame will be empty too. [10]

Figure 4-21 shows mapping of GEM frames into GTC payload, it is the same process for both the downstream and the upstream traffic.

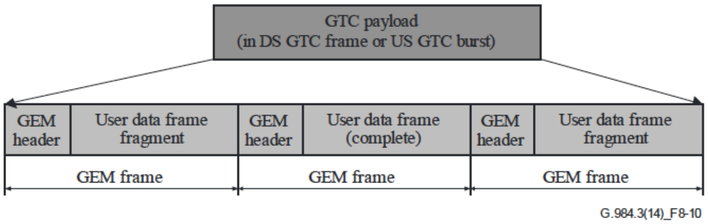


Figure 4-21 [11] GEM frames mapping into GTC payload

4.3.4 ONU activation process

When an ONU is initially powered up, it will go through predetermined sequence of events and states where it exchanges information with the OLT before it becomes fully operational.

The detailed process is well covered in G.984.3 recommendation, this process is fully controlled by the OLT. Figure 4-22 shows the state of an ONU during activation, there are three phases, synchronization phase also called parameter learning, Serial number acquisition and ranging phase.

It’s important to understand that the ONU will not be able to send and receive user data until it goes through the activation process with the OLT, where it gets an ONU-ID assigned by the OLT, and it gets registered by using its vendor serial number within the OLT, it will also receive bandwidth mapping for its T-CONTs though the OLT downstream GTC header (PCBd).

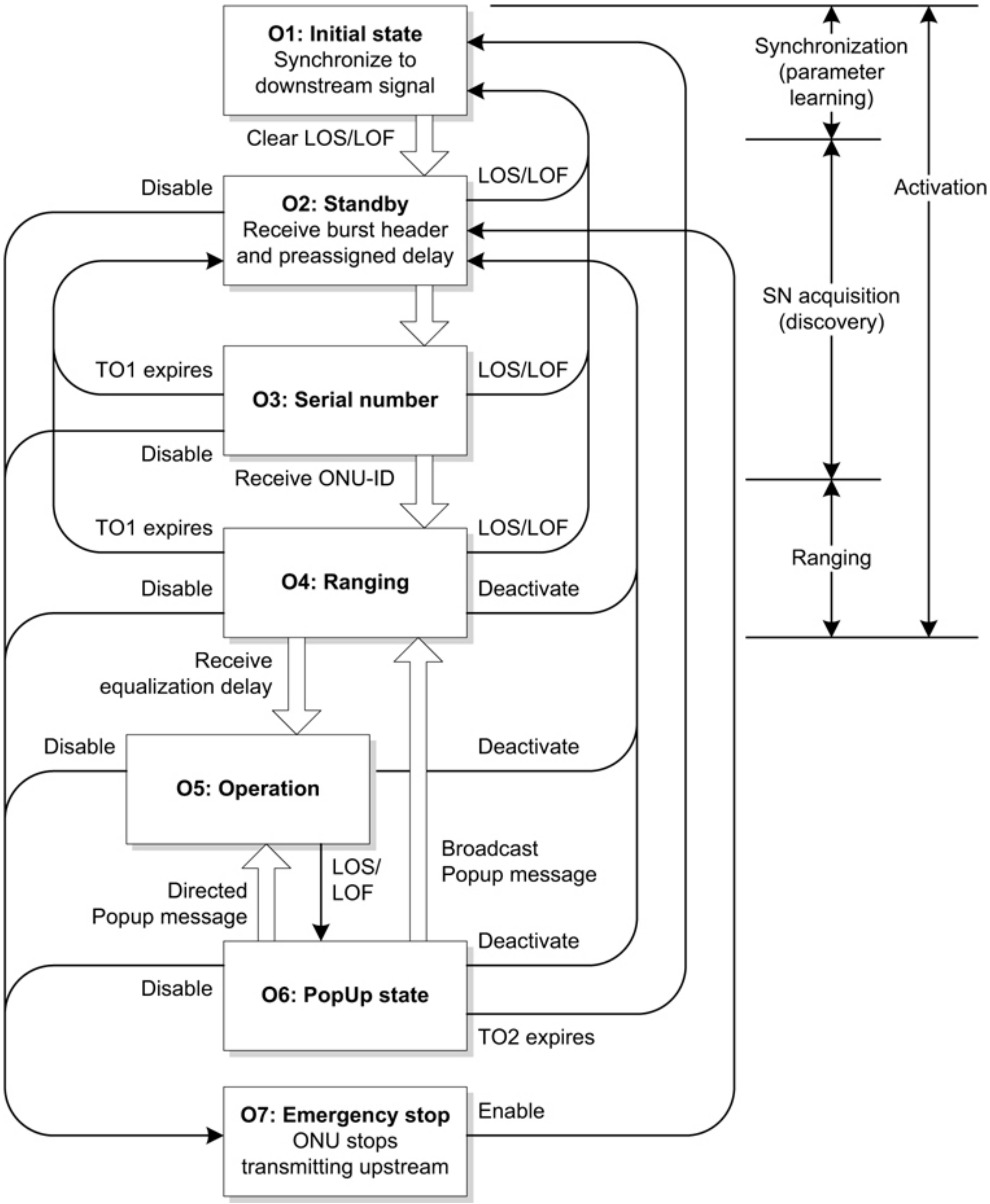


Figure 4-22 [11] ONU states diagram

During this process, the ONU adjusts the upstream signal power level, the OLT will measure the round-trip delay (RTD) which is defined as the time interval between sending a downstream frame and receiving the upstream response for the same allocation ID, and through RTD the OLT will calculate equalization delay (EqD) and communicate it to the ONU at the ranging phase. EqD is the delay that each ONU needs to add before it transmits an upstream frame, due to differences in distance and signal propagation time between ONUs, the EqD is added to make sure that the OLT will receive all upstream bursts on a correct timing without interfering with one another. [11]

During the serial number and ranging phases, there is a “quite window” interval temporary granted by the OLT to prevent all operational ONUs from sending their upstream frames, only unknown ONUs to be discovered are allowed to transmit, this will prevent possible collision between both of them. During the quiet phase the OLT will broadcast a request directed to the default allocation ID “254” which is recognized by the unknown ONU’s, and they will respond by sending their serial numbers in order to be registered and designated an ONU-ID. [11]

4.3.5 Transmitter block diagram

As shown in Figure 4-23, this nice diagram mentioned in [ITU-T G.984.3 A.8], explains the full path that a data unit traverse within a PON system, in both directions, the downstream and the upstream showing the frame life cycle. The importance of this diagram is that it summarizes what have been discussed so far in GTC layer.

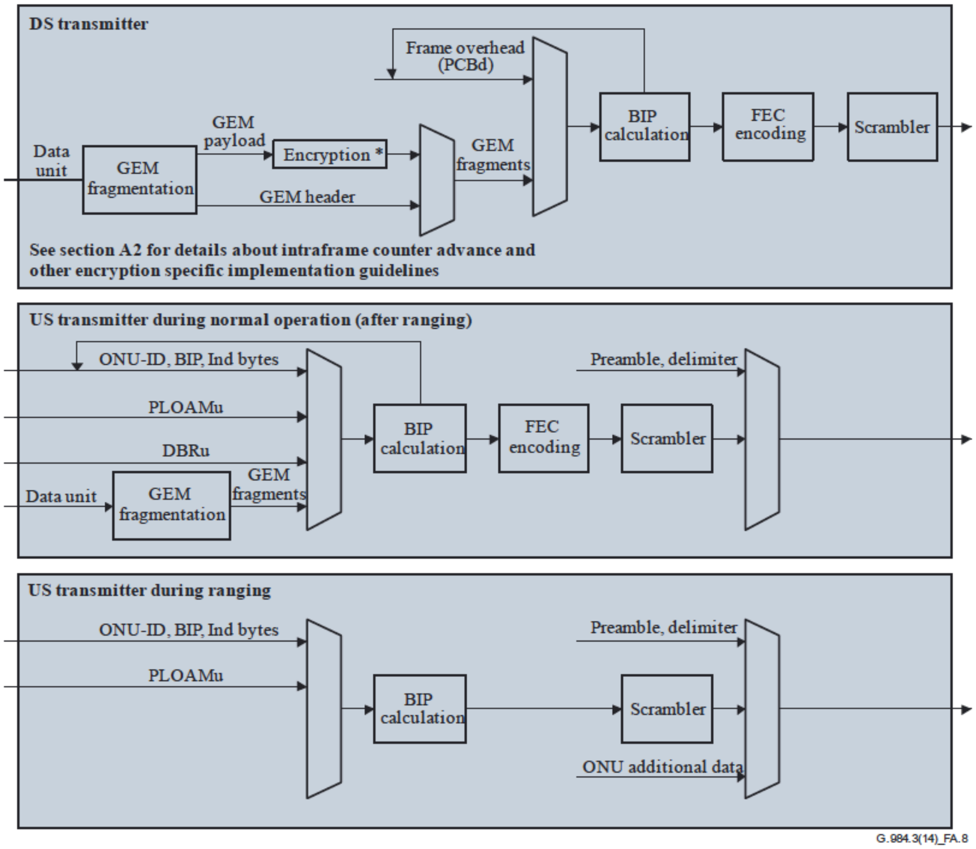


Figure 4-23 Transmitter block diagram

4.3.6 Performance Monitoring and alarms

Performance monitoring in GPON is a non-stop process performed by the OLT and the ONU to measure system’s overall performance and detect any link failures, as a result it will trigger specific alarms, and some alarms will provoke series of actions to resolve the situation as shown in Figure 4-24.

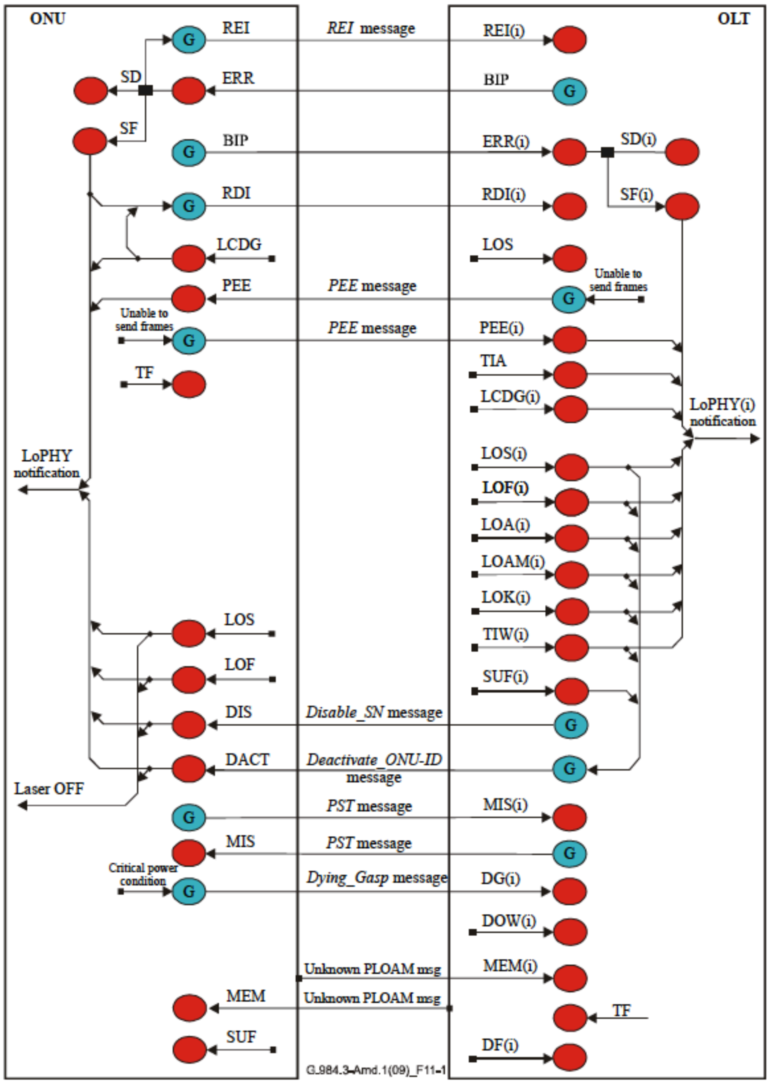


Figure 4-24 [11] Alarms triggered at OLT and ONU

4.4 System security

In a PON there is a security concern regarding the downstream data, since each ONU will receive the GEM frames that belongs to other users. However, the situation is different in the upstream where the upstream bursts is physically separated by their own fiber and can only be seen by the OLT, furthermore to intercept the upstream signal by an attacker is nearly impossible in practice, which makes the upstream link safer for encryption keys transmission (e.g. shared public keys).

Therefore, to resolve the downstream security threat of exposing users' data, the GPON standard employs advanced encryption standard (AES) to encrypt the data, where a unique discreet encryption key is used at the OLT to encrypt the downstream GTC frames before sending them, also the same key is used at the ONU to decrypt the data. The key is generated by the ONU when requested by the OLT in a special PLOAM messages, where a quite robust scheme is followed to make sure the correct key is received by the OLT, if the operation is not successful, then the OLT will request the key again, after three failed attempts the ONU will be deactivated by the OLT and a loss of key synchronization Alarm will flash.

ITU-T G.984.3 also defines an encryption key switch-over process, where the OLT under a special event requests the ONU to regenerate a new key through a PLOAM channel communication, the switchover process could happen in response to special events such as the ONU loss of synchronization (LOS/LOF). [11]

4.5 ONU Management and Control Interface (OMCI)

As discussed earlier in the discussion of management and control plane shown in Figure 4-5, the OMCI is a higher layer protocol used to manage the ONU equipment and subscriber services. The OMC station could be installed in an OLT or remotely in another data-center or even in the clouds (i.e. as a virtual machine), either way, there should be a dedicated virtual connection between the OMC station and the ONU. The OMCI Protocol data unit (PDU) is transported over a PON, encapsulated inside a GEM payload, however there should be a special port-ID dedicated for this purpose that is assigned by the OLT and communicated to the ONU during activation. The OMCI adapter at the OLT/ONU, is responsible for frames filtration based on their Port-ID, it's also performs the PDU encapsulation into a GEM frame payload before it is sent to the GTC layer, and upon receiving the GTC frame, the OMCI adapter performs the de-capsulation of PDUs from received GEM frames. Unlike the PLOAM which is used in both multi/unicast, the OMCI is a unicast protocol. [11] [1]

OMCI makes the GPON more complicated protocol to implement than EPON for an example, since the later standard do not specify OMC system where it is left to the vendor to decide how to implement it. Some GPON vendors do not implement the whole OMCI protocol, and prefer to implement only crucial functions. The OMCI is fully covered in ITU-T G.984.4 and G988. [1]

4.6 GPON protection

GPON standard has defined two ways of system protection, in case a link failure has occurred, in order to provide an alternative signal path without losing the traffic. The two types of protection defined as per the following:

4.6.1 Type B protection

As shown in figure 4-25, two optical feeders between the OLT and the splitter are installed in an active/standby setup, the OLT will keep monitoring performance on the active link and immediately switches over to a standby link in case of a failure. In this type of protection, there are two different ports used at the OLT.

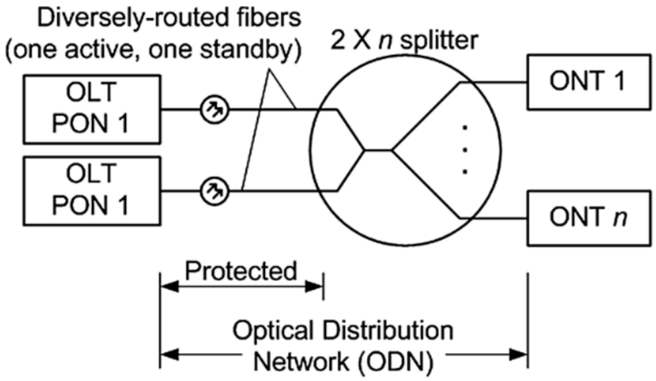


Figure 4-25 Type B protection

4.6.2 Type C protection

It's similar to Type B, however there will be a redundancy exists in the fiber between the splitter and the ONTs, where a two-fiber cables are installed for every ONT. Unlike type B, the type C will provide a better protection, if an active link between one or more ONTs goes down, then the system will switch to the standby fiber and continue smoothly without noticeable service interruption, however the tradeoff is the extra protection that comes with the extra fiber and additional required management. Figure 4-26 shows Type C protection.

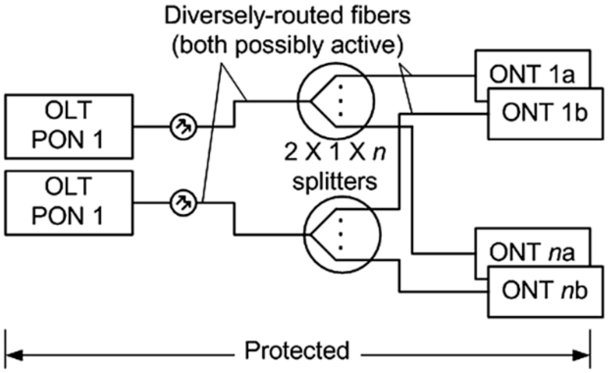


Figure 4-26 Type C protection

It’s also possible that in some cases to configure the system so that it shares the traffic between both fibers as a way of traffic balancing. It makes sense to install standby cables on a different path to avoid both fibers physical damage.

4.7 GPON summary

This chapter discussed the GPON system architecture and its key characteristics, the OLT and the ONU functional blocks. Also, it explained the protocol stack for the GTC layer and its sub layer functions such as, the GTC framing and the GEM framing, both in the downstream and the upstream, it also explained each frame structure and how to map a PDU into the GEM payload. The ONU activation process was also explained. Other GPON topics such as the system encryption, OMCI protocol, the two standard ways of protection was highlighted.

Chapter 5 10Gigabit- PON (XG-PON & XGS-PON)

5.1 Introduction

ITU-T used a generic term of the next generation PON (NG-PON) systems beyond GPON. The term NG-PON includes two branches, the NG-PON1 that uses same optical distribution network (ODN) from GPON, and NG-PON2 where the ODN might require some design changes depends on preexist technologies. [12]

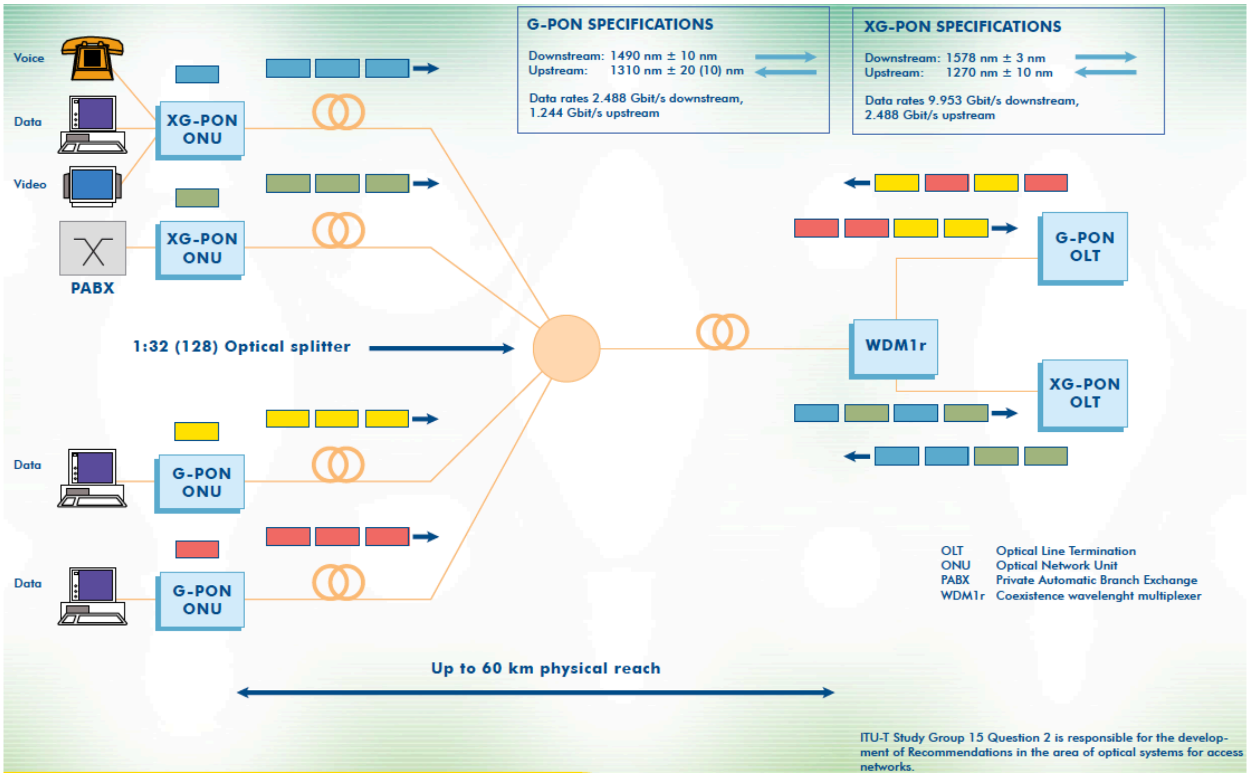

G-PON SPECIFICATIONS
Downstream: 1490 nm ± 10 nm
Upstream: 1310 nm ± 20 (10) nm
Data rates 2.488 Gbit/s downstream, 1.244 Gbit/s upstream
XG-PON SPECIFICATIONS
Downstream: 1578 nm ± 3 nm
Upstream: 1270 nm ± 10 nm
Data rates 9.953 Gbit/s downstream, 2.488 Gbit/s upstream
Voice
Data
Video
PABX
XG-PON ONU
XG-PON ONU
G-PON ONU
G-PON ONU
Data
Data
1:32 (128) Optical splitter
G-PON OLT
WDM1r
XG-PON OLT
OLT Optical Line Termination
ONU Optical Network Unit
PABX Private Automatic Branch Exchange
WDM1r Coexistence wavelenght multiplexer
Up to 60 km physical reach
ITU-T Study Group 15 Question 2 is responsible for the development of Recommendations in the area of optical systems for access networks.

Figure 5-1 [13] XG-PON Access network system

XG-PON is the first technology on the path of NG-PON, it's considered a subclass of NG-PON1. Similar to any new standard, ITU-T study group 15 designed the XG-PON standard to be compatible with the existing GPON. Figure 5-1 describes the co-existence of XG-PON with GPON.

XG-PON downstream bit rate capability is 10 Gbps (9.953 Gbps). In the upstream it comes with two options, a 2.5 Gbps upstream bit rate system called XG-PON1, and an XG-PON2 symmetrical system features 10 Gbps upstream bit rate which could be referred to as "XGS-PON" (the S after XG means symmetrical), please note that G.987.x series only covers XG-PON1 (referred to XG-PON by ITU), and for XG-PON2 (XGS-PON by ITU) is covered by G.9807.x recommendations series.[14] . To avoid any confusion, from now on, this report will follow ITU-U naming scheme, instead of using XG-PON1 and XG-PON2, it will use XG-PON to refer to XG-PON1, and XGS-PON to refer to XG-PON2, sometimes we will use the name "NG-PON" to refer to technologies came after GPON includes NG-PON2. Table 5-1 shows related standards under the two series.

Table 5-1 [9] XG-PON and XGS-PON standards published by ITU-T Group 15

Recommendation code	Description
G.987	10-Gigabit-capable passive optical network (XG-PON) systems: Definitions, abbreviations and acronyms
G.987.1	(XG-PON) systems: General requirements
G.987.2	(XG-PON) systems: Physical media dependent (PMD) layer specification
G.987.3	(XG-PON) systems: Transmission convergence specification
G.9807.1	10-Gigabit-capable symmetric passive optical network (XGS-PON)
G.9807.2	(XGS-PON) Reach Extension

5.2 Reference configuration

XG-PON and XGS-PON maintains the same high level reference configuration in BPON (G.983.x), and GPON (G.984.x), as shown in figure 5-2.

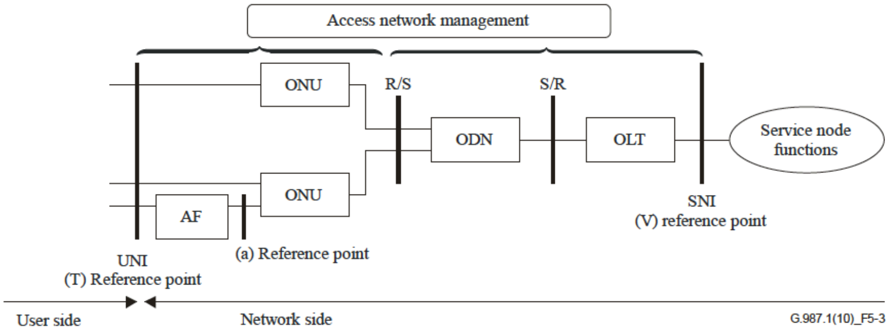


Figure 5-2 [14] [15] High level reference configuration of XG & XGS PON

A simple generic optical access network (OAN) applies for XG & XGS-PON is shown in figure 5-3, note that reach extenders (RE), defined as an active component that were first introduced in [ITU-T G.984.6], which make an optical distribution network (ODN) no longer strictly passive as it was in legacy GPON (ITU-T G.984.0 and G.983.0). However, to distinguish between legacy passive ODN and the new partially active ODN, the term "simple ODN" is used to refer to a fully passive single ODN, and a composite ODN to refer to an ODN consists of an active component that is interconnecting grouped passive ODN segments (ODS). [14] [15]

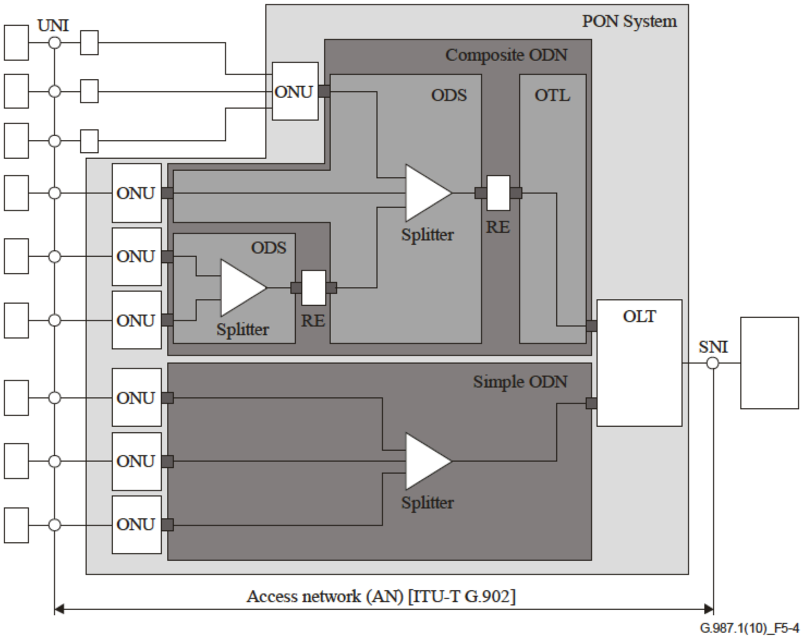


Figure 5-3 [14] generic XG & XGS-PON architecture

5.2.1 Optical network distribution (ODN) architectures

There are many scenarios for the ODN architecture that enables coexistence between XG-PON, XGSPON, GPON and any other legacy services such as Video overlay. In order to achieve the coexistence, a reserved wavelength band is defined which referred to as an enhancement band. Figure 5-4 and 5-5 shows the reference architecture for XG-PON and GPON coexistence that could be achieved with different methods. [14] [15]

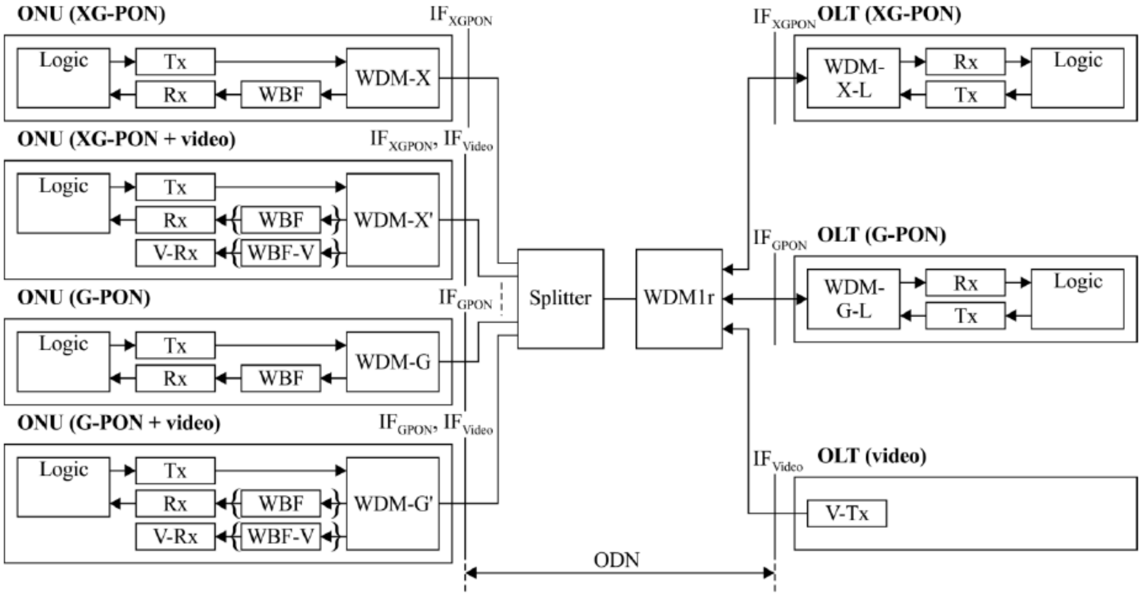


Figure 5-4 [14] XG-PON and GPON coexistence through WDM1r

Both reference architectures will provide the same coexistence using two different elements, a wave division multiplexer (WDMr1) as shown in figure 5-5, and an Optical splitter as shown in 5-6. The WDMr1 is a wave division multiplexer filter which couples and decouples XG-PON and GPON signals in both directions, as well as an optional video overlay signal. Be aware that in case of a splitter used then an extra filtering at the OLT (WDM-X-L and WDM-G-L) is needed to achieve the required level of signal isolation. Also, a wavelength blocking filter (WBF) is used at the ONU to eliminate any signal interference and to make sure an almost square signal is transmitted or received. [14] [15]

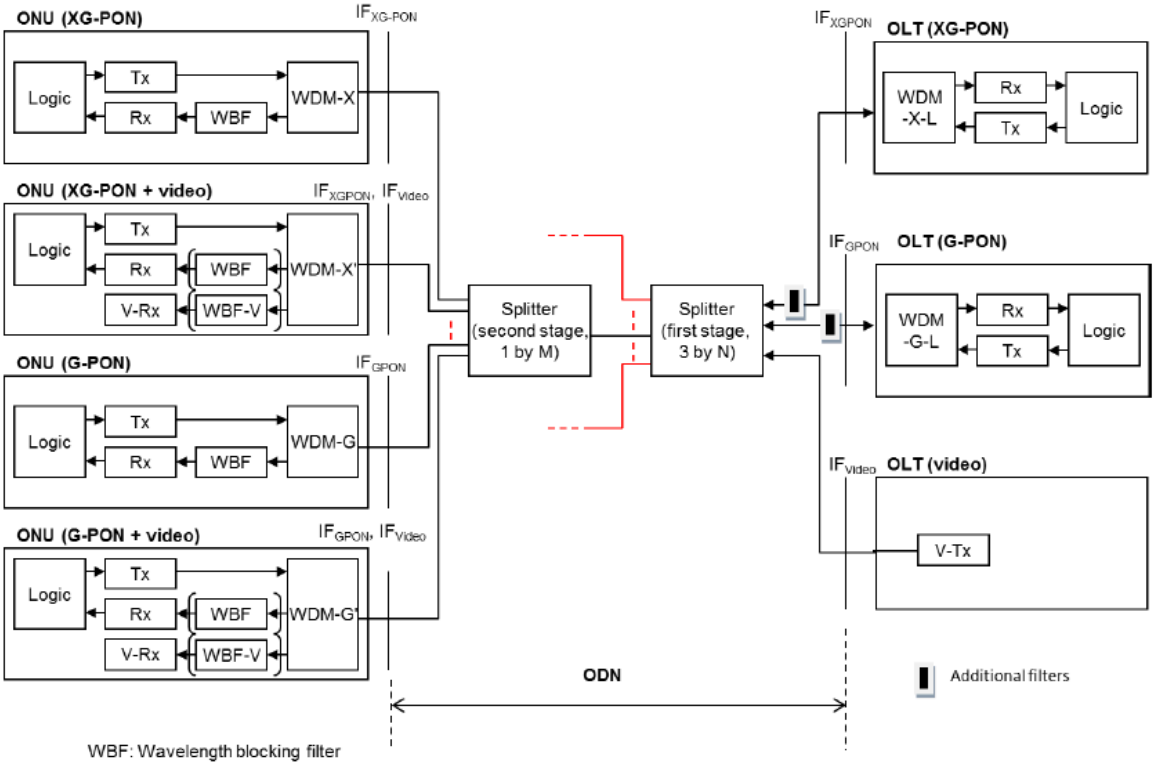

ONU (XG-PON)
$\mathrm{IF}_{\mathrm{XG\text{-}PON}}$
Logic
Tx
Rx
WBF
WDM-X
ONU (XG-PON + video)
$\mathrm{IF}_{\mathrm{XGPON}}$, $\mathrm{IF}_{\mathrm{Video}}$
V-Rx
WBF-V
WDM-X'
ONU (G-PON)
$\mathrm{IF}_{\mathrm{GPON}}$
WDM-G
ONU (G-PON + video)
$\mathrm{IF}_{\mathrm{GPON}}$, $\mathrm{IF}_{\mathrm{Video}}$
WDM-G'
Splitter (second stage, 1 by M)
Splitter (first stage, 3 by N)
$\mathrm{IF}_{\mathrm{XGPON}}$
OLT (XG-PON)
WDM -X-L
$\mathrm{IF}_{\mathrm{GPON}}$
OLT (G-PON)
WDM -G-L
$\mathrm{IF}_{\mathrm{Video}}$
OLT (video)
V-Tx
ODN
Additional filters
WBF: Wavelength blocking filter

Figure 5-5 [14] XG-PON and GPON coexistence through splitter

Regarding XGS-PON, there are another three reference configurations for coexistence scenarios as shown in figure 5-6, 5-7 and 5-8.

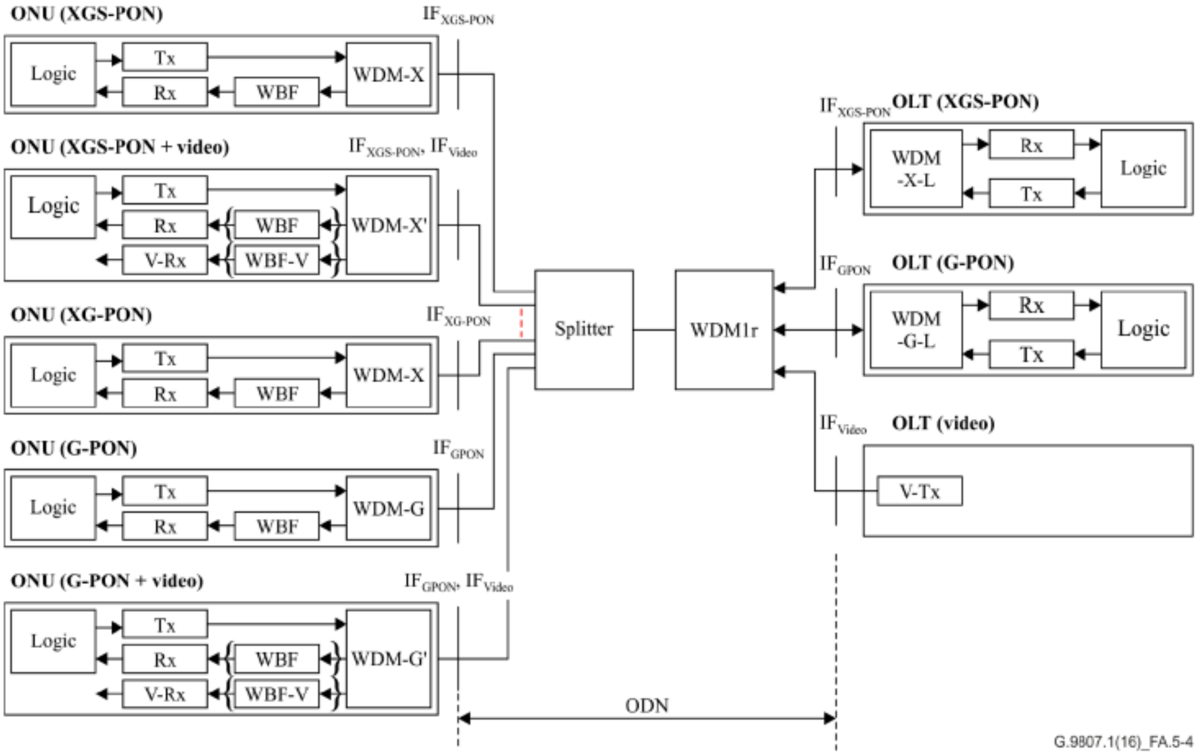


Figure 5-6 [15] XGS-PON coexistence with GPON through WDM1r using XGS-PON basic wavelength set

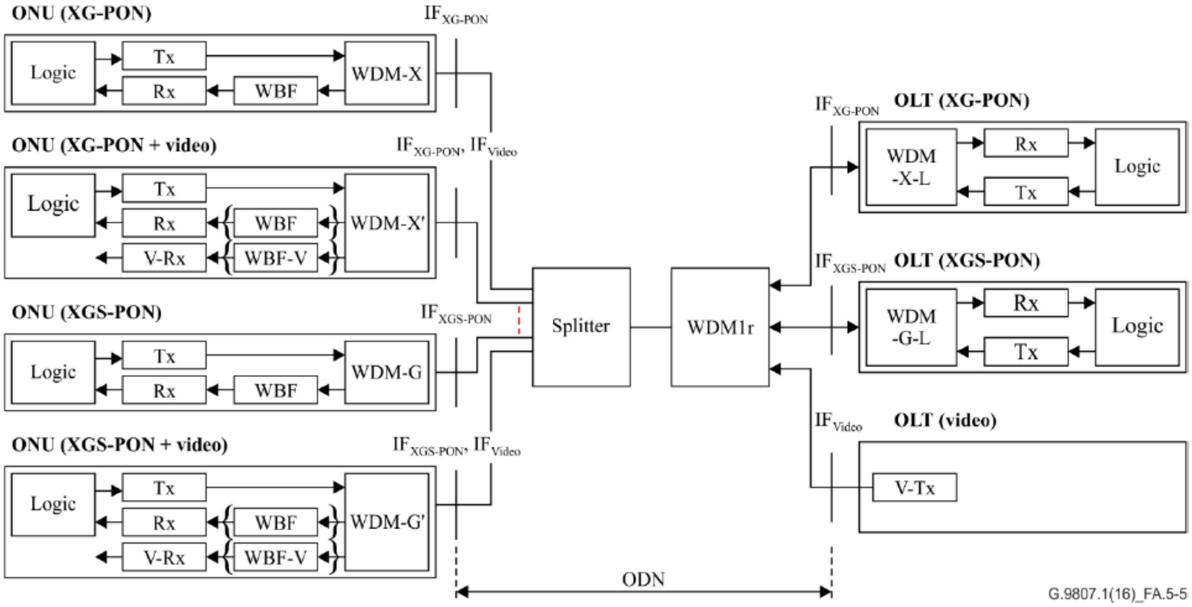


Figure 5-7 [15] XGS-PON coexistence with XG-PON through WDM1r using XGS-PON optional wavelength set

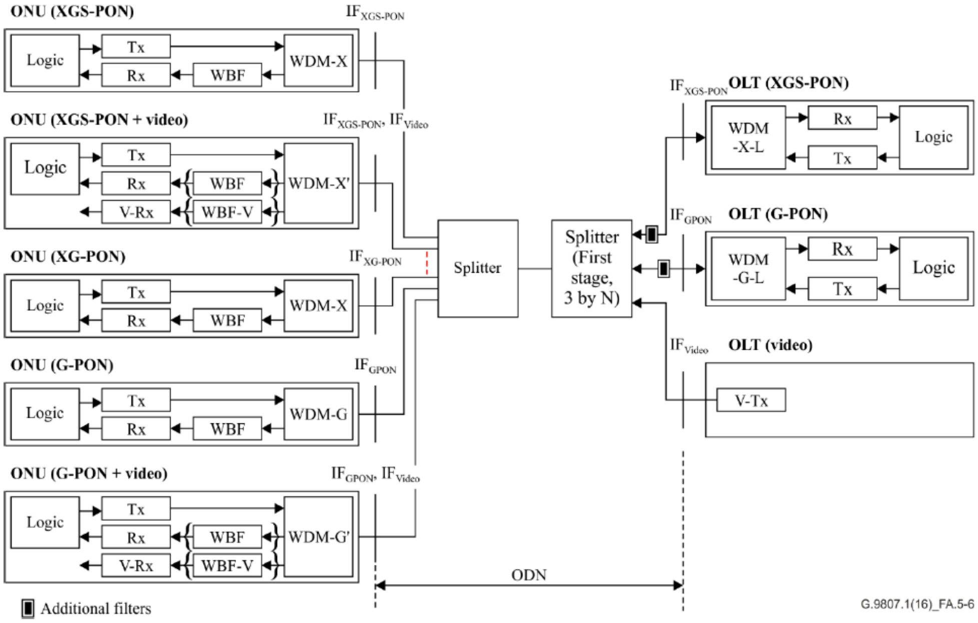


Figure 5-8 [15] XGS-PON coexistence with GPON through splitter using XGS-PON basic wavelength set

In the three figures above (5-6, 5-7 & 5-8), note that WBF is used when XG-PON, XGS-PON and GPON are combined with video overlay on the same ODN, in order to make sure that the required level of signal isolation is achieved. In figure 5-6 and 5-8 the XG-PON ONU is supported by the XGS-PON OLT (this can't be the opposite, means XG-PON OLT cannot support XGS-PON ONU), where TDM in the downstream and TDMA in the upstream is used to multiplex XGS-PON and XG-PON ONU signals on the XGS-PON OLT port. [15]

5.2.2 Operating Wavelength

To enable the co-existence of GPON, XG-PON, XGS-PON and the future NG-PON2 technologies, a reserved wavelength bands (referred to as "enhancement bands") was redefined to accommodate any additional services such as the video overlay, the additional band can be used to carry an upstream or downstream signals as needed. Please note wavelength allocation to be discussed is happening on a single fiber system (e.g. a single fiber carries the upstream & the downstream traffic). [14] [15]

5.2.2.1 Operation wavelength for GPON and XGPON

The XG-PON downstream band, also known as the "basic band", is located at 1575-1580 nm, the upstream band is located at 1260-1280 nm. Figure 5-9 shows the wavelength allocation which contains the XG-PON bands and the enhanced bands that may carry the GPON and a video overlay. It also shows a guard band between XG-PON and enhancement bands to avoid any interference between these bands. However, since the signal is not perfectly square and even with the guard band a small component of the signal may cause interference, therefore using a wave blocking filter (WBF) is required to filter out any signal components that extends out of the guard band. [14]

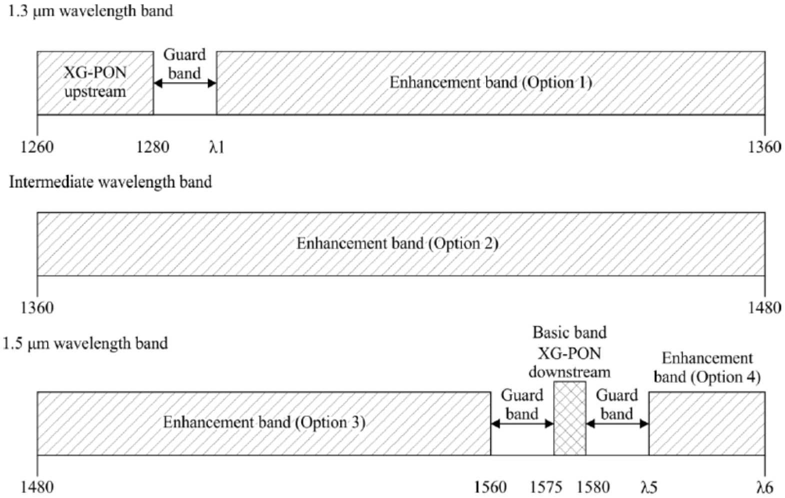

1.3 µm wavelength band
XG-PON upstream
Guard band
Enhancement band (Option 1)
1260
1280
λ1
1360
Intermediate wavelength band
Enhancement band (Option 2)
1360
1480
1.5 µm wavelength band
Basic band XG-PON downstream
Enhancement band (Option 4)
Guard band
Guard band
Enhancement band (Option 3)
1480
1560
1575
1580
λ5
λ6

Figure 5-9 [14] wavelength allocation in XG-PON

Table 5-2, shows the recommended wavelength values for each band, taking into consideration the achievable and practical guard bands.

Table 5-2 [14] wavelength allocation as described in figure 5-9

Item	Notation	Unit	Nominal value	Application
XG-PON upstream				For use in XG-PON1 upstream
Lower limit	-	nm	1260	
Upper limit	-	nm	1280	
Enhancement band (option 1)				For use in G-PON upstream (Reduced option: 1290-1330 nm).
Lower limit	λ1	nm	1290	
Upper limit	-	nm	1330	
Enhancement band (option 2)				Reserved for future use.
Lower limit	-	nm	1360	
Upper limit	-	nm	1480	
Enhancement band (option 3)				For use in G-PON downstream (1480-1500 nm) and/or video distribution service (1550-1560 nm).
Lower limit	-	nm	1480	
Upper limit	-	nm	1560	
XG-PON downstream (Basic band)				For use in XG-PON1 downstream. Please note for outdoor application 1575 – 1581 nm is used.
Lower limit	-	nm	1575	
Upper limit	-	nm	1580	
Enhancement band (option 4)				Reserved for future use.
Lower limit	λ5	nm	TBD	
Upper limit	λ6	nm	TBD to 1625	

5.2.2.2 Operating wavelength for XGS-PON

There are two operating wavelength options for XGS-PON, the basic wavelength set, which is the same as XG-PON wavelengths basic band (1575-1580 nm downstream, and 1260-1280 nm upstream), where TDM in downstream and TDMA in upstream is used to accommodate the coexistence of XGS-PON ONU and XG-PON ONU signals.

The other option is called optional wavelength set, which is the same as the G-PON wavelength (1490 nm/1310 nm), however this option only used when ODN does not have any GPON deployment that may already exists. [15]

5.3 Migration scenarios

One main requirement for any NG-PON system, including XG-PON and XGS-PON, is to provide a smooth migration for the existing GPON users towards the new technology. In case of a brownfield, where a legacy GPON infrastructure already deployed and XGPON OLT/ONU will coexist with the GPON OLT/ONU, or in the case the XG-PON is the brown technology and the XGS-PON is the new technology to coexist, this requires that the new technology should share the same single fiber that's already deployed. Therefore, in this case the new technology is required to support the old technology users with same level of service quality, in other words it should support same GPON or XGPON legacy services while implementing the new services. However, in case of a green field, where GPON or XGPON doesn't exist, and the service provider decided to replace the copper infrastructure with an XG-PON or an XGS-PON, then there are no coexistence requirements. [14] [15]

The following coexistence scenarios will be discussed:

- Scenario 1, as shown in figure 5-10, the GPON and an XG-PON coexist through a WDM1r with an optional video overlay.
- Scenario 2, as shown in figure 5-11, a coexistence of a GPON and an XGS-PON with a video overlay (basic band)
- Scenario 3, as shown in figure 5-12, the coexistence of an XG-PON and an XGS-PON through a WDM1r, where the XGS-PON will reuse the GPON base wavelength.
- Scenario 4, as shown in figure 5-13, where an XG-PON an XGS-PON coexist in a TDMA/TDM scheme.
- Scenario 5, as shown in figure 5-14, a GPON, XG-PON and an XGS-PON coexist

through a WDM1r, where the XG-PON and the XGS-PON coexist in a TDMA/TDM scheme.

5.3.1 Migration from GPON to XG-PON

To achieve a coexistence of an XG-PON and a GPON ONUs and OLTs, a WDM1r is installed to couple/decouple signals as shown in figure 5-10. The Bandwidth allocation should change according to [G.984.5 Amd 1 & G.987.1], though a new bandwidth allocation will be as follows:

- GPON downstream remains the same, 1480-1500 nm (1490 nm).
- GPON upstream, is changed to a reduced band, 1290-1330 nm (1310 nm).
- XGPON upstream, 1260-1280 nm
- XGPON downstream (basic band set), 1575-1580 nm
- Video overlay 1500-1560 nm (1550 nm RF)

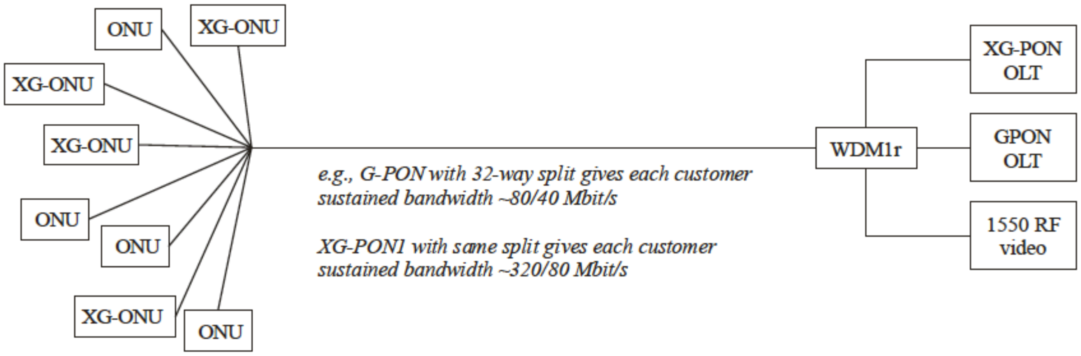

ONU
XG-ONU
XG-ONU
XG-ONU
ONU
ONU
XG-ONU
ONU
e.g., G-PON with 32-way split gives each customer sustained bandwidth ~80/40 Mbit/s
XG-PON1 with same split gives each customer sustained bandwidth ~320/80 Mbit/s
WDM1r
XG-PON OLT
GPON OLT
1550 RF video

Figure 5-10 [14] Coexistence example of GPON and XGPON with video overlay

5.3.2 Migration from GPON to XGS-PON

To achieve the coexistence of XGS-PON and GPON ONUs and OLTs, a WDM1r is installed to couple/decouple signals as shown in figure 5-11. A bandwidth allocation should change according to [G.984.5 Amd 1 & G.9807.1], though a new bandwidth allocation will be as follows:

- GPON downstream remains the same, 1480-1500 nm (1490 nm).
- GPON upstream, changes to a reduced band, 1290-1330 nm (1310 nm).
- XGS-PON upstream, 1260-1280 nm
- XGS-PON downstream (basic band set), 1575-1580 nm
- Video overlay 1500-1560 nm (1550 nm RF)

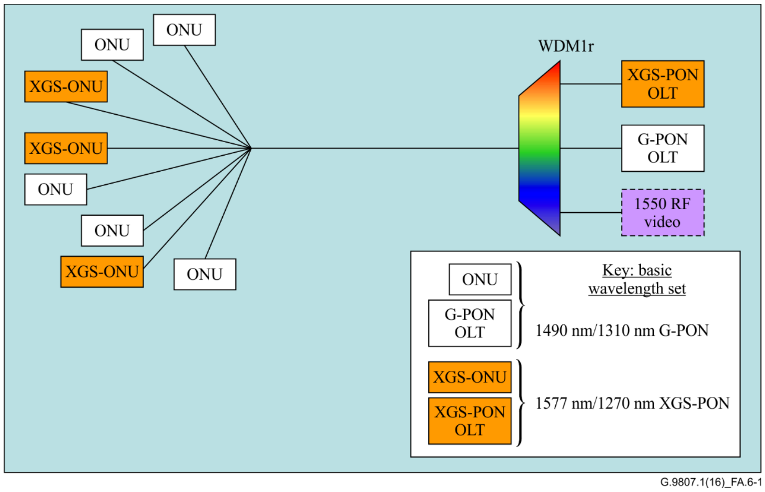


Figure 5-11 [15] Coexistence of GPON and XGS-PON with video overlay (basic band)

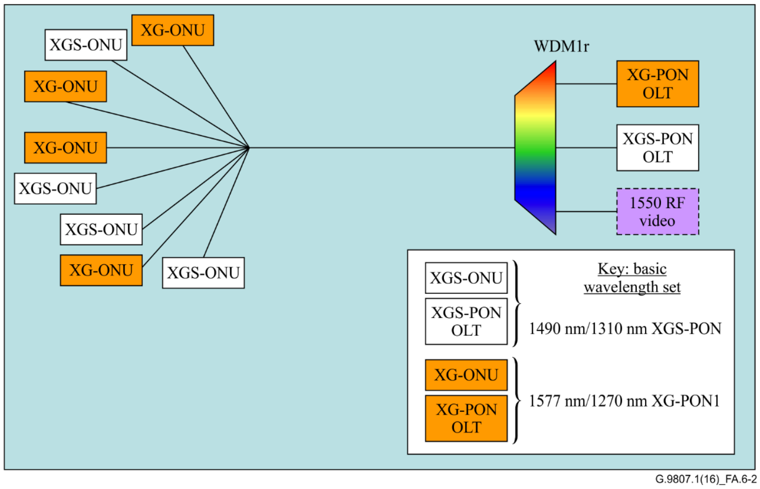


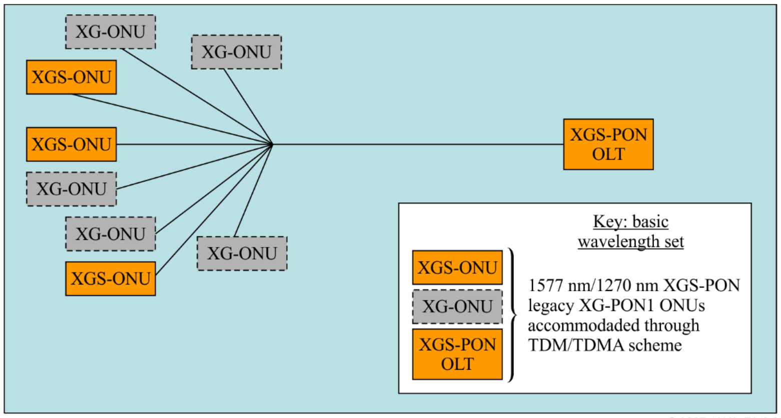
Figure 5-12 [15] Coexistence of XG-PON and XGS-PON (use GPON band) with video overlay.

5.3.3 Migration from XG-PON to XGS-PON

In this scenario XGS-PON and XG-PON ONUs are served by XGS-PON OLT as shown in figure 5-13. Both are using 1577 nm down/ 1270 nm up, though TDM / TDMA modes are used to accommodate signals in both directions.

Figure 5-15 shows the fifth possible scenario, coexistence of GPON, XG-PON, XGS-PON and optional video overlay, in this case XGS-PON and XG-PON are configured similar to scenario 4, a TDM/TDMA scheme used on XGS-PON OLT port to serve XGS-PON and XG-PON ONUs. WDM1r will couple/decouple all signals on a single fiber. [15] Wavelength allocation plan will look like this:

- GPON, 1490nm /1310 nm (GPON reduced band)
- XGS-PON and XG-PON, 1577nm/1270nm (XG-PON basic wavelength set)



G.9807.1(16)_FA.6-3

Figure[15] 5-13 XG-PON and XGS-PON coexist, both are using basic wavelength set.

5.4 NG-PON services

Moving from legacy circuit switched networks towards packet-based high speed broadband next generation networks (NGN) which planned to be a flexible and a seamless transmission in order to not to disturb ongoing businesses and services, and to provide an attractive business model for service providers to follow. Therefore, NGNs are required to keep providing those legacy services while keep moving toward the new business model. Table 5-3 shows an example of dome services that an NG-PON able to provide and support.

Table 5-3 Example of Services provided by NG-PON

Service	
Telephony	VOIP
	POTS
TV (real-time)	IPTV (IP multicast)
	Digital TV broadcasting (RF Video overlay)
Leased line	T1
	E1
High speed Internet access	
Mobile Backhaul	
L2 VPN services	
IP services	i.e. L3 VPN AND VoIP

However, to support this wide range of services with very different needs in terms of a bandwidth and Quality of Service (QoS), for example POTS service needs a guaranteed fixed bandwidth to achieve a low-delay (less than 1.5ms) and a low-jitter. To support all these services including those legacy ones, an NG-PON must support sufficient traffic management mechanisms, more details are provided in [ITU G.987.1 7.6 & G.9807.1 A.7]. [14] [15]

5.5 Bit rate

Table 5-4 shows supported bit rates

Table 5-4[16] [15] XG-PON and XGS-PON supported bit rates

Item	Transmission direction	Nominal rate (Gbps)
XG-PON	Downstream	9.95328
	Upstream	2.48832
XGS-PON	Downstream	9.95328
	Upstream	9.95328
Note: when an XGS-PON co-exists with an XG-PON in a basic wavelength set scenario (see figure 5-13) through a TDMA scheme in the upstream, then the XGS-OLT should also support a bit rate of 2.48832 Gbps in upstream, bear in mind that the XGS-PON ONU will only support a 9.95328 Gbps bit rate.		

5.6 Split ratio

XG-PON and XGS-PON can support different split ratios up to 1:256 (logically and physically), however the more split ratio causes more power loss, therefore a different power budget is required for a different ODN split and reach. In addition, for an XG-PON and XGS-PON to support a coexistence with GPON, the minimum split ratio they could support is 1:64. It’s possible to achieve 1:64 split with coexistence in a single split configuration as showing in figure 5-14. However, to achieve a higher split ratio (e.g. 1:256) then it will require that both systems (XG-PON and XGS-PON) support the split logically (e.g. TDMA) and physically where more power budget is required to achieve the split that might be possible to achieved through installing an active component called a “reach extension”. [14] [15]

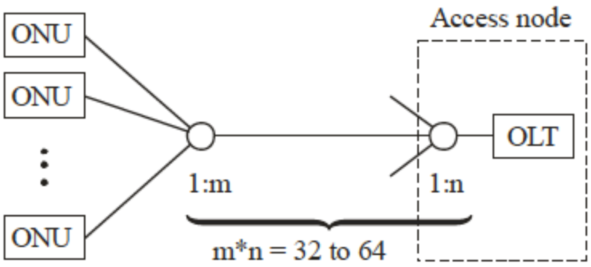


Figure 5-14 [14] [15] Single split

5.7 XG-PON/XGS-PON Transmission convergence layer (NG-PON TC)

The TC layer of XG-PON and XGS-PON (which we refer to as NG-PON), is almost identical with tiny little changes, this report will explain NGPON TC layer and show any differences between the two system's TC layers. NG-PON TC is the heart and soul of the XG-PON and XGS-PON protocol stack, it interfaces between the upper layer protocol data unit (PDU), or as per ITU-T called the Service data unit (SDU), and the physical medium dependent (PMD) layer. Unlike the GPON GTC layer which has two sub layers, the XGTC and XGSTC layer contains three sub layers, two upper sub layers are similar to the GTC sub layers, with the addition of a new sub layer at the bottom. These sub layers are, XGTC/XGSTC service adaptation sub layer, XGTC/XGSTC framing sub layer, and XGTC/XGSTC Physical adaptation sub layer. The fact that NG-PON is a subsequent technology driven from the legacy GPON, many similarities can be observed between their TC layers, the structure, framing, field names and contents inside the frames, bandwidth allocation, and many more. Besides that the XGTC is designed to support more splitting and higher bit rates as well as co-existence requirements, therefore it differs from the GPON GTC layer, for an example some unused fields in the GTC overhead framing have been removed and some fields are increased in size to support more logical identification, as it will be explained in further details later. [17]

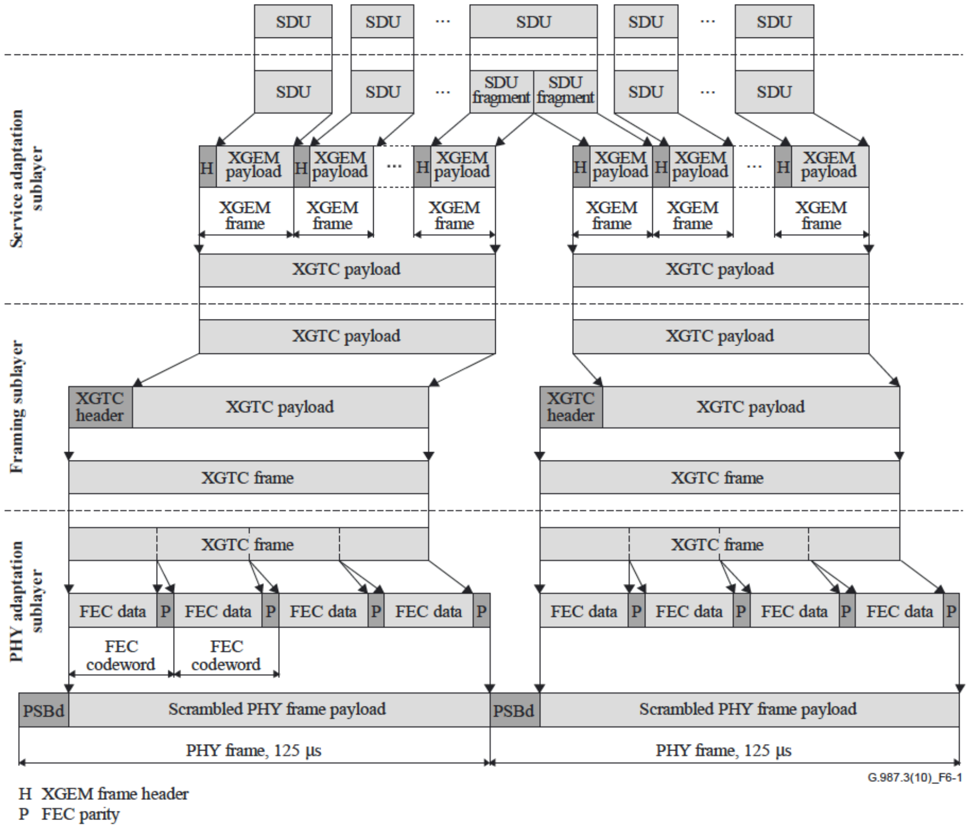
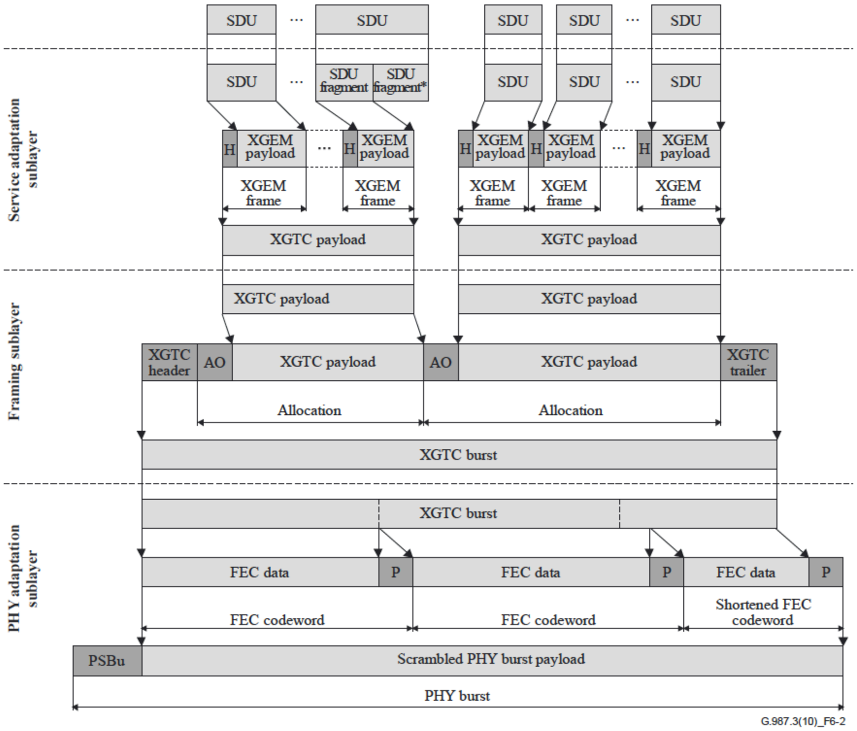


Figure 5-15 [17] Downstream XGTC sub layers and SDU mapping

One of the inherited characteristics from the legacy GPON/BPON, is the physical frame length of 125 µsec which remained unchanged to serve the purpose of coexistence. Figure 5-15 shows the downstream SDU mapping process through the XGTC sub layers. Note that the structure is identical in XGS-PON TC layer, only the XGS-PON framing sub layer frame is called an “FS frame” as per ITU-T G.9807.1 in order to differentiate between both technologies. [17]

Figure 5-16, shows the upstream SDU mapping steps through the XGTC sub layers until the frame is sent on the physical interface as a sequence of accurately planned bursts (TDMA). Note that the structure is identical with XGS-PON TC layer, only the XGS-PON framing sub layer’s frame in the upstream is called an “FS burst frame” as per ITU-T G.9807.1. Also it’s noticeable that the GPON influence is very obvious as all of this structure sounds familiar from what we have discussed in chapter 4. [17]



* The remaining fragment of the SDU is transmitted in the subsequent allocation with the same Alloc-ID.
H XGEM frame header
AO Allocation overhead
P FEC parity

Figure 5-16 [17] Upstream XGTC sub layers and SDU mapping into physical frame

5.7.1 XGTC/XGSTC service adaptation sub layer

On the transmitter side (OLT or ONU) the XGTC/XGSTC service adaptation sub layer carries an SDU (user data or OMCI) by encapsulating it inside an XGEM frame payload, it also performers a traffic multiplexing function by adding a port ID to the

XGEM header to identify the final destination end user. An optional encryption can be applied on the XGEM frames. Upon receiving the XGEM, this sub layer will go through all XGEM frame headers and performs filtering according to their port-ID, in case encryption was applied, it will also decrypt the payload. In addition, it will perform fragmentation/defragmentation of SDU data as required. [17]

5.7.2 XGTC/XGSTC framing sub layer

At the data transmitter, the framing sub layer is responsible to map a group of contiguous XGEM frames into an XGTC/FS payload in a downstream or into an upstream XGTC/FS bursts interval's payload, then it will add an XGTC/FS frame header to form a full XGTC/FS frame. The XGTC/FS header carries the OLT and ONU control information by embedding a PLOAM and an OAM messaging into the header in both direction. These control information is responsible for key protocol functions such as:

- Bandwidth allocation and reporting
- ONU discovery, activation, power management and state management.

At the receiver side, this sub layer will perform de-framing, header parsing and forwarding/processing accordingly. Framing and header information will be discussed in more details. [17]

5.7.3 XGTC/XGSTC PHY adaptation sub layer

This layer is new compared to the legacy GPON, though many of its functions was already performed by the GPON GTC framing sub layer and/or the PMD layer, with some differences. At the transmitter side, the PHY adaptation sub layer will receive the XGTC/FS full frame or burst from the framing sub layer, then it will append a mandatory forward error checking (FEC) in the downstream and the an optional FEC in upstream only if was enabled, (Note that FEC was optional in GPON in both directions, but still optional in NG-PON upstream). Therefore, it appends FEC by splitting the XGTC frame into FEC codewords and adds a calculated parity at the end of each word, then scrambles all the words and appends a physical synchronization block header (PSB). The final frame (scrambled payload and PSB header) is still a 125 µsec which is compatible with legacy GPON. [17]

At the receiver, PHY adaptation sub layer (at OLT or ONU), will synchronize the bit stream and delineate the bytes, then it will descramble the frames, check the FEC words for errors, if no error found, it will remove the FEC parity and forwards the XGTC/FS frame to the framing layer.

One of the difference, unlike GPON, the FEC process is required on the XG-PON/XGS-PON where the OLT and the ONU both must support the FEC process, this is due to the higher requirement that an NG-PON has to provide, such as a splitting ratio, a longer reach, the coexistence, and a better sensitivity and overload tolerances at the physical receiver where higher bit error rate (BER) are expected than in legacy GPON. [17]

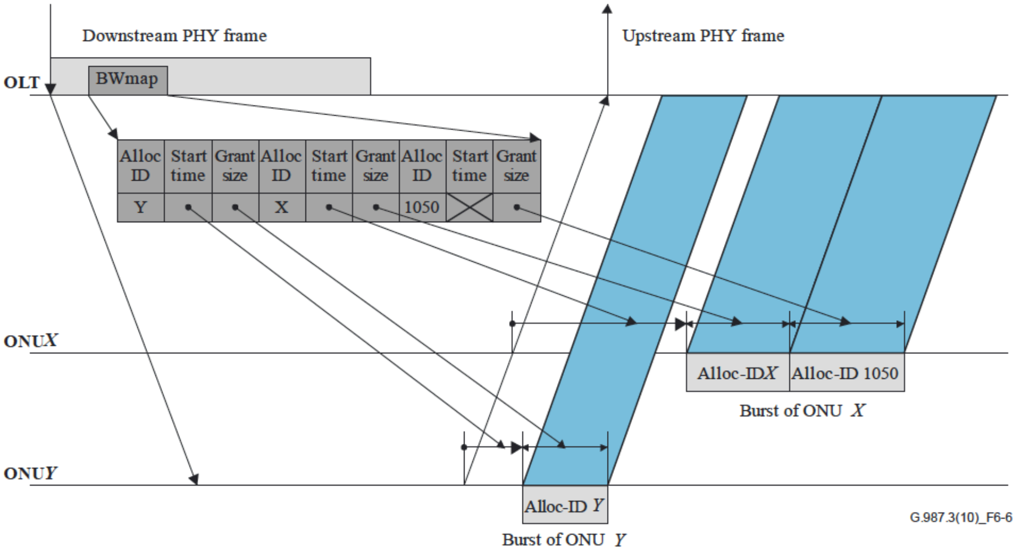
5.7.4 TDM and TDMA in XGTC/XGSTC layer

Similar to GPON, the TDM is achieved in the downstream using, the port-ID inside the header of each XGEM frame, where all user's data (XGEM payload) is sent to all ONU's that read through received XGEM headers and filter the frames according to the PORT-ID value, whether it belongs to an ONU user to forward it to the intended service port or if it doesn't match to discard it.

In the upstream direction, a TDMA is orchestrated by the OLT through the BWmap allocations. After an ONU discovery and activation is completed and the ONU moves to the operating state, at this point the following should be accomplished:

- The OLT already gave each ONU a default ONU-ID.
- The OLT learned how many traffic bearing entities (T-CONT), every ONU owns.
- The OLT already mapped these T-CONTs, to an allocation ID (Alloc-ID).

Now, in the next downstream 125 µsec frame, the OLT will send within the XGTC/FS frame header a bandwidth allocation map (BWmap), see figure 5-17. The BWmap will specify every Alloc-ID (or T-CONT) a dedicated transmission time, in other words, every ONU's T-CONT is granted a part of the upstream 125 µsec physical frame to transmit its data without interfering with another ONU's T-CONT in the upstream. Information such as the start time and grant size will tell each Alloc-ID when to start the transmission and for many bytes. The ONU upstream burst frame contains one or more T-CONT interval allocation, at the OLT, this will look like if the all bursts are combined together to form a physical upstream in a 125 µsec frame.[17]



Alloc ID	Start time	Grant size	Alloc ID	Start time	Grant size	Alloc ID	Start time	Grant size
Y	•	•	X	•	•	1050	☒	•

Figure 5-17 [17] XGTC/XGSTC upstream TDMA

This raises the following question; how does the system decide bandwidth allocation and what are the mechanisms in use to achieve it? luckily that answer was already discussed during 4.3.2. Similar to the DBA in GPON, the NG-PON also dynamically allocates the bandwidth, where the OLT grants every T-CONT a bandwidth based on its traffic status (T-CONT buffer status) and the contracted bandwidth according to SLA. Similarly, a lack of transmission by the ONU (through idle XGEM frames) will cause the OLT to re-allocate its T-CONT's bandwidth to another busy ONU.

5.8 XGTC/XGSTC layer framing

In the upcoming subtitles, we will discuss NG-PON framing configuration and its header format in both directions, the downstream and the upstream. This will help to provide a clear idea on how the system protocol functions.

Similar to the previously discussed NG-PON specifications, the framing of NG-PON structure is also inherited from the GPON, nonetheless because of NG-PON higher performance expectations than the GPON, such as the split ratio and distance reach with higher data rates and more tolerance to BER, therefore the XGTC/XGSTC layer is designed to fulfill these requirements.

Unlike the one byte alignment in GPON frame. NG-PON has a 4-byte alignment words when applicable. Also, the NG-PON is more robust by accommodating a 13-bit HEC within the critical fields which is able to detect and correct more errors than GPON's 8-bit CRC.

Another key difference related to the PLOAM message field, wherein the GPON TC header there used to be only one PLOAM message per each 125 µsec physical frame, where the OLT will keep sending it even if it doesn't have any PLOAM messages to send, where it will transmit an empty PLOAM field, as a result this will reduce the bandwidth utilization efficiency. This have been changed in NG-PON downstream frame, where the frame may carry as many as one broadcast PLOAM message and one additional unicast PLOAM message intended for each ONU, this dramatic change tremendously decreased the PON initializing time, for example when many ONUs are powered up simultaneously. Another key change is that the XG-PON/XGS-PON OLT will not transmit the PLOAM field in the downstream if there are no pending PLOAM messages planned for transmission, which saves a precious bandwidth that could be used to carry a useful data instead of sending empty PLOAM fields.

NG-PON has improved the security when compared with GPON, for an example the NG-PON supports downstream multicast encryption and upstream encryption, it also has a bigger superframe counter in its XGTC/FS header which gives a better security level. In addition, a new field was added into the XGTC/FS physical adaptation sub layer header called "PON-ID", this field is useful if two OLTs (active and standby) were implemented for redundancy, in order for the ONU to identify which OLT is in active state and in control.

Overall, It Looks like that the ITU-T has done a lot of cleaning of unnecessary fields from the GPON GTC where some fields became longer to support more logical capacity, and some new fields that didn't exist in GPON were added to achieve the higher performance expectations and the coexistence of the NG-PON.[17]

5.8.1 XGEM framing

XGEM frames are carried inside the XGTC/FS layer payload in both directions, downstream and upstream, inside each XGEM frame's payload there is an encapsulated SDU which could be either a user data as an Ethernet frame or an MPLS packet, or a fragment of either one of them. It also encapsulates/de-capsulates the OMCI PDU. Figure 5-18 shows the XGEM frame structure in the downstream and the upstream direction. [17]

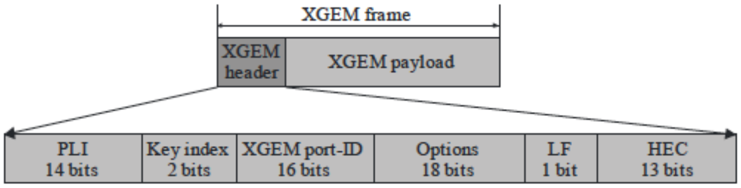


Figure 5-18 [17] XGEM frame structure

It contains the following fields:

- Payload length indication (PLI), this field's value indicates the size of the XGEM payload in bytes, because it's a 14-bit, therefore the maximum payload it could indicate is 16,383 bytes. This means a larger Ethernet frames (e.g. jumbo 9000-byte frames) will be fragmented in order to be mapped into XGEM frames.
- Key index, is a 2-bit field used to indicate if the XGEM frame is encrypted (its value = 01 or 10) or not (value = 00), if it was encrypted then the key index will indicate which one of the two encryption keys is used for each encryption key type, either a unicast or broadcast, the encryption key type is implicitly identified by PORT-ID.
- XGEM PORT-ID, is a 16-bit field, which could indicate a maximum of 65535 unique port-ID, each traffic bearing entity (T-CONT or Alloc-ID) will get one or more unique port-ID assigned to each individual logical connection by the OLT. Also, a unique PORT-ID is designated for the OMCC channel communication per each ONU, this PORT-ID will equal in value the ONU-ID. The process of PORT-ID

assignment is done between the OLT and the ONU through OMCI communication during the ONU activation process. Table 5-5 shows XGEM PORT-ID values.

Table 5-5 [17] XGEM PORT-ID values

XGEM PORT-ID	Designation	Comment
0 to 1022	Default	Used for OMCC channel, and it equals to ONU-ID
1023 to 65534	Assignable	Can be assigned for each individual logical connection within each ONU.
65535	Idle	Used for Idle XGEM.

- Options field, is n 18-bit saved for future use, it carries all zeroes for now.
- Last fragment (LE), is a one bit, if it was set, then it indicates that the SDU fragment is the last fragment, if it was zero then the XGTC service adaption sub-layer will buffer the fragment until it receives the last fragment, then it sends the assembled SDU to its final destination (service port).
- Hybrid error correction (HEC), is a 13-bit used for error correction and detection of the header. Since the header is very important in the process of XGEM frame delineation, where the system will be able to locate the next XGEM frame by using the length of the current XGEM frame through PLI value, however if the HEC process finds the header information to be corrupted beyond correction, then the system will discard this XGEM frame and all the subsequent frames within the same XGTC payload.
- XGEM payload, is variable in size, its size is in bytes, indicated by the value of PLI field in the header. The minimum payload is 8 bytes, this could by 1 to 8 bytes of actual data, if it was less than 8 bytes, then it will be filled by padding bytes (value 0x55) until it reaches the minimum 8 bytes. [17]

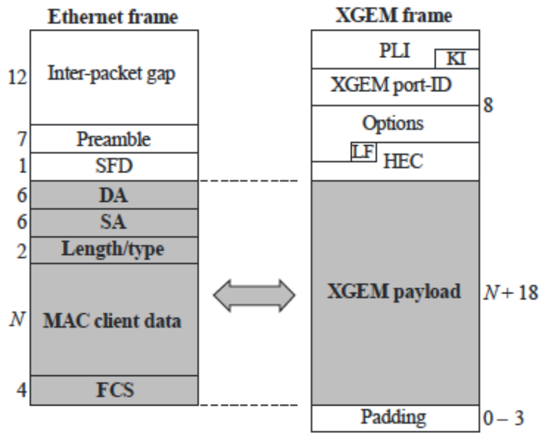


Figure 5-19 [17] Ethernet mapping into XGEM frame

5.8.2 Mapping of SDU into XGEM frames

Similar to GPON (ITU G.984) GEM mapping, Ethernet is mapped directly inside the XGEM payload as shown in figure 5-19, similarly the preamble and SFD will be discarded. According to the Ethernet frame size or the remaining size in the XGTC payload, the Ethernet frame could be encapsulated in one or more XGEM frames, however every XGEM frame will carry only one Ethernet frame. Figure 5-20 shows the mapping of multi-protocol label switching packets (MPLS) directly into an XGEM frame, the same rules apply for the Ethernet, will apply for the MPLS encapsulation. [17]

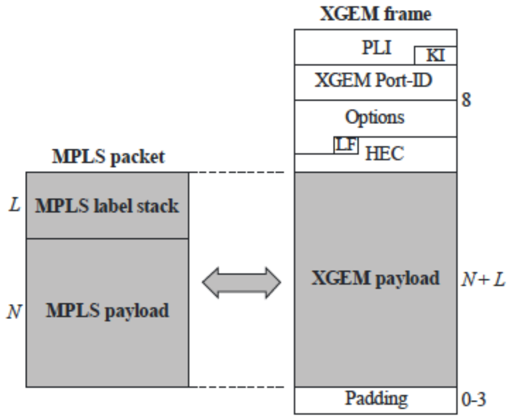
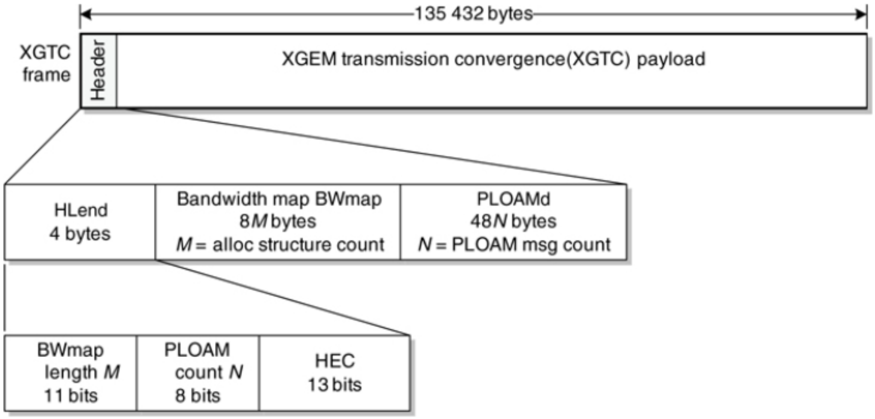


Figure 5-20 [17] MPLS mapping into XGEM frame

5.8.3 XGTC/XGSTC downstream framing

The downstream XGTC/FS frame fixed size equals 135,432 bytes, as shown in figure 5-18, it includes the header followed by a payload of contiguous XGEM frames. The header consists of three fields, HLend which is the only fixed size field, and the other two fields are changing according to the situation, they are the Bandwidth map (BWmap) and the PLOAMd. [17] [10]



135 432 bytes
XGTC frame
Header
XGEM transmission convergence(XGTC) payload
HLend 4 bytes
Bandwidth map BWmap 8M bytes M = alloc structure count
PLOAMd 48N bytes N = PLOAM msg count
BWmap length M 11 bits
PLOAM count N 8 bits
HEC 13 bits

Figure 5-21 [10] XGTC downstream frame

5.8.3.1 Header Length in downstream (HLend)

It's a 4-byte field that specifies the length of the other non-fixed size fields in the header, it contains three subfields. BWmap length is 11-bit in size which used to specify the number of bandwidth allocations within the BWmap field, it can specify up to 2048 allocation. The PLOAM count, is an 8-bit in size, it specifies the number of PLAM messages inside the PLAOM field, it can specify up to 256 PLOAM message. [17]

5.8.3.1.1 Bandwidth map (BWmap)

The BWmap structure is shown in Figure 5-22. BWmap field grants the upstream transmission for every Alloc-ID (T-CONT), it can address a zero to many allocations, every allocation's structure within a BWmap is a fixed 8-byte field. The BWmap field size is specified by the BWmap length field in the HLend, which equals, M*8 bytes. Every allocation structure is intended for only one allocation interval (Alloc-ID). Nevertheless, there could be one or more Alloc-ID that belongs to the same ONU which in this case called (burst allocation series). Each field's function is explained as per the following:

- Allocation ID (Alloc-ID): is a 14-bit field used to address either a (T-CONT) that will consume the bandwidth allocation in the next upstream transmission, or an upstream OMCC channel within an ONU. Similar to GPON, the first allocation ID equals the default Alloc-ID, which is the same value as the assigned ONU-ID. The default Alloc-ID is used to assign bandwidth to the T-CONT that carries either user data or OMCC PDU, or to the PLOAM only allocations intended to a particular ONU. A total of 16384 Alloc-ID can be designated as per table 5-6. I believe we both agree that the Alloc-ID is the essential entity that facilitates TDMA function performed by the OLT to orchestrate all upstream traffic.

- Flags: a small 2-bits field. Any bits that were not useful in the GPON allocation structure flag field have been removed, therefore only two bits were kept inside the NG-PON flag field. The DBRu and the PLOAMu.
  - DBRu, is a one bit that will be set if the OLT wants a buffered (queued) traffic status from the ONU, if it was set, the ONU will send a DBRu report in the T-CONT associated with the Alloc-ID, which is used by the OLT to manage PON dynamic bandwidth assignment (DBA).

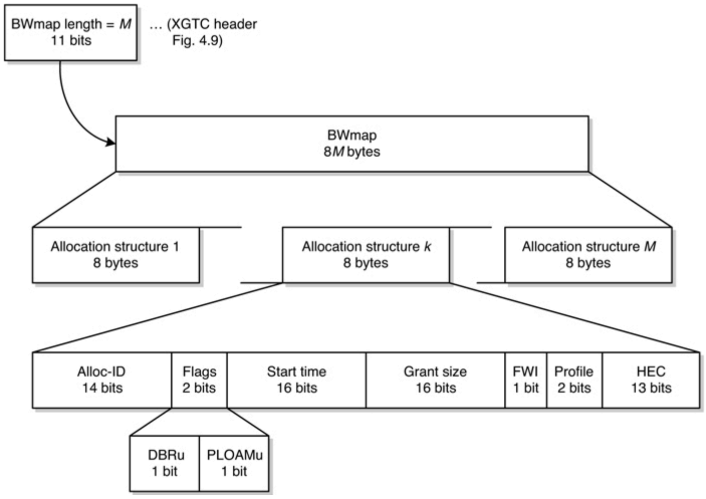
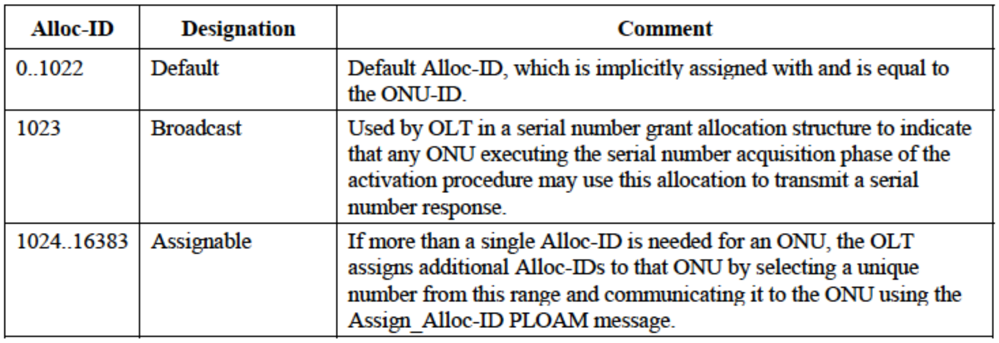


Figure 5-22 [10] XGTC BWmap structure

Table 5-6 Alloc-ID designation



Alloc-ID	Designation	Comment
0..1022	Default	Default Alloc-ID, which is implicitly assigned with and is equal to the ONU-ID.
1023	Broadcast	Used by OLT in a serial number grant allocation structure to indicate that any ONU executing the serial number acquisition phase of the activation procedure may use this allocation to transmit a serial number response.
1024..16383	Assignable	If more than a single Alloc-ID is needed for an ONU, the OLT assigns additional Alloc-IDs to that ONU by selecting a unique number from this range and communicating it to the ONU using the Assign_Alloc-ID PLOAM message.

  - PLOAMu, is a one bit that is set when the OLT want the ONU to send a PLOAMu message, and is only sent using a default Alloc-ID, in other words, it will always show up in the first Allocation interval of a contiguous allocation (burst allocation).

- Start time: is a 16-bit field, that indicates the first byte of an upstream XGTC/FS burst which appears right after the PSB frame, the granularity of the start time field is one word, where one word equals 4 bytes for a 2.48832 Gbps upstream bit rate, and for a 9.95328 Gbps it equals 16 bytes for each data block. In case of a burst allocation, the start time field will only show in the first allocation, the other allocation will take the value of 0 x FFFF to indicate continuity of the intervals.

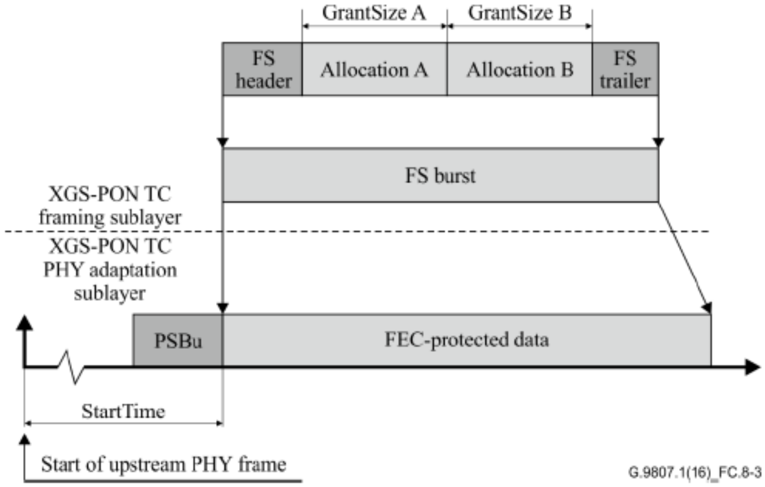


Figure 5-23 Interpretation of Start-time and Grant-Size

- Grant size, is a 16-bit field that indicates the size of the XGTC payload, that includes the allocations plus the DBRu field, in other words XGTC/FS header, trailer and FEC words are not included, in case only a PLOAMu burst is sent then the Grant size value is zero, also if a DBRu only sent then the Grant size value is equal one word. Furthermore, the minimum user data payload to be carried is 4 words (16 bytes). Similar to the start time, the grant size granularity depends on the upstream bit rate, wherein 2.48832 Gbps speed it equals 4 bytes block, and in 9.95328 Gbps it equals 16 bytes block. Figure 5-23 explains the Start Time and the Grant Size of an XGS-PON.
- Forced Wakeup Indicator (FWI), is a one bit, it has a meaning only within first allocation of a burst allocation series, when it set, and the ONU in a low power mode (sleep mode) will wake up.
- Profile Index or Burst profile field, is a 2-bit field used to tell the ONU which header profile (predefined by OLT) to use in the upstream burst.

- Hybrid error correction (HEC), is a 13-bit used for error detection and correction that covers each allocation structure. [17] [10][15]

5.8.3.1.2 PLOAMd field

PLOAMd field structure is shown in figure 5-24, the PLOAMd size is determined by a PLOAM count field in the HLend and it equals (P * 48 bytes), where the size for each PLOAM message is 48 bytes, PLOAMd value could be zero which means it won't be sent, alternatively it might have one or more messages to send. [17]

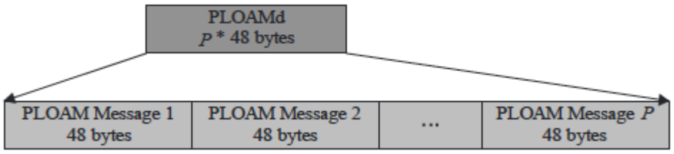
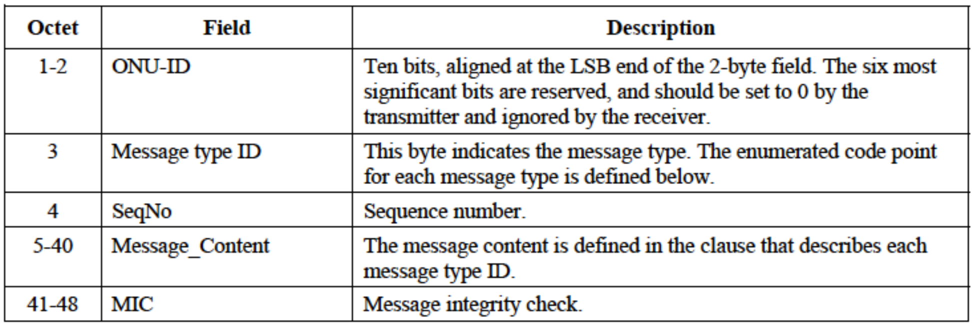


Figure 5-24 [17] PLOAMd field structure

Table 5-7, shows PLOAMd message structure and function of each field.

Table 5-7[17] PLOAM message structure



Octet	Field	Description
1-2	ONU-ID	Ten bits, aligned at the LSB end of the 2-byte field. The six most significant bits are reserved, and should be set to 0 by the transmitter and ignored by the receiver.
3	Message type ID	This byte indicates the message type. The enumerated code point for each message type is defined below.
4	SeqNo	Sequence number.
5-40	Message_Content	The message content is defined in the clause that describes each message type ID.
41-48	MIC	Message integrity check.

- ONU-ID, is used to identify which ONU will receive the message, where each ONU-ID is unique within a PON and remains unchanged until the unit is reset.
- Message Type ID field, is used to specify the message type, there are 12 message types in the downstream and 5 types in the upstream, as per ITU-T G.987.3.
- Sequence number (SeqNo) field, is an 8-bit counter that counts from 0 to 255. The OLT will keep a separate counter for each unicast PLOAMd it sends to each ONU, and a separate counter for a broadcast PLOAMd (where ONU-ID value is 1023), when the OLT sends a PLOAMd message, it will include the message count into the SeqNo field, when an ONU responds to a PLOAM message request, it will send the PLOAMu message with the same SeqNo in order for that OLT to identify this response as an answer to its request.

- Message content field, carries the payload of the message which is the message semantic and it varies in size according to the message type.
- Message integrity check (MIC), is used to identify the protection of the sender, where the sender will calculate MIC using a shared PLOAM integrity key (PLOAM-IK) for each message, then it appends the calculated MIC into a PLOAM message and sends it, when the message arrives at the receiver, the receiver will calculate its own MIC using the same PLOAM-IK and compares the calculated MIC with received one, if they don't match, then the message will get discarded. This process is to achieve PLOAM messages integrity, for example to protect the system against any possible PLOAM injection attack. [17] [10]

5.8.3.2 FS frame trailer

XGS-PON FS frame contains a 4-byte FS trailer field, it's a 32-bit interleaved parity (BIP), however this field is not really useful since the ONU will monitor BER by counting all performed FEC corrections, and also because the FEC on the downstream in NG-PON is always enabled (by design it's mandatory). In this case, it acts as an extra BER monitoring.

5.8.4 XGTC/FS upstream framing

XGTC/FS upstream burst consists of a burst header and one or more allocation interval linked with a designated Alloc-ID (T-CONT) and an upstream payload that is carried within each XGTC/FS payload inside each interval. Unlike the GPON upstream burst, the XGTC/FS upstream burst frame includes a trailer. Figure 5-25 shows an XGTC burst structure, note that the FS Burst structure will be identical. [17]

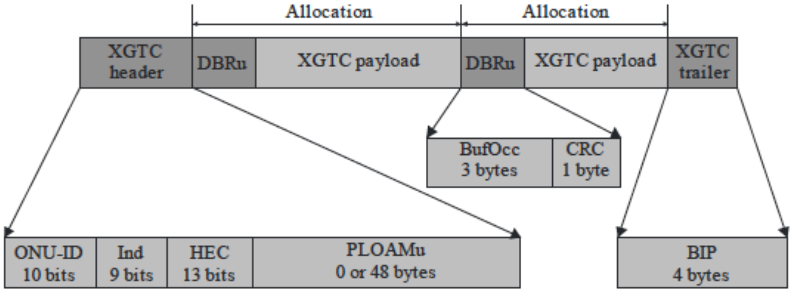


Figure 5-25 [17] XGTC burst structure

Likewise, an XGTC/FS header in the upstream burst will have two sections, a fixed and a non-fixed size. The fixed size section is a 4-byte in size, it contains an ONU-ID, Ind and an HEC, where the non-fixed size contains a PLOAMu field. Header fields are explained below:

- ONU-ID, is a 10-bit field contains a unique ONU-ID assigned to the ONU during the activation process by the OLT, where the ONU-Id is used for management communication between the OLT and the ONU.

In XG-PON, the ONU-ID values that range between 0 to 1022 are assignable to ONUs at activation, and ONU-ID value 1023 is reserved for a broadcast use in order to send a PLOAMd messages to unknown ONUs, and it is also used by the ONU to send a PLOAMu to the OLT.

IN XGS-PON, the same applies except that the 1021 ID is reserved for future use, 1022 is only used to broadcast PLOAMd by OLT (but not PLOAMu), and the rest of the ONU-ID values follow the same designation as in the XG-PON.

- Indication (Ind) field, has a 9-bit where the 8th most significant bit (MSB) called "PLOAM queue status", when it's set it indicates that the PLOAM messages queue is still not empty after sending the current message. The least significant bit (LSB) or bit zero is called "Dying gasp" or (DG), if it set then it indicates that a local issue at the ONU might prevent it from sending data.
- HEC field, is a 13-bit used for an error detection and correction, if the header is corrupted then the whole burst will be discarded.
- PLOAMu field, it's also a 48-byte in size, however, unlike PLOAMd that might contain more than one message. The PLOAMu contains only one message which only exists if it was requested by OLT (i.e. a PLOAMu flag is set).

Inside each allocation there is an optional DBRu header and an XGTC/FS payload. The DBRu only exists if it was requested by the OLT through setting a DBRu flag within that allocation. The DBRu structure is shown in figure 5-25, it's a 4-byte in size that contains two fields as follows:

- Buffer occupancy (BufOcc), is 3 bytes in size, its value represents the number of words (4 bytes each) of waiting SDU packets to be sent upstream, these SDU belongs to services (logical ports) that have been mapped into this specific T-CONT.
- CRC-8 field, is used for DBRu field error detection and correction.

XGTC/FS trailer, is 4 bytes in size, and used for bit interleaved parity (BIP) for the whole burst exclude the trailer, the BIP is used by the OLT to measure bit error rate (BER) on the upstream fiber, however this BER is only used when the optional upstream FEC is disabled. [17]

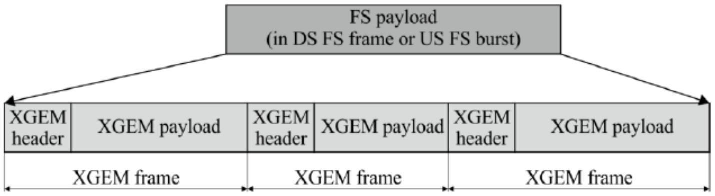


Figure 5-26 [15] XGSTC payload structure

5.8.5 XGTC/FS payload

XGTC/FS payload has the same structure in the downstream and upstream, however it only varies in size, and it contains a contiguous XGEM frames. The downstream XGTC payload size is 135432 bytes minus the XGTC header. The XGTC payload in the upstream burst, is equal to each allocation interval size minus the allocation header if it exists. Figure 5-26 shows FS payload structure. [17]

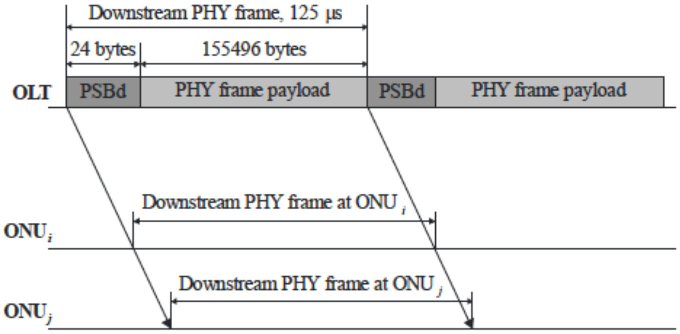

Downstream PHY frame, 125 µs
24 bytes
155496 bytes
OLT
PSBd
PHY frame payload
PSBd
PHY frame payload
Downstream PHY frame at ONU i
ONUi
Downstream PHY frame at ONU j
ONUj

Figure 5-27 [17] NG-PON downstream physical frame

5.8.6 Downstream physical (PHY) frame

As shown in Figure 5-27, the downstream physical frame is a non-stop train of frames, each has a fixed length of 125 µsec, since the bit rate is 9.95328 Gbps, thereby each physical frame could carry 155,520 bytes, divided to 24-byte Physical synchronization block (PSBd) and physical frame payload that will carry an XGTC/FS frame after the FEC party addition and the scrambling as shown in figure 5-27. [17]

PSBd is fixed in size (24-byte), as shown in Figure 5-28, it contains the following fields:

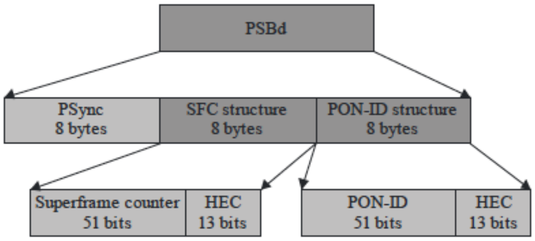


Figure 5-28 [17] XG-PON PSBd structure

- Physical synchronization sequence (PSync), is a 64-bit pattern as shown in figure 5-28, the ONU receiver will match against it and use it to find the frame boundaries.

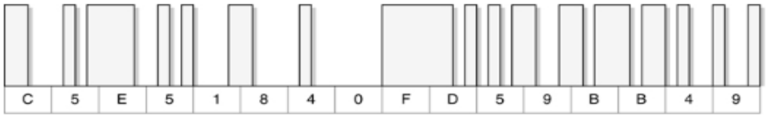


Figure 5-29 [10] PSync pattern (0xC5E5 1840 FD59 BB49)

- Superframe counter (SFC) structure, as shown in figure 5-27, it contains a 51-bit superframe counter that is protected by a 13-bit HEC, it is a counter for the transmitted physical frames, and it will keep incrementing until it equals all ones then it will restart from zero.

Until this point we have seen that the header is identical in both XG-PON and XGS-PON, however the last 8-byte field is slightly different, and where it's called a PON-ID structure in the XG-PON, and the same field is called an Operation control structure OC in the XGS-PON. The following bullets will explain each field in details.

- PON-ID structure in XG-PON, is 8 bytes divided into 51 bits PON-ID plus a 13-bit HEC. The PON-ID is set by the OLT (all zeroes as a default value). Nevertheless, Annex E in G.987.3 presents another optional implementation of the PON-ID structure in a "complaint" XG-PON OLT and ONUs, which makes the later discoverable by the OMCI. Furthermore, implementing this optional requirement will enable the site engineers to get extra information such as PON-ID and OLT/RE-launch power. According to Annex E, the PON-ID structure will have a different interpretation as shown in Figure 5-30, and it contains the following subfields:

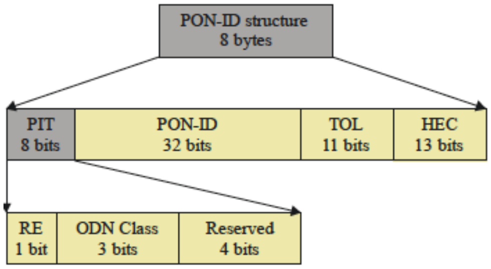


Figure 5-30 PON-ID structure according to Annex E G.987.3

  - PON-ID Type (PIT), is an 8 bit that is statically provisioned by the network operator to identify the optical distribution network (ODN) architecture and its optical path loss (OPL) class. It contains the following subfields:

- Reach extender (RE) flag, is a one bit, it's used to identify the content type of a transmit optical level (TOL) field, where if the RE equals zero, it means that the TOL carries the lunch power of the OLT, and if the RE equals one, means the TOL carries the launch power of the reach extender instead.
- ODN class, is 3 bits, it's used to identify the XG-PON ODN class. See Table 5-8.

Table 5-8 XG-PON ODN classes

Code value	ODN budget class
000	N1
001	N2a
010	N2b
011	E1
100	E2a
101	E2b
110	Reserved
111	Reserved

- Reserved 4 bits for future.
- PON-ID, is a 32 bit statically provisioned by the network operator to logically label the OLT.
- Transmit Optical Level (TOL), is an 11 bit dynamically specified by the system, it indicate the transceiver launch power of either the OLT or the reach extender based on the RE flag value. In XG-PON, the TOL value should support launch power ranges compatible with XG-PON ODN classes.
- HEC, 13-bits error correction and detection covers the PON-ID structure.

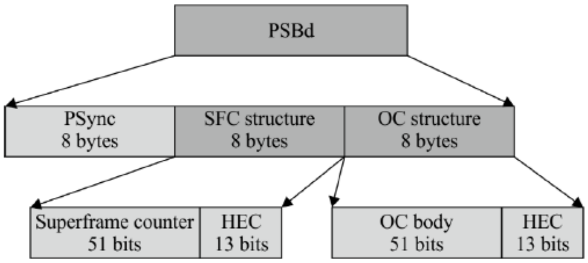


Figure 5-31 XGS-PON PSBd structure

- Operation control structure (OC) in XGS-PON, as shown in Figure 5-31, is divided into a 51 bit OC body protected by 13-bit HEC. The OC body is illustrated in Figure 5-32, it contains the following fields:

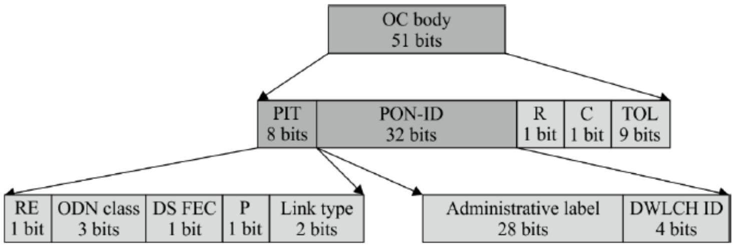


Figure 5-32 XGS-PON OC body structure

  - PON-ID Type (PIT), is an 8 bit that is statically provisioned by the network operator to identify an optical distribution network (ODN) architecture and its optical path loss (OPL) class. As shown in Figure 5-32, PIT contains the following subfields:
    - Reach extender (RE) flag, is a one bit, it's used to identify the content type of transmit optical level (TOL) field, if RE equals zero, it means that the TOL carries the lunch power of the OLT, if RE equals one, it means that the TOL carries the launch power of the reach extender instead.
    - ODN class, is a 3-bit, used to identify an OPL class.
    - Downstream FEC (DS FEC) flag, it's a one bit, if it's set it means that the FEC is enabled in the downstream, if it's zero it means that no FEC in the downstream. However, it's always set in XGS-PON.
    - Protocol (P) flag, is a one bit, if it is set then it means that the XGS-PON (G.9807.1) and NG-PON2 protocol (G.989.3) is used in the TC layer. If it's zero then it means that the XG-PON (G.987.3) protocol is in use.
    - Link Type, is a 2-bits subfield reserved for future use and always set to 00 in XGS-PON.
  - PON-ID field, is the same as the PIN field, it's statically provisioned by the network operator to logically identify the XGS-PON OLT. As shown in

Figure 5-32, PON_ID contains the following subfields:

- Administrative label, a 28-bit field, which is used as a logical label presented by the Element management system (EMS) or the Operation support system (OSS).
- Downstream wavelength channel identifier (DWLCH-ID), it is 4 bits in size that is used by the network operator to statically provision a label for the downlink channel, however this is not used in XGS-PON, it will be used in future NG-PON2. Furthermore, in case of co-existence with XG-PON ONUs, the DWLCH-ID is then combined with the administrative label to form a 32-bit PON-ID field identical to the PON-ID in XG-PON standard (G.987.3).
- R and C, are always set to zero in XGS-PON, they are part of NG-PON2 protocol, and have no use in XGS-PON.
- Transmit Optical Level (TOL), a 9 bit dynamically specified by the system, it indicates the transceiver launch power of the OLT or the reach extender based on the RE flag value mentioned above. To serve the co-existence compatibility with the XG-PON ONU, R & C fields are combined with the 9 bits TOL to form 11 bits TOL field identical to the TOL in G.987.3 standard.

Until now, we knew that scrambling is done only for the XGTC/FS frame before it's transmitted within the PHY frame payload, however that's not entirely true, because the PSBd header is also scrambled individually by using a special pattern value of (0x0F0F0F0F0F0F0F0F). The PHY frame payload consists of the scrambled XGTC/FS 155496 bytes frame, which is, before the FEC party addition, it has a fixed size of 135432 bytes, as shown in Figure 5-15. [17]

5.8.7 Upstream physical (PHY) frame

Upstream physical frame is also a 125µsec long, it carries one or more PHY burst, depending on the operating bit rate, each frame could carry 38880 bytes with a bit rate of 2.48832 Gbps (XG-PON), and carries 155496 bytes with a bit rate of 9.95328 Gbps (XGS-PON). The ONU transmits the upstream short Physical bursts then it stays quite in-between transmissions. These bursts are TDMA orchestrated by the

OLT using the BWmap as discussed earlier, however each ONU have to offset its upstream burst, depends on how far the ONU is set from the OLT, in order that all bursts arrive at the OLT during the right designated allocation so they don't collide, this offset time is calculated by the OLT during the ONU activation ranging process. The physical burst contains a physical synchronization block (PSBu) and a payload that contains an XGTC/FS burst. As shown in Figure 5-33, PYH burst PSBu header could be transmitted within the boundary of the previous PHY frame, also the PHY burst payload could extend into the boundaries of the next PHY frame. [17]

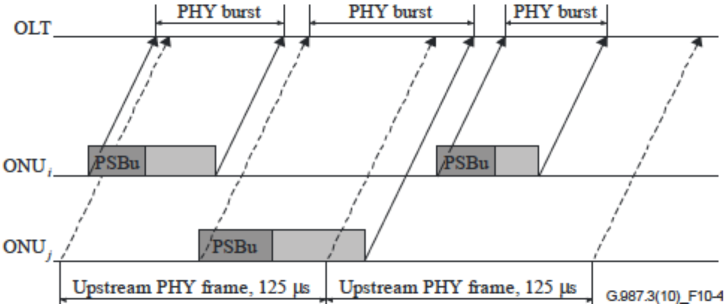


Figure 5-33 [17] Upstream PHY frame and bursts

Similar to the downstream physical frame header, the upstream physical header is called "Physical synchronization block" or (PSBu), as shown in Figure 5-34, it contains two fields, a preamble and a delimiter, they are utilized to help the OLT to achieve byte synchronization and delineation. The size and pattern of the PSBu is chosen by the OLT and communicated to the ONU through a profile field within the BWmap allocation, however these burst profiles are predefined by the OLT and communicated to the ONU through a PLOAM profile message. The recommended preamble size is 160 bits in an XG-PON protocol (G.987.2). Where XGS-PON G.9807.1 recommends a preamble size of 1028 bit for a 10 Gbps. [17]

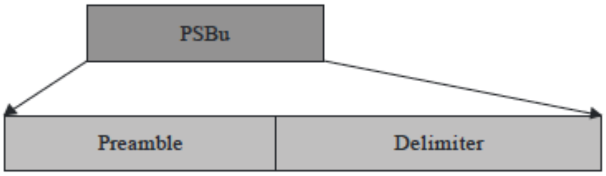
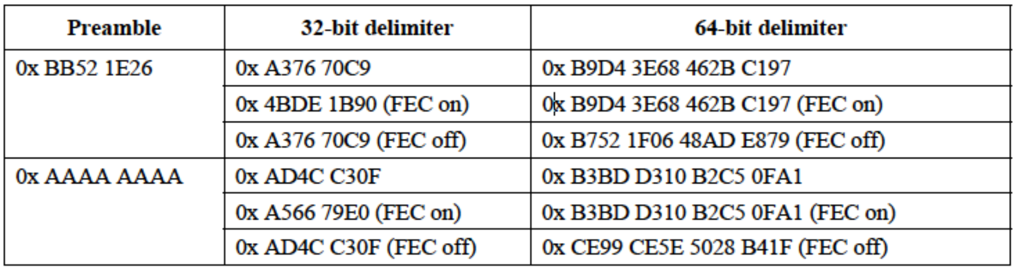


Figure 5-34 [17] PSBu structure

Apart from the burst delineation, PSBu profiles is also used to recover the clock signal. The delimiter recommended size is a 32 bit, though if a high bit error rate (BER) occur or expected, then a 64-bit delimiter is preferred. The OLT could tell from the profile pattern's value if the FEC is enabled in the upstream or not. Table 5-9 shows the different patterns. [17]

Table 5-9 [17] Preamble and delimiter values for different burst profile



Preamble	32-bit delimiter	64-bit delimiter
0x BB52 1E26	0x A376 70C9	0x B9D4 3E68 462B C197
	0x 4BDE 1B90 (FEC on)	0x B9D4 3E68 462B C197 (FEC on)
	0x A376 70C9 (FEC off)	0x B752 1F06 48AD E879 (FEC off)
0x AAAA AAAA	0x AD4C C30F	0x B3BD D310 B2C5 0FA1
	0x A566 79E0 (FEC on)	0x B3BD D310 B2C5 0FA1 (FEC on)
	0x AD4C C30F (FEC off)	0x CE99 CE5E 5028 B41F (FEC off)

It's also important to mention that there should be a guard time between adjacent physical bursts that belongs to different ONUs, in order to prevent collision in the upstream, in the XG-PON the minimum guard time equals a 64-bit, wherein the XGS-PON the recommended minimum guard time equals 512 bits. The guard time is considered by the OLT within each BWmap allocation. The mechanism of implementing the guard time within an upstream transmission is achieved by the ONU by means of transmission (Tx) enable and disable technique, in other words all ONUs should be quiet during a guard time. [17]

5.8.8 FEC process in XGTC/FS PHY adaptation sublayer

Forward error checking (FEC) in the NG-PON standards (G.987.3 & G.9807.1), uses Reed-Solomon (RS) codeword technique. However, both XG-PON and XGS-PON use a shortened version of the RS code for more convenience (of the 125 µsec). Also in the XG-PON system it is a requirement that the FEC supported by the OLT and the ONU, having said so, FEC is always enabled in the downstream, where it is optionally enabled in upstream, governed by the OLT. The reason we need an FEC in NG-PON, it's because of the extra system requirement compared to the legacy GPON, where XG-PON and XGS-PON expected to support a higher bit rate, more split ratio, more low signal power sensitivity or overload at the receiver, longer reach and more tolerance towards a high BER, thereby utilizing an FEC will help achieve these requirements in the logical layer, since the physical layer boundaries already stretched to their limits. [17]

5.8.8.1 FEC in downstream

Downstream uses the FEC code (248, 216), which means 248 bytes the size of the FEC codeword split into 216 bytes of data and 32 bytes of an FEC parity, since XGTC/FS frame size is 135,432 bytes, divided by 248 bytes, this will give us a total of 627 FEC code word. Figure 5-31 shows the process of insertion of the FEC code during a transmission, and figure 5-32 shows the removal process of the FEC codeword and a reconstruction of the original XGTC/FS frame. Note that the PSBd header is not involved in the FEC process.

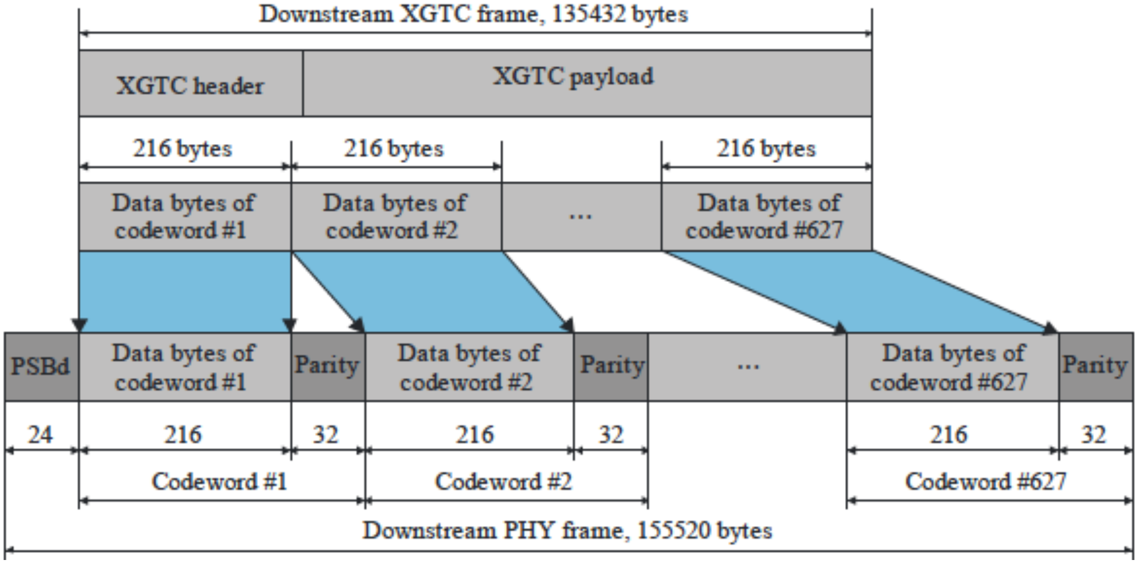

Downstream XGTC frame, 135432 bytes
XGTC header
XGTC payload
216 bytes
216 bytes
216 bytes
Data bytes of codeword #1
Data bytes of codeword #2
...
Data bytes of codeword #627
PSBd
Data bytes of codeword #1
Parity
Data bytes of codeword #2
Parity
...
Data bytes of codeword #627
Parity
24
216
32
216
32
216
32
Codeword #1
Codeword #2
Codeword #627
Downstream PHY frame, 155520 bytes

Figure 5-35 [17] Insertion of FEC before transmission in downstream

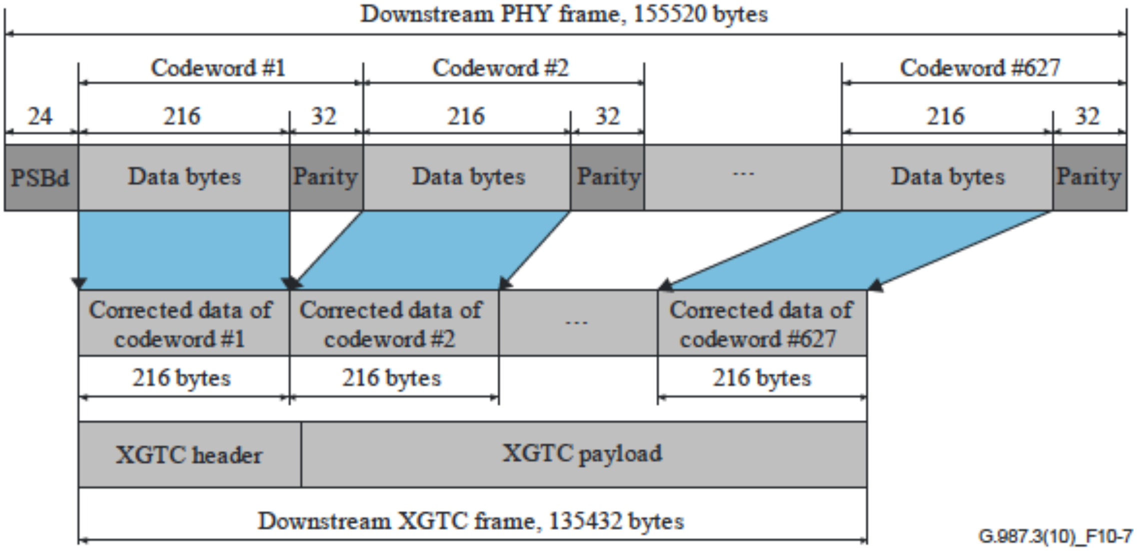


Figure 5-36 [17] removal of FEC at ONU and reconstruction of XGTC frame

5.8.8.2 FEC in upstream

5.8.8.2.1 XG-PON FEC in upstream

The FEC code word in the upstream is (248, 232), which means that 248 bytes is the size of the FEC codeword split into 232 bytes of data and 16 bytes of a FEC parity. The PSBu header will not be included in the FEC parity process. However, because of the non-fixed size of the XGTC burst frame, if the last data byte was less than 232 bytes, in that case to perform an FEC process, the system will pad zeroes at the beginning of the short data byte until it’s 232 bytes long, then it will calculate the FEC, after that it will remove the extra zeroes, then attaches the FEC parity to form a shortened FEC code word. Upon receiving of the upstream physical frame, where the FEC is enabled, the OLT will perform an FEC correction, removes the FEC parity, for the last shortened FEC code word, it will pad zeroes at the beginning of the code word until it’s 248 bytes, then decodes the FEC parity, and removes the extra zeroes before reconstructing the XGTC burst frame. The FEC encoding process at the ONU and the decoding at the OLT is shown in figure 5-33 and figure 5-34. [17]

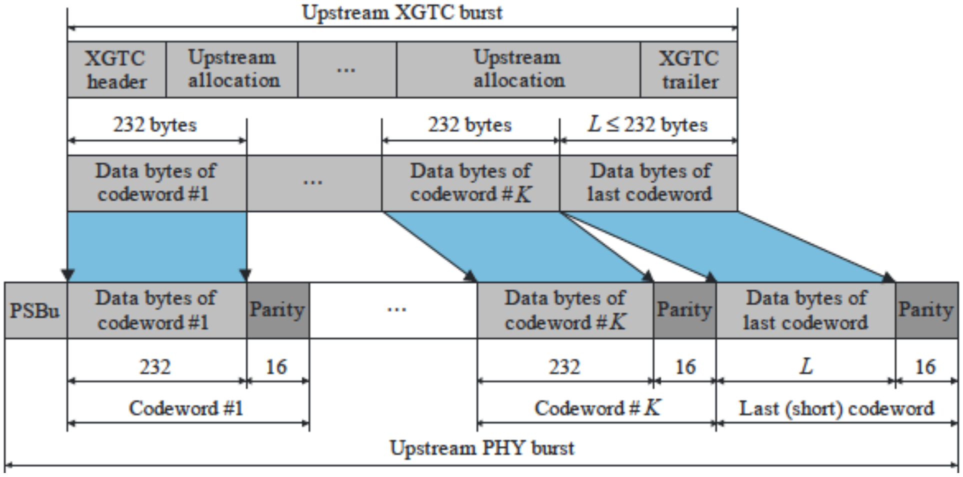


Figure 5-37 [17] 2.5 Gbps FEC encoding at ONU

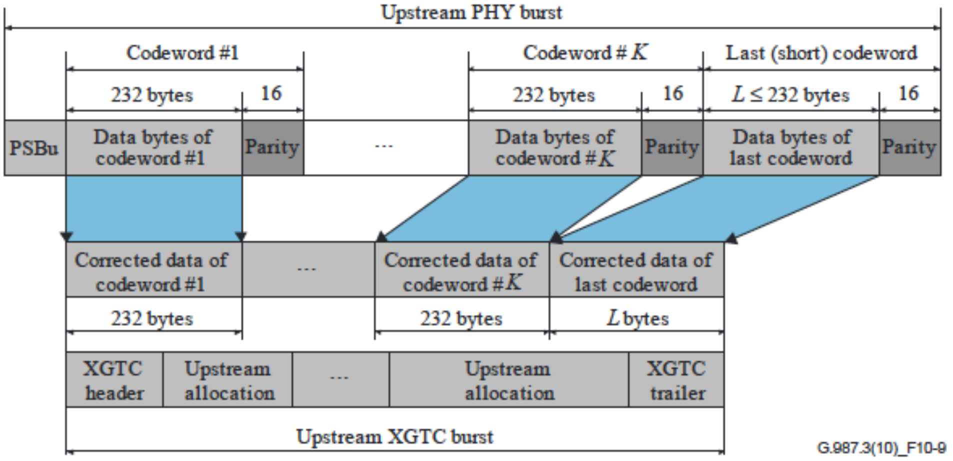


Figure 5-38 [17] 2.5 Gbps FEC decoding at OLT

5.8.8.2.2 XGS-PON FEC in upstream

Similar to a 10 Gbps downstream, the FEC codeword in the upstream is (248, 216), which means that 248 bytes is the size of the FEC codeword, splits into 216 bytes of data and 32 bytes of an FEC parity. The PSBu header will not be included in the FEC parity process. However, because of the non-fixed size of the FS burst frame, if the last data byte was less than 216 bytes, in that case to perform the FEC process, the system will pad zeroes at the beginning of the short data byte until it’s 216 bytes long, then it will calculate the FEC, removes the extra zeroes and attach the calculated FEC parity to form a shortened FEC code word. Upon receiving of the upstream physical frame where the FEC is enabled, the OLT will perform an FEC correction, removes the FEC parity, for the last shortened FEC code word, it will pad zeroes at the beginning of the code word until it’s 248 bytes, decodes the FEC parity, then removes the extra zeroes before reconstructing the FS burst frame. The FEC encoding process at the ONU and the decoding at the OLT is shown in Figure 5-39 and Figure 5-40. [15]

FEC in the upstream can be toggled on/off for each ONU by the OLT, however if the FEC was disabled, then the OLT will consider using an FS trailer to measure the BER, until the FEC is enabled again in the upstream. [15]

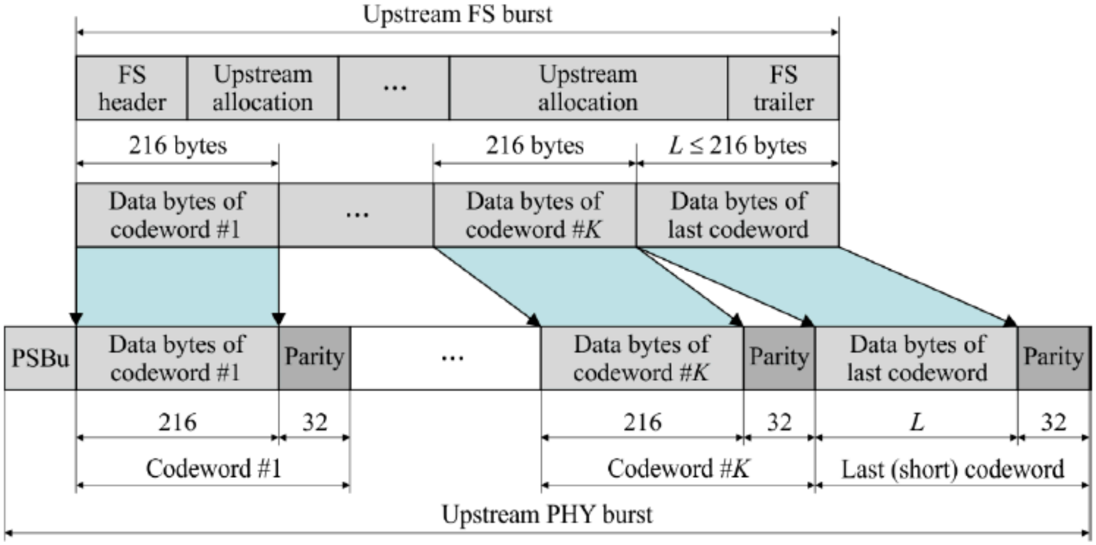


Figure 5-39 [15] 10 Gbps FEC encoding at ONU

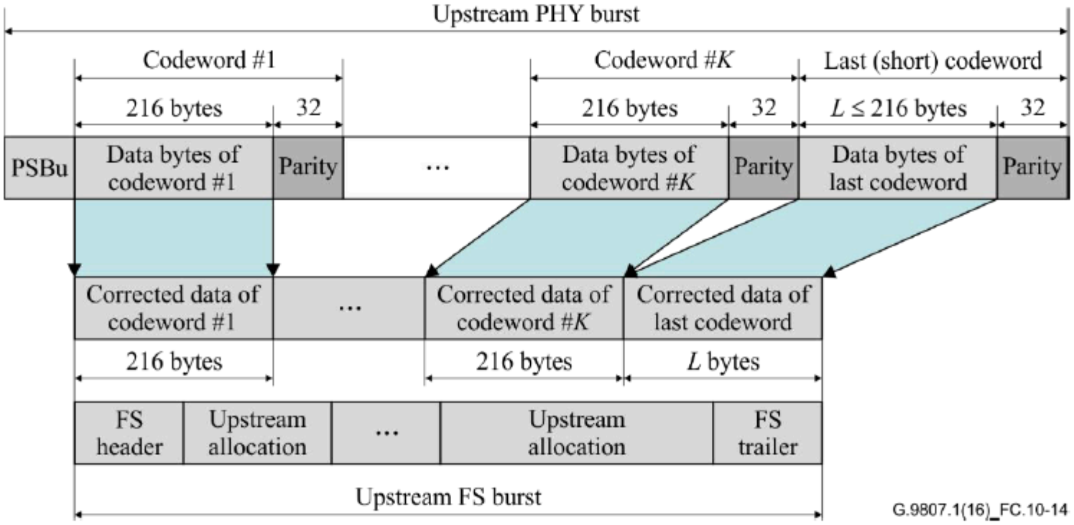


Figure 5-40 [15] 10 Gbps FEC decoding at OLT

5.9 ONU activation in XG-PON & XGS-PON

ONU activation follows the same process as discussed in 4.3.4 (G.984.3) with some differences.

As discussed earlier, the OLT is TDMA orchestrating the all upstream bursts in order to avoid collision, it achieves that by measuring a round trip time (RTD) for each ONU individually, the RTD is described as the time passes from transmitting a downstream frame to a specific ONU until the OLT receives the first response from this ONU, the RTD time will include the signal propagation delay within the fiber plus the ONU response time, the OLT calculates what is called “equalization delay” time or (EqD) for every ONU, then it communicates the EqD to each ONU. Every ONU has to wait an amount of time equals the EqD before it transmits its upstream physical burst. EqD is needed because each ONU is distanced differently from the OLT, and also because the BWmap process logically assumes that all ONU’s have the same distance from the OLT.

Then, what this has to do with the ONU activation? The answer for this question is that during the activation process, there are three phases, and the last phase will include measuring the EqD before the ONU becomes fully operational. The three ONU activation phases are, synchronization phase, serial number acquisition phase and ranging phase.

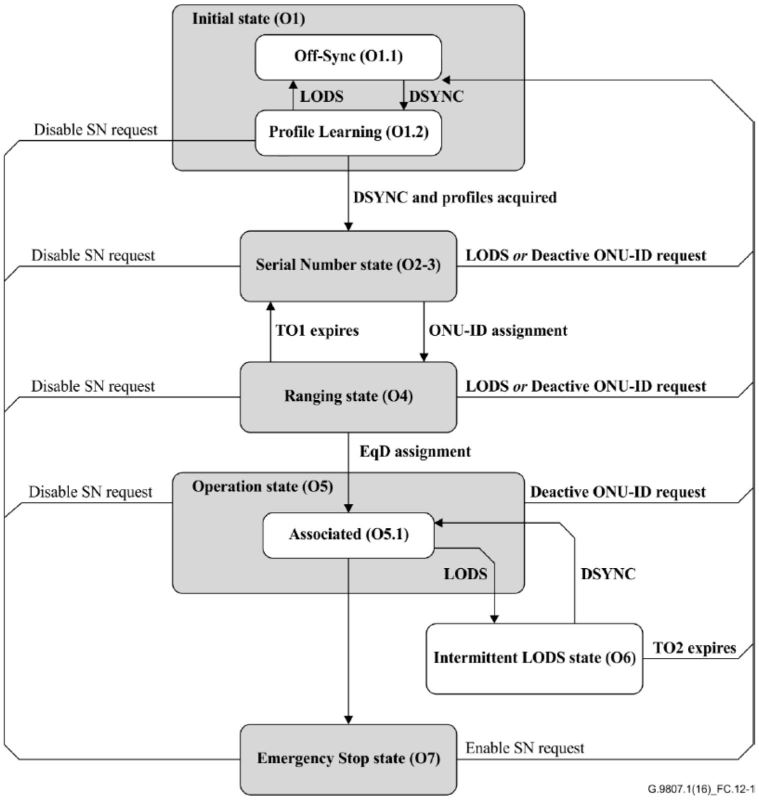


Figure 5-41 ONU state diagram

In the synchronization phase, the ONU will start hunting the downstream frame and try to achieve synchronization with it. In the serial number acquisition phase, the ONU will specify the burst profile it should use, then it waits for an OLT serial number request which called a “serial number grant” or (SNG), the SNG is a downstream allocation directed to a broadcast ONU-ID 1023, with a PLOAMu flag is set, once unregistered ONU’s sees this broadcast, it will respond with its serial number. During the serial number grant, the OLT will announce a “quite window” also called a “discovery grant” during which all operational ONU’s (that already have a default ONU-ID assigned) will suspend their transmission, in order to not to collide with a serial number upstream response from a new unregistered ONU. Unregistered ONUs will follow ALOHA protocol to avoid collision between each other when responding to the OLT SNG. Once the OLT learns the ONU serial number, it will assign a unique default ONU-ID to the ONU to be used in future communication.

During the ranging phase, the OLT measures the RTD by sending a downstream communication to the specific ONU, using a default Alloc-ID equal to its assigned default ONU-ID with a PLOAMu flag field set, the OLT then uses the response from the ONU to calculate an EqD and communicate the same to each ONU. Throughout the activation process the ONU goes through seven states as shown in Figure 5-41, the Initial state (O1), the Serial number state (O2 –O3) and the ranging state (O4) covers the three explained phases, after executing those three phases the ONU goes to an Operation state (O5), in this state the ONU becomes fully operational where it can send and receive data, if for any reason the ONU losses synchronization then it moves to an Intermittent LODS state (O6) where it tries to re-establish the synchronization, if it succeeds then it goes back to the O5 state, if it fails to synchronize until a timer TO2 expires then it goes back to O1 state. Last state is called “emergency stop” state (O7) where the ONU gets disabled by the OLT, however if it wants to resume transmission then it has to restart from state O1.

5.10 XG-PON Management and control

Figure 5-42 shows the XGTC protocol stack, and summarizes the information flow through all sub layers. Note that the PLOAM processor is an individual entity resides inside the XGTC layer (exists in both OLT and ONU), unlike the OAM processed which consists of different entities existent inside and outside the XGTC layer, according to the function that each field serves, this is because the embedded OAM messages is a group of separate fields that live inside the XGTC header, in upstream BWmap, Identification field and the BIP. [17] [15]

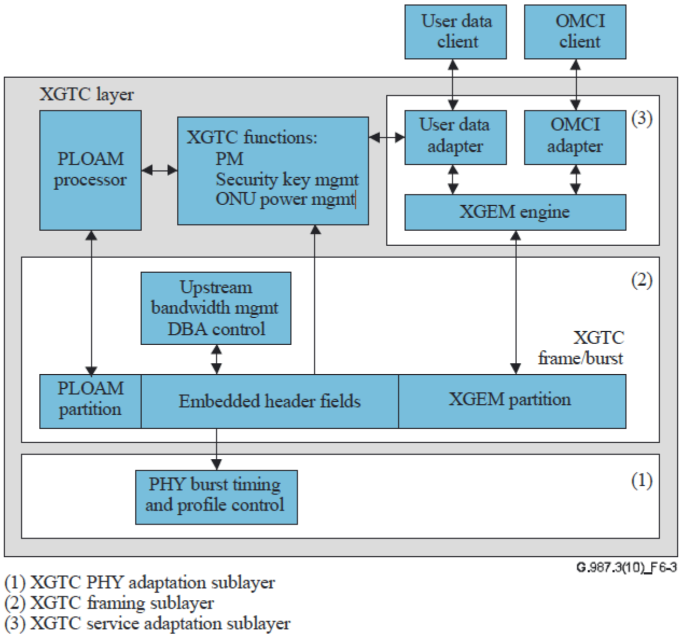


Figure 5-42 [17] XGTC protocol stack and information flow

Influence by the GPON GTC M/C plane, the management and control plane structure in XGPON is quite similar. It is also represented by three channels, ONU management and control channel (OMCC) that uses OMCI messages protocol, embedded OAM and PLOAM, where the last two are used for physical layer control and management. The OAM will carry the control information for critically urgent signals that should be executed immediately on the physical layer without further waiting, for an example functions like an upstream interval allocation timing of each burst, encryption key management and dynamic bandwidth reporting. On the other hand, the PLOAM organized in a dedicated independent field block which carries PLOAM messages that exchanged between the OLT and the ONU, to perform tasks such as ONU management or ONU activation and all of its subsequent processes. Figure 5-43 shows the structure of the management and control channels in the XGS-PON which is identical in all NG-PON technologies.

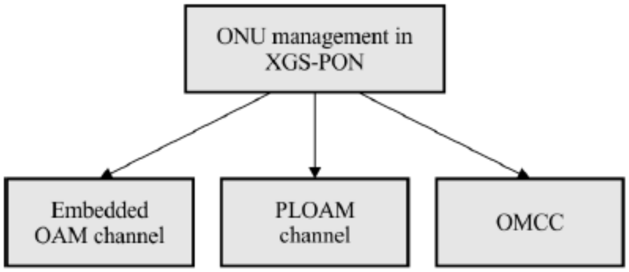


Figure 5-43 [15] ONU management channels in XGS-PON

The OMCC channel (uses an OMCI protocol/channel), already discussed in 4.5, it takes care of the ONU management functions in the XG-PON. In fact, the OMCI protocol works the same in all NG-PON systems as per standard (ITU-T G.988). The

OMCI protocol data units (PDU) will be encapsulated within an XGEM payload, using a dedicated Port-ID. [17] [15]

5.11 NG-PON summary

This chapter discussed the XG-PON and the XGS-PON protocol's reference architecture and possible co-existence scenarios between new and legacy technologies, the wavelength band distribution within each co-existence scenario and explained how this will enable a fixable migration towards the new technology implementation, while keep providing the same quality services to legacy GPON users. This chapter discussed the transmission convergence layer for both systems in fair details, highlighted the main differences between the TC layer structures, and explained how these differences will help to achieve the high-performance expectations of the new NG-PON systems.

Chapter 6 40 Gigabit – PON (NG-PON2)

6.1 Introduction

This chapter will cover NG-PON2 architecture and requirements, it will discuss system specifications such as wavelengths and speeds. In addition, it will disclose some migrations and coexistence scenarios with the legacy PON systems in a brownfield and also will discuss Greenfield implementation.

6.2 Architecture

According to ITU-T G.989, the NG-PON2 OLT should support multiple wavelength channels, these channels organized in channel pairs (CP), each channel pair contains one downstream channel and one upstream channel on a single fiber. The channel pairs are combined together by a wavelength multiplexer (WM). Figure 6-1 shows NG-PON2 logical architecture (Green-field scenario), be aware that all the channel terminations (CT) shown are conceptually exists in a single NG-PON2 OLT. [18]

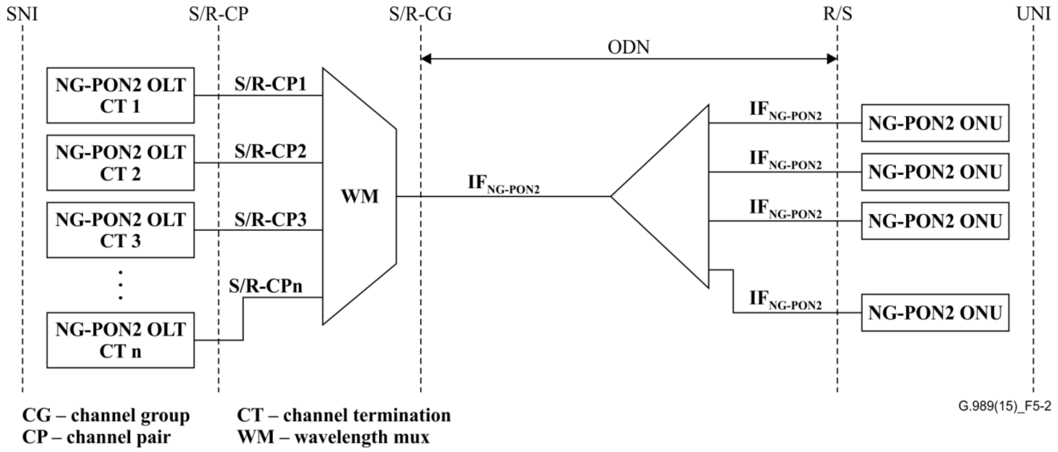


Figure 6-1 [18] NG-PON2 (Greenfield architecture)

NG-PON2 reference architecture enables a P2MP mode and a virtual P2P connectivity mode. The P2MP mode system shares multiple wavelength channels by means of a time and a wave division multiplexing (TWDM), where several CPs combined by a WDM and each CP could be shared between multiple ONUs by a TDM. The CPs can be deployed gradually as the system grows, in other words it could start with one CP (similar to the legacy PON systems), and adds a new CP as needed, in order to support the increase in capacity. G.989.1 standard specifies 4 to 8 TWDM CPs configurable per an OLT. Regarding the P2P NG-PON2, it provides a dedicated wavelength channel per an ONU for the downstream and a dedicated wavelength channel per an ONU for the upstream transmission, also multiple p2p channels could be combined/de-combined from/to a single OTL by a WDM.

Similar to the XG-PON1 standard, an NG-PON2 architecture supports coexistence with the legacy PON systems, therefore it should be able to support old bit rates of a 10Gbps in both directions, a 10Gbps in the downstream and a 2.5 Gbps in the upstream as well as a 2.5 Gbps in both directions. [18]

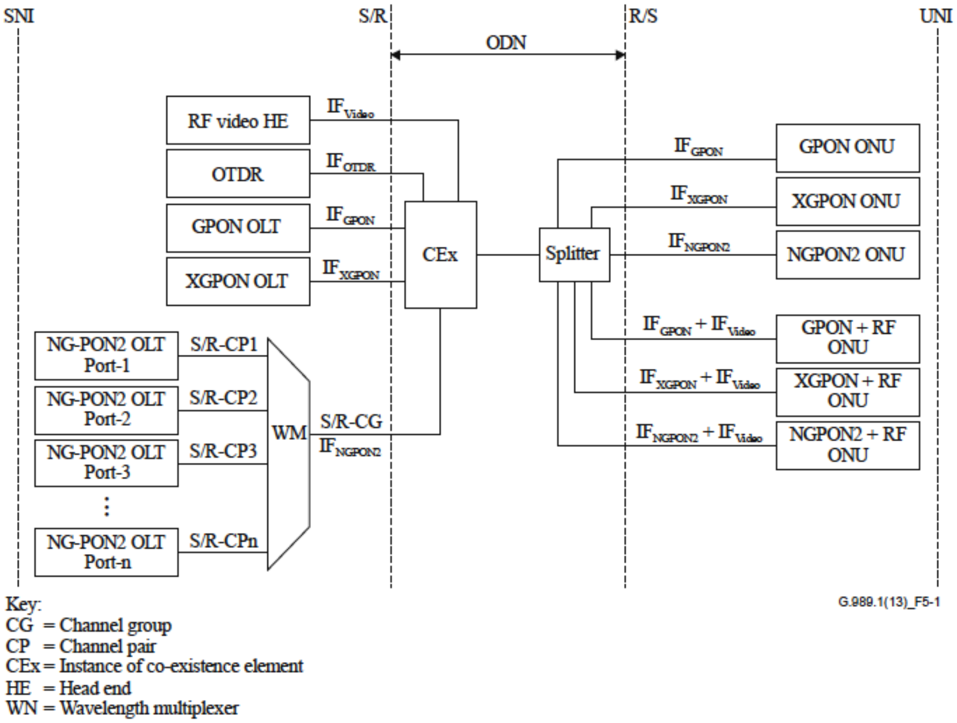


Figure 6-2 [19] NGPON2 reference architecture and coexistence.

In ideal world, the system fiber reach (passive) is a minimum of a 40 km and up to a 60 to 100 km using a mid-span reach extenders. The maximum differential fiber distance is up to a 40 km (configurable max DD 20 km and 40 km). The OLT is able to support a split ratio of 1:256 or higher. However, these numbers vary in relation with each other, though the system must be able to maintain the 40 Gb/s downstream capacity (multiple of 10Gbps channels) & 10 Gbps upstream capacity (multiple of 2.5 Gbps channels), a 20 km reach with a minimum of 1:64 split ratio or could achieve a longer reach with a lower split ratio. Another suggested combination is mentioned in G.989.1 as per the following:

- 40 Gbps upstream capacity (four 10 Gbps upstream channels), 20 km reach and minimum 1:64 split.
- 2.5 Gbps per downstream channel and 2.5 Gbps per upstream channel, 40 km

reach with 1:32 split.

- P2P WDM supported, explained as a dedicated CP configured between NG-PON2 OLT and ONU where these P2P CP are also combined using WM. P2P WM should be able to support coexistence.

Figure 6-2, illustrates the NG-PON2 reference architecture beside a coexistence with another legacy PON system. The coexistence was achieved by a wavelength multiplexing, however due to the overload of the term, and to be able to distinguish between the function of each wavelength multiplexers. Therefore, the wavelength multiplexing used for coexistence is referred to as a *"Coexistence Element" (CEx).*[19]

6.3 Coexistence and migration path

One of the most important aspect about NG-PON2 technology is its high coexistence flexibility with legacy PON systems (GPON and XG-PON1), to be more specific, NG-PON2 could share the use of already deployed ODN infrastructure, without the need to upgrade the ODN, while other old PON systems continue to provide services simultaneously, therefore new NG-PON2 services could be presented to customers without affecting ongoing old services. NG-PON2 provides a smooth staged migration solution for the existence old deployment, which enable service providers to easily upgrade their infrastructures toward the next generation services without disturbing the ongoing businesses.

Besides reusing the deployed ODN components, as a requirement of a coexistence implementation, the NG-PON is able to operate in a wavelength that is not occupied by another legacy PONs. Nonetheless, it can re-use one of the dedicated legacy PON spectrums if that PON system was not present in the implementation. The Old system services are also supported, such as RF-video overlay.

G.989.1 provided three migration path options, straight, flexible and all-embracing migration.

In a straight migration scenario, the old technology is upgraded to the next technology first before moving toward NG-PON2, in other words, the GPON is fully upgrade to an XG-PON1, and then the XG-PON1 is fully upgraded to an NG-PON2. In this scenario, the GPON should be totally removed from the ODN before the implantation of the NG-PON2 begins, therefore the NG-PON2 will be able to re-use the GPON spectrum while coexists with the XG-PON1.

In a flexible migration option, there is a direct migration from a GPON to an NG-PON2 (no XG-PON1), so there will be a co-existence requirement only between NG-PON2 and GPON.

The third and last migration option is an all-embracing migration, where the three PON systems co-exist together, GPON, XGPON1 and NG-PON2. This model could be the best option for the business as it enables a more flexibility in upgrading the two older PON systems towards the NG-PON2, however it requires a careful planning due to the wide spectrum in use.

During a coexistence between NG-PON and legacy PON’s, the old ONU and OLT should remain the same, unless if a subscriber wants to upgrade to the NG-PON2 service, then all he needs to do, is to simply replace the old ONU with a new NG-PON2 ONU. [19]

6.4 NG-PON2 additional key system characteristics

The following are additional NG-PON2 system characteristics, which make NG-PON2 an upgrade friendly for the majority of service providers as it helps them to flexibly grow their businesses without a significant increase in their Capex and Opex.

- In NG-POn2, the deployed ONUs are colorless, which means they are not bound to a specific wavelength, they could operate on almost all known PON bands (O, E, S, C and U).
- Spectral flexibility of the NG-PON2 is demonstrated by its ability to operate under multiple wavelengths or wavelength bands, which enables capacity upgrades in a modular fashion (i.e. pay as you grow). To facilitate the smooth capacity increase, the OLT supports flexible channel group for a TWDM and a P2P WDM. Furthermore, spectral flexibility allows coexistence with legacy PON without signal interference.
- NG-PON2 supports a rogue ONU/OLT detection and mitigation by monitoring their traffic and detects any situation where the ONU or the OLT violate the bandwidth allocation, for an example if the ONU transmitted during the wrong allocation, or if the OLT used a wrong wavelength band that is belong to another OLT, in either case, it will cause a collision and should be detected and solved by the OLT.
- The ONU is remotely manageable where the system provides a full FCAPS (Fault, configuration, accounting, performance and security) management for the ONU.

The system also supports a PON monitoring and supervision to identify performance issues, faults and their locations which helps to provide an enhanced customer service experience.

- NG-PON2 supports power reduction techniques that helps to reduce the energy cost and improves the greenhouse emission. A new power reduction technique called OLT-port sleep is provided, by combining low TWDM traffic from multiple CTs (see Figure 6-3) into a single CT or a port, then the OLT places the rest of free CTs into a sleep mode, which greatly saves power as illustrated in Figure 6-4. Again, thanks to the colorless ONU and the spectrum flexibility which made the OLT-port sleep mode is possible to implement. [19]

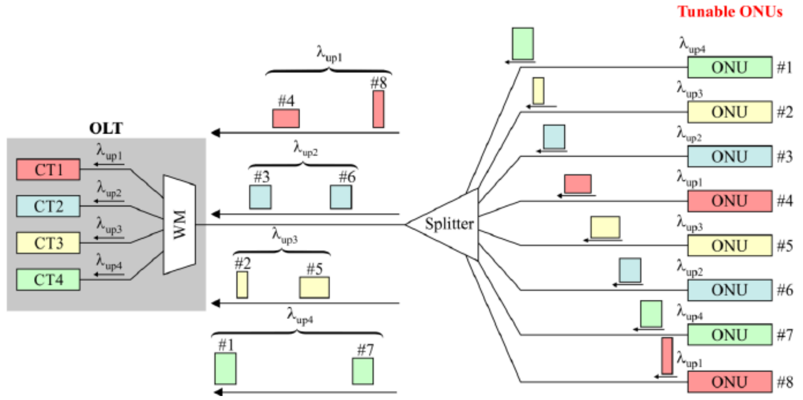


Figure 6-3 [19] NG-PON2 OLT ports before activating port sleep mode.

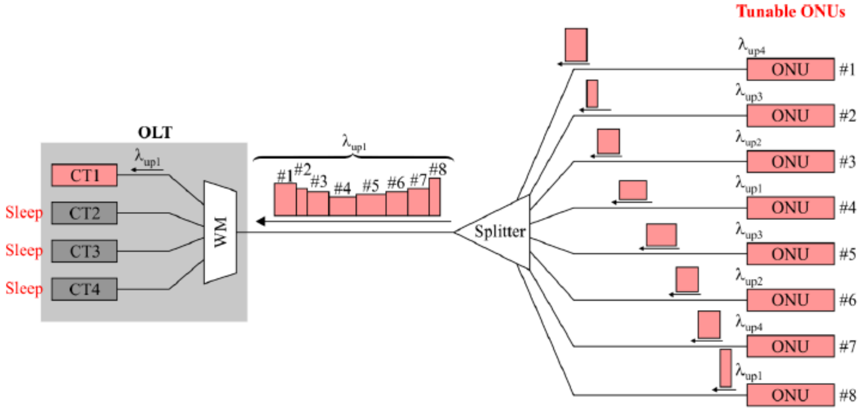


Figure 6-4 [19] NG-PON2 OLT port sleep mode after activation.

6.5 NG-PON2 wavelength plans

In determining NG-PON2 wavelength many factors should be taken in consideration. The coexistence with other legacy systems should be considered in order to not to interfere with their frequency bands, Figure 6-5 shows wavelength plans for legacy PON systems.

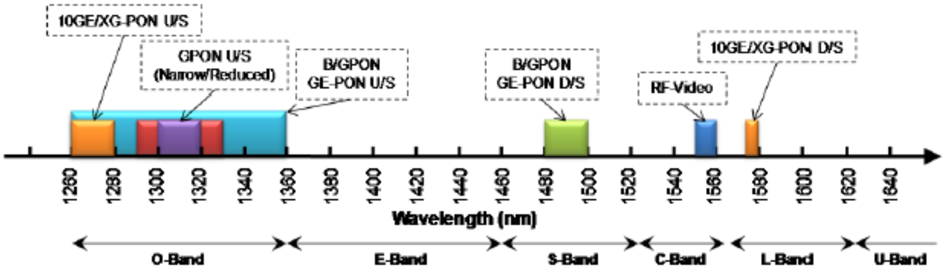


Figure 6-5 [19] Wavelength plans for legacy PON systems

Due to the fact that most of the PON implementations utilizes a single mode fiber (SMF), the later performance is dependent upon the frequency band. For an example the SMF Attenuation is very low in the C-band & L-band, and it's very high in the E-band between (1360-1430 nm), not forgetting to mention that the influence of the fiber age, since older fibers will produce a higher attenuation. Another aspect is the chromatic dispersion which increases proportionally with the wavelengths. Furthermore, the chromatic dispersion reduces the physical reach with higher data rates, Figure 6-6 shows an SMF attenuation and the chromatic dispersion in relation with the wavelength and the fiber age. [19]

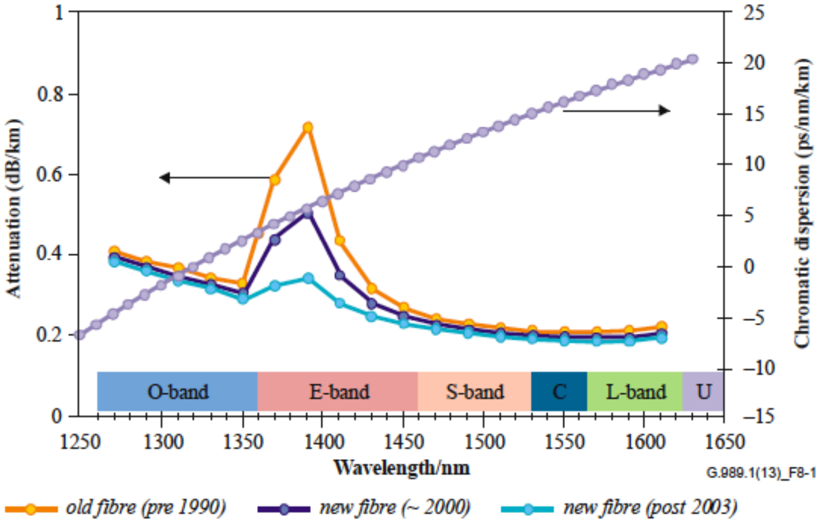
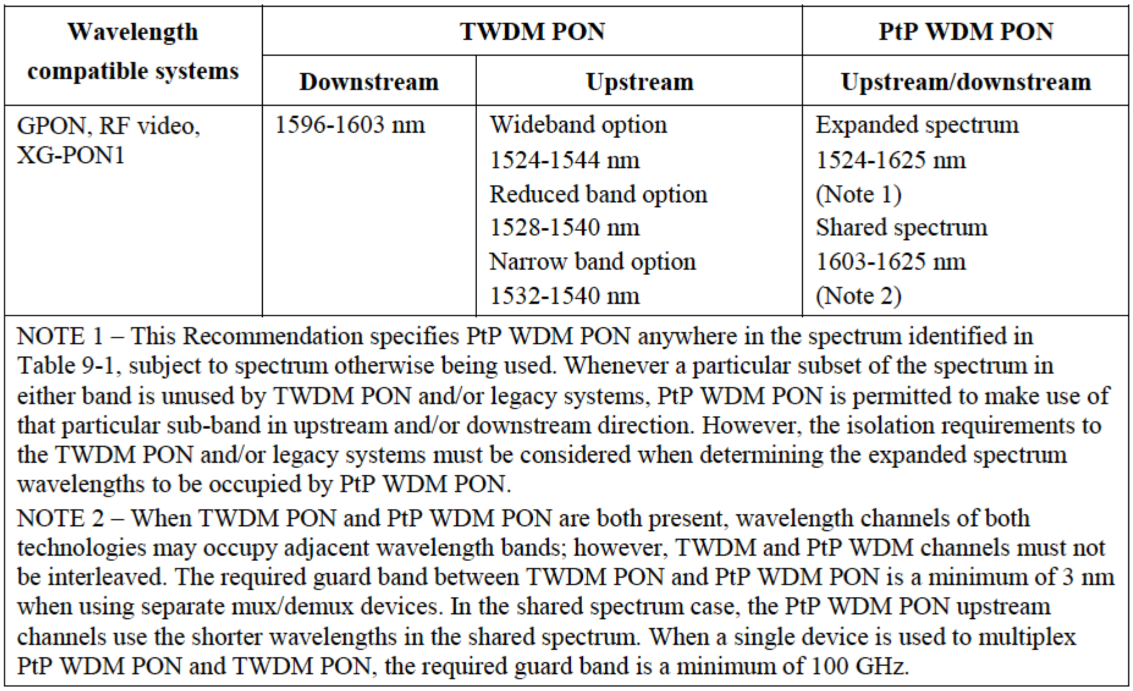


Figure 6-6 [19] SMF attenuation and chromatic dispersion in relation with wavelength

Additional factor affects the NG-PONs performance, is the heat that's produced by different hardware components in relation with used wavelength, where the generated heat could decrease the photodiode responsiveness. Another design concern is the usage of legacy filters in an old ODN, where some of these filters adds unnecessary constraints on the frequency available for use, such as RF-video filter which uses a guard band that occupies most of the C-band, as explained in the previous paragraph, amongst all other bands, C-band has the lowest SMF attenuation, and most of the available commercial Erbium doped fiber amplifiers (EDFAs) can operate on C-band, which makes it the most desirable band for the PON implementation, in other words, it’s not the best idea to use it as a “guard band”!!! [19]

Table 6-1 [20] NG-PON2 wavelength bands plan



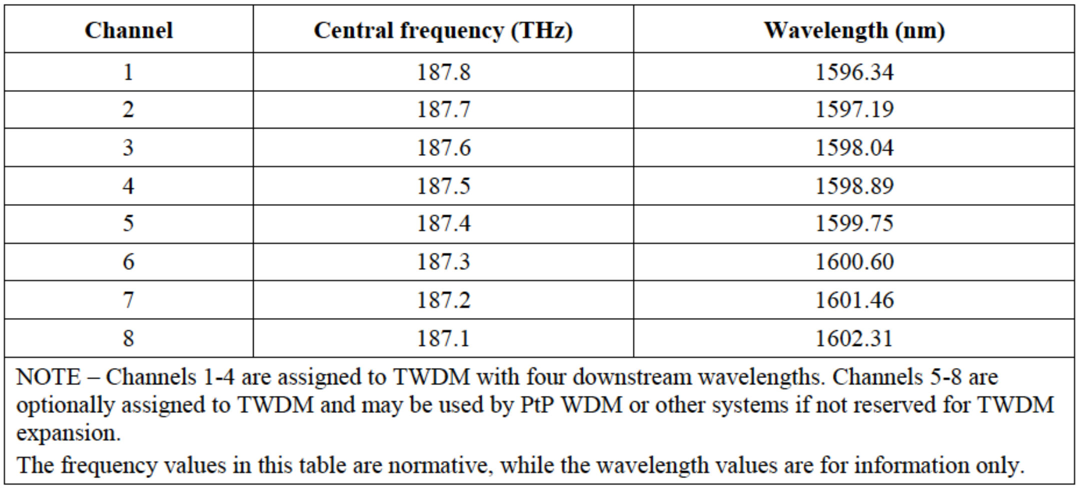
Wavelength compatible systems	TWDM PON		PtP WDM PON
	Downstream	Upstream	Upstream/downstream
GPON, RF video, XG-PON1	1596-1603 nm	Wideband option 1524-1544 nm Reduced band option 1528-1540 nm Narrow band option 1532-1540 nm	Expanded spectrum 1524-1625 nm (Note 1) Shared spectrum 1603-1625 nm (Note 2)

NOTE 1 – This Recommendation specifies PtP WDM PON anywhere in the spectrum identified in Table 9-1, subject to spectrum otherwise being used. Whenever a particular subset of the spectrum in either band is unused by TWDM PON and/or legacy systems, PtP WDM PON is permitted to make use of that particular sub-band in upstream and/or downstream direction. However, the isolation requirements to the TWDM PON and/or legacy systems must be considered when determining the expanded spectrum wavelengths to be occupied by PtP WDM PON.

NOTE 2 – When TWDM PON and PtP WDM PON are both present, wavelength channels of both technologies may occupy adjacent wavelength bands; however, TWDM and PtP WDM channels must not be interleaved. The required guard band between TWDM PON and PtP WDM PON is a minimum of 3 nm when using separate mux/demux devices. In the shared spectrum case, the PtP WDM PON upstream channels use the shorter wavelengths in the shared spectrum. When a single device is used to multiplex PtP WDM PON and TWDM PON, the required guard band is a minimum of 100 GHz.

Table 6-1 shows wavelength plans for NG-PON2 where it will be able to co-exist as an overlay with legacy PON systems, the selection of a band is greatly dependent on the coexistence scenario. For an example, a shared spectrum should be used with a full coexistence where all of legacy PONs plus a TWDM and a P2P NG-PON are present. Furthermore, the expanded spectrum facilitates the ability of re-using some of the legacy PON bands when they are not existing. [20]

Table 6-2 [20] downstream channels plan



Channel	Central frequency (THz)	Wavelength (nm)
1	187.8	1596.34
2	187.7	1597.19
3	187.6	1598.04
4	187.5	1598.89
5	187.4	1599.75
6	187.3	1600.60
7	187.2	1601.46
8	187.1	1602.31

NOTE – Channels 1-4 are assigned to TWDM with four downstream wavelengths. Channels 5-8 are optionally assigned to TWDM and may be used by PtP WDM or other systems if not reserved for TWDM expansion.

The frequency values in this table are normative, while the wavelength values are for information only.

To be more specific, in a P2MP TWDM, an OLT supports an 8 Channels in the downstream that could be assigned to a P2MP TWDM or split them in to four channels for a TWDM and another four for a P2P WDM, Table 6-2shows assigned frequency/wavelength per channel.

An example of a narrow band with four upstream channels is (1532.68 nm, 1533.47 nm, 1534.25 nm and 1535.04 nm) with 100 GHz spacing.[20]

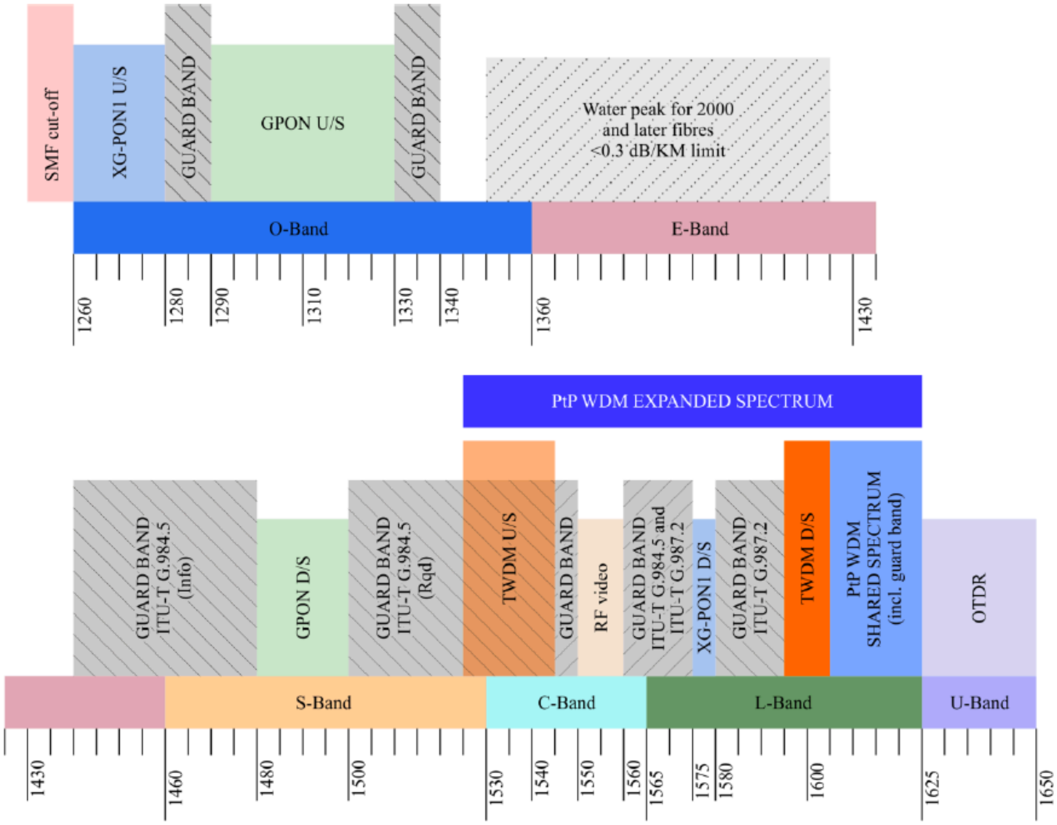


Figure 6-7 [20] NG-PON2 wavelength plan when coexist with GPON, XG-PON1, RF-Video and OTDR

Figure 6-7, summarizes NG-PON2 wavelength plan when coexists with GPON, XG-PON1 and RF-Video overlay. More details regarding the channel grid design and spacing is covered in ITU-T G.989-2 Appendix VII.[21]

6.6 Bit rate

NG-PON2 bit rate is as specified in Table 6-3, please note that, the rate value is per a channel However, multiple different rates could exist simultaneously in the same group of channels. [21]

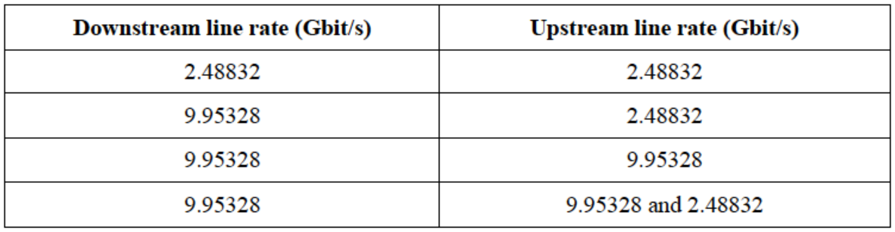
Table 6-3 [21] TWDM bit rate options

Description	Bit rate in downstream/upstream Gbps
Basic rate	9.95328/2.48832
Option 1 rate	9.95328/9.95328
Option 2 rate	2.48832/2.48832

6.7 NG-PON2 transmission conversions (TC) layer

NGPON2 has two different structures for a TC layer, a TC layer for TWDM channels and another TC layer for P2P WDM channels. The TWDM TC layer is quite similar to previously discussed TC layer in XG-PON and XGS-PON, in terms of the structure, the bandwidth allocation and the system management. However, the main difference is that the NGPON2 OLT TC layer consists of multiple CT ports that supports different line rates, as shown in Table 6-4, same for the ONUs that support different line rates, as shown in Table 6-5. [22]

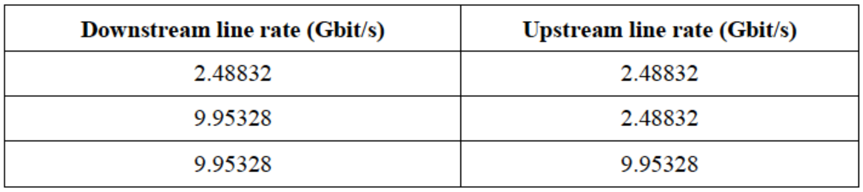
Table 6-4 [22] NGPON2 OLT supported bit rates in TWDM PON



Downstream line rate (Gbit/s)	Upstream line rate (Gbit/s)
2.48832	2.48832
9.95328	2.48832
9.95328	9.95328
9.95328	9.95328 and 2.48832

The general TWDM TC layer structure remains almost unchanged compared to XG-PON, however the only difference is that its ability to support multiple CTs and different line rates.

Table 6-5 [22] NGPON2 ONU supported bit rates in TWDM PON



Downstream line rate (Gbit/s)	Upstream line rate (Gbit/s)
2.48832	2.48832
9.95328	2.48832
9.95328	9.95328

Management of TWDM PON is similar to XG-PON with the introduction of two additional PLOAM channel options; the primary option or (In-band transportation option) and an alternative option or (AMCC transportation option) as shown in Figure 6-8. The AMCC option is only applicable at an ONU activation to Serial_Number_ONU and Tuning_response PLOAM messages. AMCC is defined as a low data rate scheme to allow a discovery of an ONU that is insufficiently calibrated, where insufficiently calibrated ONU is prohibited from using an operational TWDM channel to prevent interfering with another operational ONUs. In Addition, in NG-PON2 there is a completely new management channel/protocol that doesn't exist in any of the previous PON technologies called an "Inter Channel Termination Protocol" or (ICTP). ICTP channel is added to an NGPON2 in order to support a multi-wavelength channels, where it's used by the CTs to communicate internally amongst each other.

ICTP also supports a channel profiling configuration and status sharing, ONU activation, ONU handover and a rogue ONU mitigation. [22]

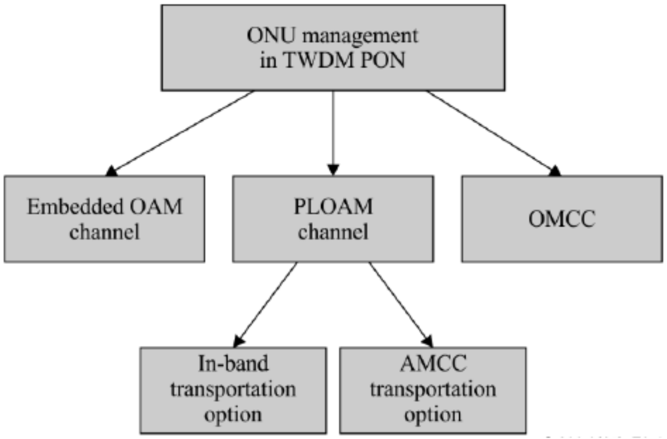

ONU management in TWDM PON
Embedded OAM channel
PLOAM channel
OMCC
In-band transportation option
AMCC transportation option

Figure 6-8 [22] NGPON2 ONU management channels

Likewise, the TWDM architecture is a very similar to the previous PON TDM architecture, the only difference is that its ability to support multiple TWDM channel terminations (CTs), each CT is linked with a channel, the system can support up to eight CTs.

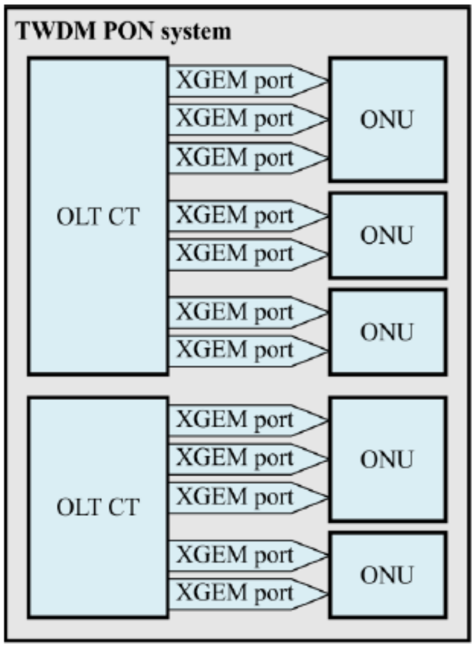


Figure 6-9 [22] TWDM downstream architecture

The downstream OLT will use an XGEM port-ID to TDM-multiplex the user and OMCI data (SDU), in the upstream it will also use XGEM port-ID in each burst within a specific allocation interval (Alloc-ID) to TDMA-multiplex between multiple users' data (ONUs). Figure 6-9 shows the downstream TWDM PON architecture, and Figure 6-10 shows the upstream architecture. [22]

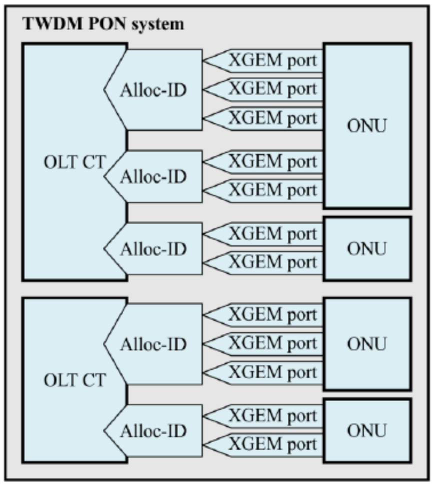


Figure 6-10 [22] TWDM upstream TDMA architecture

6.8 Why NG-PON2 is needed?

There's no doubt that the future demand for more bandwidth and higher speeds will continue to grow, according to Ovum 2017, by 2022 a 10G PON may overtake the legacy GPON. Currently various operators around the world are deploying 10G PON driven by technology's hunger for more bandwidth, higher bit rates demand, service convergence and mobile front-haul 5G readiness, Figure 6-11.[20]

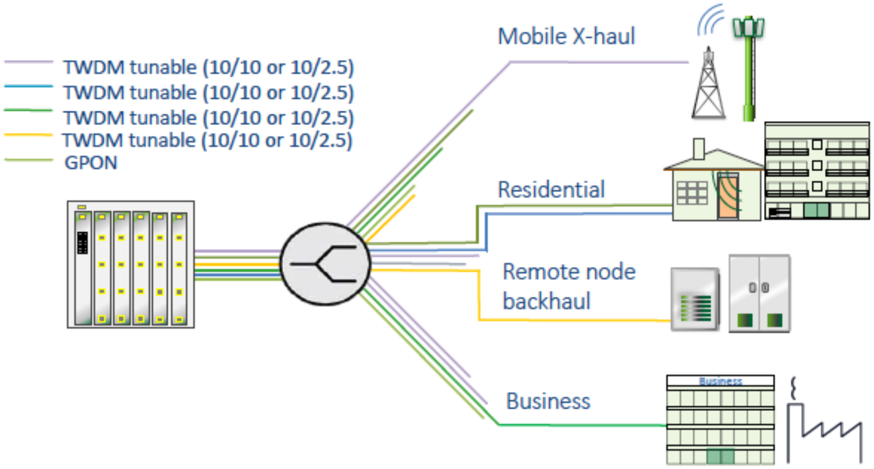
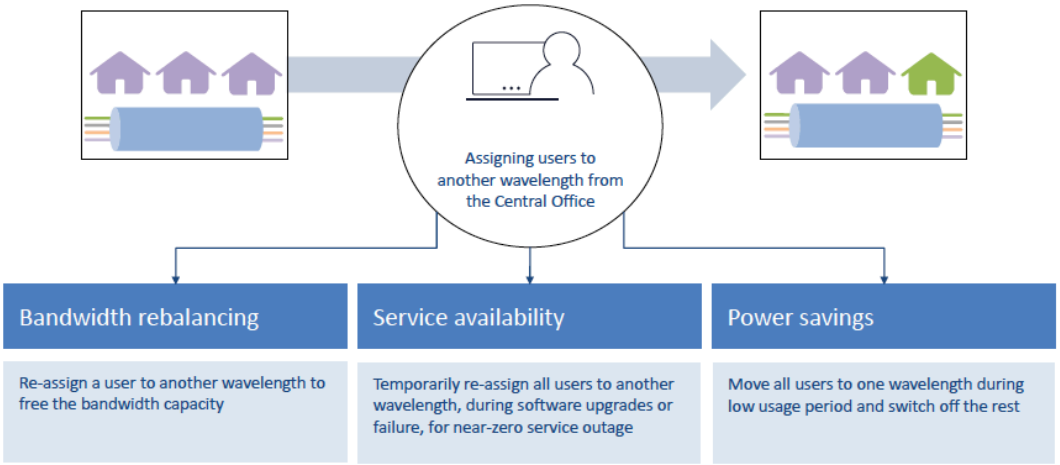

TWDM tunable (10/10 or 10/2.5)
TWDM tunable (10/10 or 10/2.5)
TWDM tunable (10/10 or 10/2.5)
TWDM tunable (10/10 or 10/2.5)
GPON
Mobile X-haul
Residential
Remote node backhaul
Business

Figure 6-11 [20] TWDM enables sing PON for more added value

NGPON2 TWDM unlocks the full potential of fiber broadband, as it leverages the traditional fiber to add more value, where NG-PON2 provides an efficient (all-in-one) single PON implementation that supports the residential fiber, the Mobile backhaul/front-haul and businesses segments, with a very flexible operational efficiency that became possible through a TWDM-PON. The fact that TWDM-PON supports multiple wavelength channels, made new features are possible, such as bandwidth rebalancing where the OLT can redistribute bandwidth capacity by changing the ONU wavelength. Also, a better service availability could be achieved, for example during software upgrades or failures, where some users may be assigned an alternative wavelength and not to suffer outages. In addition, a power saving as described in section 6.4 goes well with reducing the greenhouse emission and cutting energy costs, Figure 6-12 illustrates these new features. Moreover, the ability to add new wavelengths as needed, allows a new "pay as you grow" business strategy. [20]


Assigning users to another wavelength from the Central Office
Bandwidth rebalancing
Service availability
Power savings

Re-assign a user to another wavelength to free the bandwidth capacity

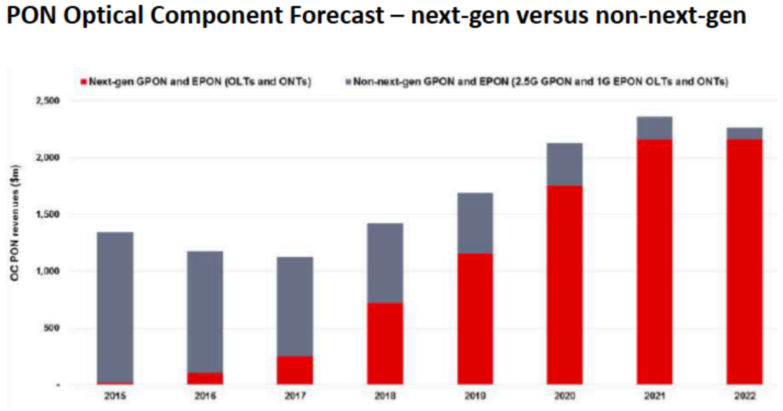
Temporarily re-assign all users to another wavelength, during software upgrades or failure, for near-zero service outage

Move all users to one wavelength during low usage period and switch off the rest

Figure 6-12 [20] NG-PON2 TWDM increases operational efficiency

6.9 What is next?

NG-PON2 TWDM technology will make a paradigm shift within the PON industry, it satisfies the market's demand and the uprising megatrend applications such as IoT, 5G, gigabit Wi-Fi, self-drive vehicles, cloud services, 8K IPTV, virtual reality and much more.



Source: Ovum

Figure 6-13 [20] NGPON2 dominates the forecast

The NG-PON2 ability to offer a high-bandwidth, a low latency, a growth and smooth migration flexibility makes it the perfect future solution for service providers, Figure 6-13 predicts that the NG-PON2 will become the de-facto deployment in near future.

In addition, the fast pace deployment of network virtualization technologies, such as SDN & NFV in cloud data centers, opens the door for a new architecture that offers modularity, simplifies scalability and agility as well as adds value to the service providers.

Central office re-architecture as a data center (CORD), which is the topic for this thesis, is a complete platform that revolutionize the CO design as a data center and help consolidate a converged network services within one flexible smart platform. In Section 2, we will talk about CORD in fair details and discuss some real-life use cases. [20] [23]

6.10 NGPON2 summary

This chapter discussed NG-PON2 reference architecture, possible co-existence and migration scenarios with legacy GPON and XG-PON. It discussed a new concept of multiple wavelength channels sharing through a TWDM and P2P-WDM. It explained the NG-PON2 wavelength plan challenges. Also, it discussed new system features that didn’t exist with the previous PON technologies.

Section 2: CORD Platform

Chapter 7 Introduction to SDN and NFV

7.1 Introduction

Network routers (or L3 switches) main function is to route and forward data packets between different network domains in order to connect end users to each other or to a server. In other words, any time you request information on the internet (accessing a website for an example), you will generate a traffic in both directions, this traffic will pass through several routers, and probably some of the packets will travel through different paths before you receive the requested data.

To be on the same page, from now on, this report will use the word switch to refer to either a network router or a L3-switch. Network switches will perform traffic forwarding decisions based on a predefined routing protocols. Routing protocols are special algorithms that run by switches to calculate the best possible path that the packet may traverse, based on a predefined rules and metrics, an example of these metrics is; shortest path, link bandwidth and/or latency, current load…etc. There are several known routing protocols that varies in their complexity and mechanism of calculating routing paths, however, all of them serve one, and only one main goal, which is to find and forward network traffic through the best path, in order to better utilize the network bandwidth, thus provides the best possible user experience. [24]

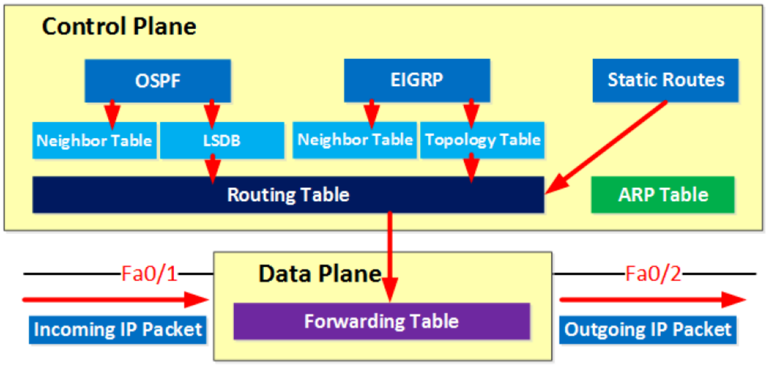


Figure 7-1[25] router/L3-switch architecture shows control and data planes

As shown in Figure 7-1, a switch architecture contains two planes, a control plane and a data plane. The control plane is the switch part that is responsible for making routing decisions by running routing algorithms to calculate the best path, then it stores calculation results inside routing tables. The Data plan is responsible for packets fast (wire speed) forwarding, based on forwarding-tables fast lookup, in other word there is no much intelligence included in the data plane, which is the reason that it is called a “forwarding plane”. Forwarding tables in the data plane are populated from the already calculated routing tables of the control plan. [24]

In order to calculate the network performance, let’s try to find out the delay that a single packet traverse a network, from the moment it arrives to a network edge switch ingress, until it leaves the egress switch’s port and steps on a wire, which is defined as T, and expressed by the following equation:

$$T = \sum_{i=0}^{H}(t_i^p + L_i \times v_i + t_i^w)$$

Equation 1[24] Network Performance in Sec

Where H is the hop count, t^p is the transmission delay needed by a network interface card (NIC) port to physically place the bits of the packet on the wire and it’s affected by the buffering time on the NIC, the link speed, and the size of packet. $L_i \times v_i$ equals the time that a signal takes to spread over a link, which equals the link’s length (L_i) times the signal’s velocity (v_i = speed of light). t^w equals the packet delay inside a switch, which is the total time that the packet will spend inside a switch from the moment it enters the switch port, until it leaves the output port, it’s also affected by the routes calculation and buffering time. [24]

Therefore, calculating routing paths could be a time-consuming process, since it’s done by a software execution over the switch’s processor, where most likely other running processes are sharing the same processor adding an extra processing delay. The processing delay could get longer if it was implementing a network policies and a QoS as in advanced switches. Moreover, routing path calculation is performed for each packet without a matching forward entry. Packets that have the same destination domain and/or class of service will use the same forwarding entry, but because of routing tables entries expire after a certain time is passed, hence they need to be recalculated/updated frequently. Furthermore, the expiration could be triggered by timers or an external event, such as a network status change, for an example if a link goes up or down. The dynamic change of routing paths of a network in response to a trigger is called a “network convergence”. The network convergence time is defined as the time that a network takes to dynamically adjust its traffic paths in response to a trigger, which is comparable to a domino effect, where a small link status change in a network will cause a wave of changes through the entire network switches until the new paths are updated. Back to equation 1, the above-mentioned events will cause $t^p \, and \, t^w$ to grow and drastically affect the network performance. The fact that within a traditional network, every switch is independent, thus it takes its own routing decisions and calculates its own routing/forwarding tables, this means unnecessary duplication of routing protocols execution at every switch, therefore same protocol stack run on every switch over and over again, that is an extra duplicated processing time at each switch. In addition, this independence comes with the cost of management traffic overhead, since the independent switches need to talk to each other in order to manage the traffic, create adjacencies and detect changes (e.g. hello messages), which varies depends on the protocol stack in use. In addition, some protocols allow each switch to realize the entire network topology and be fully aware of all other active switches through exchanging large link status tables and keep refreshing these tables frequently, this is a quite large overhead which slices a big chunk of the network bandwidth. Now, the TCP/IP traffic is very bursty in nature, and if the network resources were not over-provisioned enough, therefore network usage peaks will result in a considerable delay and even congestion, where switches port's buffer will overflow causing a lower priority packets to be dropped, to add insult to injury, the transmitting TCP layer will keep retransmitting the dropped traffic desperately, significantly degrading the network performance. The only solution to fix traffic peaks in a traditional network is to overprovision its resources, which means low network utilization most of the time and increase both Opex and Capex. [24]

Another issue is the traditional network management, where in order to build a new topology or install a new feature, you need to configure each device individually, this might take a very long time and if the network is large enough it might take several days to weeks, during which you have to live with all the side effects of service availability, configuration errors, failures…etc. [24]

Now let's think about the performance issues of a private network that connects a cloud data center (DC), which carries the internal traffic between clusters of virtualized machines (VM) living on servers. The virtualized cloud DC traffic has a total new level of unpredictability, it's so dynamic in nature that the traditional solution of over-provisioning the network resources might not save you, especially at services peaks. In addition, the virtualized servers will always try to scale their resources up or down in response to changing loads, which includes spinning up or shutting down VM instances rapidly, where these instances not necessarily positioned on the same physical servers and they could exist at any server within the DC. Furthermore, in order to reduce the energy consumption, therefore smart solutions such as scaling down and reclaiming unnecessary resources are deployed to free underutilized servers by moving and consolidating VMs into fewer servers, then place free servers into a power save (sleep) mode, which will save a tremendous amount of energy, especially in a large DC. As a result, most of the cloud DC traffic is moving east and west between the servers internally, where VMs are moving and communicating across clusters, in addition a large amount of management system traffic will be carried on the same network fabric (in some designs). So, for the sake of the DC services availability, the network infrastructure should be extra over-provisioned and redundant. Overprovisioning and redundancy will result in higher costs of Opex and Capex. In addition, the network will be underutilized most of the time. Not forgetting to mention about the extra complexity of the network configuration and maintenance. [24]

To solve all above-mentioned issues related to the traditional network, a new network architecture is required. This led to the creation of a Software-defined network (SDN), which is the solution for all these issues, especially when applied in a virtualized cloud DC environment.

7.2 SDN architecture

SDN idea is based on three main principles. First principle is splitting the control plane from the data plane Second principle is to centralize network management instead of individual devices, in order to manage the entire network directly at once. The third principle is that the SDN architecture should provide open software programmable APIs between network applications and control plane. The control plan where the intelligence happening (e.g. routing protocols) will be a virtualized centralized application (a VM) called a “Controller” running on a server, this will leave only a network fabric consists of forwarding devices to perform fast forwarding of packets without any ability to make any routing decision. This architecture will reduce the software complexity, remove the unnecessary processing duplication in traditional network switches and will have the benefit of centralized network management. [24]

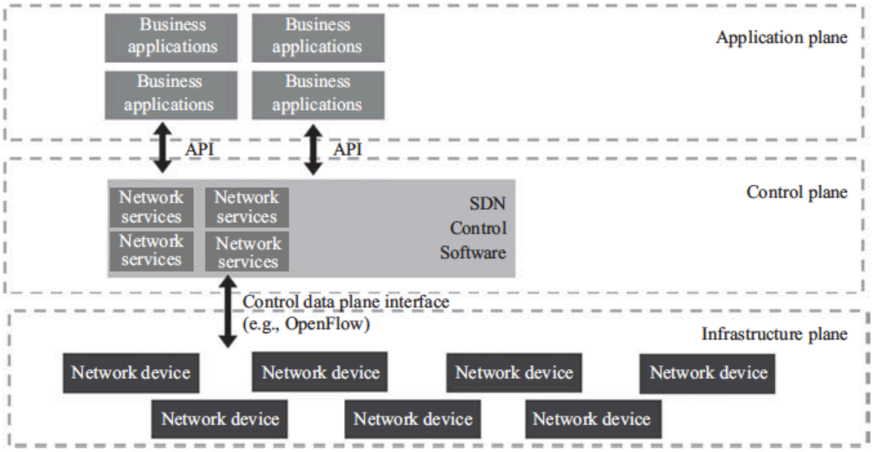


Figure 7-2 [24] SDN network architecture

Figure 7-2 shows an SDN network architecture. The Infrastructure plane includes network devices (hardware), they are acting as forwarding devices. The control plane or the controller which perform network services (e.g. routing protocols), also it provides API with applications in the upper layer to be able to interact and program the network. Application plane where the network applications will be running. The Interface between the controller and infrastructure plane is called a southbound, and between the applications and controller is called a northbound.

7.3 Why SDN?

SDN came to solve many issues as discussed in 7.1, the following are some of the main benefits of an SDN:

- More efficient and effective network management by having a centralized controller. Instead of configuring every single network device to introduce a new service, which could take several days and weeks in a big DC, with SDN it could be done in a matter of minutes, as the centralized controller are able to configure hundreds of devices at once.
- Remove unnecessary processing duplication at every network device, since route calculation happens only once at the controller, then the forwarding tables are populated down to each device's forwarding tables, this will eliminate the duplication in processing time. In addition, the convergence time and network availability is significantly improved, for example in case of a link failure, the time the network takes to calculate different path is tremendously small compared to the legacy network. The overall network performance is much higher.
- Network devices will become cheaper and much simpler by removing the control plane processing part. In addition, the ability of using white boxes to implement the network fabric provides more flexibility and royal free customized solutions that could be implemented and owned by the service providers.
- The ability to program forwarding devices opens an unlimited possibility for inventing new applications, solutions and services.
- Flexibility and granularity of network monitoring and control, for an example per client, per service…etc.
- SDN network could run multiple logical topologies simultaneously on the same physical fabric, where every logical topology's traffic is split from one another, also the topology could adjust its resources dynamically according to traffic load. In

addition, the ability to consolidate network traffic, where idle forwarding devices could be placed in a sleep mode to save power consumption and achieve a power aware network.

- New market of controllers and network application came to life with SDN.
- SDN technology perfectly solves the issue of managing an unpredictable cloud DC traffic, where it can rapidly adapt to the traffic change by scaling it's resource up or down as needed, this means a better network performance and utilization.
- Cloud DC are able to provide their clients with the ability to design their own network topologies (virtually) and assign resources as needed in an infrastructure as a service (IaaS) model. [24]

7.4 Network functions virtualization (NFV)

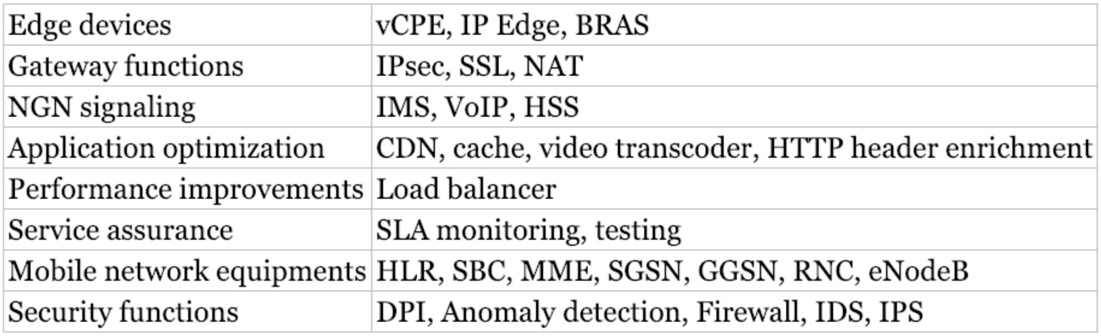
Typical network functions, such as firewalls and load balancers exist as vendor-proprietary hardware products. Figure 7-3 shows an example of a typical hardware firewall. They are complex expensive independent devices that have their own processing units and run their own pre-installed software stack, which is, when executed, provides a network function (NF). Similar to traditional switches, tradition network appliances are difficult to manage and expensive to maintain, also they have same underutilization issues linked with overprovisioning, especially when they are installed in big data centers. Furthermore, these devices are a vendor bind, thus do not provide APIs to integrate with a vendor-agnostic orchestration platforms. Besides the overprovisioning, flexibility, scalability design requirements that comes with a cloud DC makes the traditional hardware embedded NF model is not really efficient. [26]

Figure 7-3 Cisco ASA5540 adaptive security appliance (reference cisco.com)

Network functions virtualization (NFV) is the answer for the above issues, where the typical complex network functions will become a small a software (VMs) running on servers in a virtualized cloud. The well-known network functions such as authentication, normal routing, access control and QoS, are now running as VMs, this is where the name "network function virtualization" came from. NFV will bring all benefits of virtualization into the NFs such as scalability in response of load demand, high availability, and power-aware/efficient network. Expensive Independent hardware NF components and their associated running costs are therefore eliminated. Hence, NFV significantly reduces Capex and Opex, which made it the choice for many cloud DC owners.

Table 7-1, shows examples of VNFs that have replaced the traditional middleboxes and some of the critical devices in a mobile broadband core network. Furthermore, the introduction of micro services architecture, enabled a new modular VNFs architecture that allows a flexible construction of a variety of customized services that consists of several linked micro services which provides the customer with a unique service fits its specific need. [26]

Table 7-1 [26] Types of VNFs



Edge devices	vCPE, IP Edge, BRAS
Gateway functions	IPsec, SSL, NAT
NGN signaling	IMS, VoIP, HSS
Application optimization	CDN, cache, video transcoder, HTTP header enrichment
Performance improvements	Load balancer
Service assurance	SLA monitoring, testing
Mobile network equipments	HLR, SBC, MME, SGSN, GGSN, RNC, eNodeB
Security functions	DPI, Anomaly detection, Firewall, IDS, IPS

7.5 NFV architecture

Figure 7-4 shows an NFV reference architecture published by European Telecom Standards Institute (ETSI) consortium.

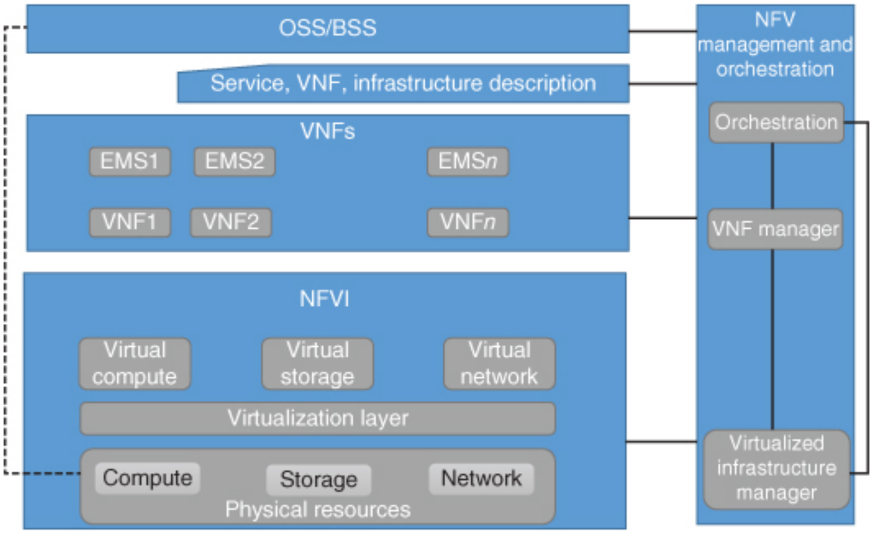


Figure 7-4 [26] ETSI NFV reference architecture

NFVI, is a technology platform that allow the execution of VNFs. The NFV orchestrator, consists of many distributed components, that handles network orchestration and NFV resources management. VFN manager, manages the VNF life cycle such as a VNF creation, resource allocation, migration and termination. The virtualized infrastructure manager, manages NFVI's three domains (computer and storage domain, hypervisor domain and the network infrastructure domain). [26]

7.6 Why NFV

The following are some of the benefits of implementing an NFV architecture:

- NVS (VM) is dynamically scalable according to the load requirement, this makes it fit for the cloud DC traffic nature.
- Reduced infrastructure costs, by eliminating expensive traditional NF hardware appliances and their associated management and maintenance costs.
- Reduction of power consumption costs, by employing scalability and smart power reduction techniques.
- NFV services is much faster to set up and deploy compared with the traditional NFs, which enables the operators to serve the market faster.
- Allows sharing of resources across services and different customers.
- NFV created a new services, where innovating new services has no boundaries only imagination of developers. [26]

7.7 NFV orchestration

NFV orchestration is an integral part and critical component of an NFV system, in other words you can't have an NFV implementation without orchestration. It manages every piece part of the NFV structure from networking resources to the compute and storage. ETSI management and network orchestration (MANO) working group has defined three key components for an NFV orchestration as shown in Figure 7-4, the following will explain the function of each component:

- NFV orchestrator, is responsible for the network service onboarding (installation) process. In other words, it manages the service entire life cycle (e.g. resource allocation, validation, authorization and terminating the service).
- VNF manager, operates on a lower level than NFV orchestrator, it manages the VNF instances life cycle (e.g. configuration, VM instance spin up, monitor each instance utilization and if needed scale it up or down, terminate instance and recall its resources).
- Virtualized infrastructure manager (VIM), it manages NFV infrastructure

components and provides the necessary resources for each individual NFV instance (e.g. CPU, servers, VMs, memories), it also controls the networking infrastructure (e.g. network devices, links status, routing, QoS...etc.), therefore the SDN controller is an integral component of the MANO. [26]

7.8 Summary

In this chapter, we started with discussing the traditional network performance issues, and architecture where each switch is an individual vendor proprietary that runs its own copy of TCP/IP protocol stack and makes its own routing decisions, also we explained why the traditional network design will not fit modern cloud DC east-west unpredictable and massive traffic. Then we explained how SDN architecture will fix the traditional network performance issues, and its suitability for DC networks. Then we explained the NFV architecture, its main components and benefits, then explained how it will reduce Opex and Capex while providing flexibility and scalability in modern cloud DC environment. [26]

Chapter 8 CORD

8.1 Introduction

Current cloud-based service model that lives in a remote data center and serves various devices such as a smartphone or a stationary PC, is widely accepted as the common cloud services architecture of today. However, the rapid growth of online services such as HD video on demand, online gaming and massive population of smartphones with their cloud-based apps, placed a tremendous load on current network infrastructure, according to the CTO of AT&T Labs Krish Prabhu, data traffic has grown 100,000% in the last eight years [27]. Despite the efforts spent by network owners to migrate their network infrastructure toward new broadband technologies such as NG-PON and LTE, the traffic will just keep growing. Furthermore, the near future technologies such as 8k-IPTV, augmented reality, real-time critical services, plus the upcoming tsunami of IoT data streams will congest the networks, creating a bottleneck. Another issue of the current cloud services architecture is the poor response time, even in an ideal non-congested network, the fact that the remote DC servers are geographically located far away from the end-user which creates a significant response delay, hence this architecture doesn’t meet the real-time requirements of the new futuristic services such as intelligent transport control system (ITS). Another issue of having a cloud centralized real-time processing and control which is used, for an example, in an IoT environment managed by a backend controller, is that it requires an Internet connectivity to be always available, which creates a single point of failure, and for similar systems that lack the minimum intelligence to self-function will be disabled until connectivity is recovered. [28]

8.2 Edge computing

To meet all these challenges of high bandwidth demand, poor response time and uncertain availability, while maintaining the advantages of agile, scalable and programmable virtualized cloud. Edge computing concept was created to fix these issues. The basic idea of the edge computing is to push part of backend applications and data processing away from the could (DC) to the network edges in order to be physically and geographically closer to end user (the data source). The term fog (first introduced by Cisco) refers to an edge computing that is associated with the IoT networks, which describes a new layer between the clouds (DC) and the ground (end user), but much closer to the ground. However, the term fog is now commonly used to describe all sorts of edge computing. Therefore, edge computing could be implemented by locating small servers (edge servers or nodes) positioned near the end users, and delegate the logic and give the control to these servers, Figure 8-1 shows an edge computing architecture. [28]

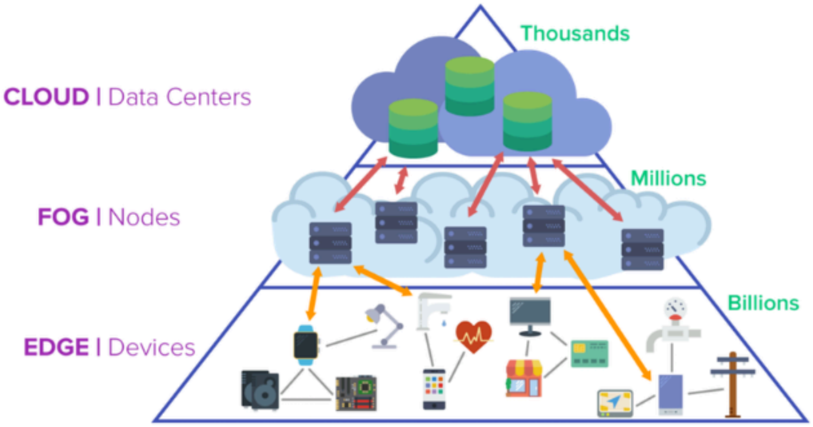


Figure 8-1 [29] Edge computing (FOG) architecture

The following are some benefits of the edge computing:

- provides a low latency high speed response that is required by the real-time applications, by becoming physically close from the end-users which will reduce the response time to few milliseconds compared to thousands of milliseconds of today's cloud apps. Therefore, this will improve QoS, and overall customer experience.
- Moving the big data processing to the edges, will direct a heavy traffic toward the edge servers instead of traversing all the way to/from the cloud core, which will relieve the back-haul links between the edges and the core network.
- The user devices will become cheaper by removing some of the intelligence to edges and make it shareable with other users. Therefore, applications will become less dependent on end-user hardware specifications.
- Services will become internet independent, for an example if the edge lost connectivity with the network core or the Internet, therefore end-user services won't be affected.
- Low latency response will enable innovation of new real-time applications.

Figure 8-2 shows some examples of applications that will use an edge computing, such as intelligent transport control system (ITS), augmented reality communication, M2M gaming, smart city solutions, medical monitoring solutions, smart home/buildings…etc. Most of real-time are interactive applications, they require a fast response time. In other words, the time it takes to collect the data, transmit it to an edge node, analyze it and return the control signal back to the end-user, should be less or equal to the human reaction time, and that could be achieved through an edge computing. Other services such as a web hosting, a cloud storage and emails will most likely continue to be served from the cloud. [28] [30]

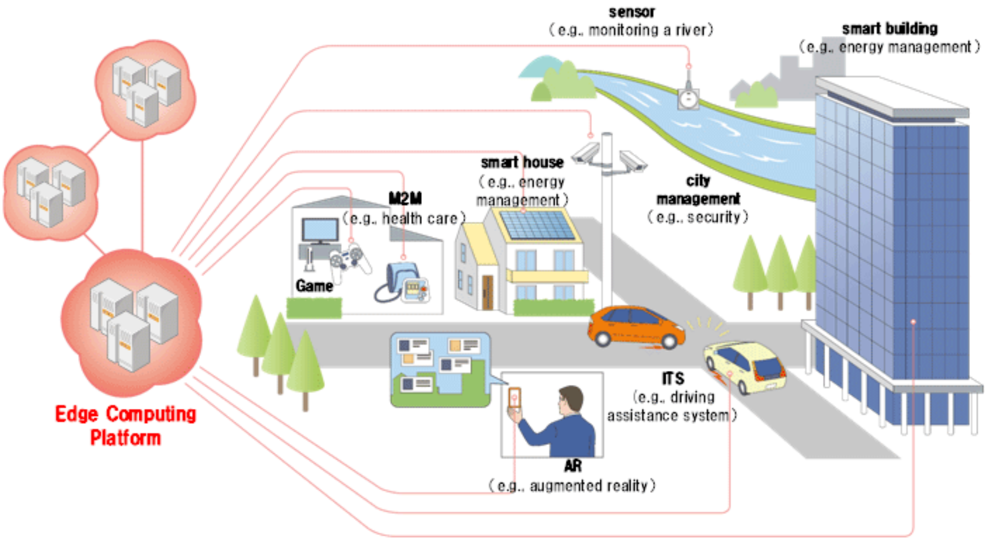


Figure 8-2 [28] Example of Application of edge computing

8.3 What is CORD?

Traditional legacy COs are constructed from vendor proprietary purpose-built devices owned by a Telecom operator, their architecture becomes a performance anchor and cannot fulfill today's extraordinary challenges, plus they are costly to manage and maintain.

On the other hand, network owners are attracted to the cloud benefits and wants to bring the cloud efficiency, agility, flexibility and programmability into their network infrastructure to be able to offer new user-centric services.

Central office re-architecture as a datacenter (CORD) is an Edge service delivery platform, a realization of the edge computing concept. It will transform the legacy COs into distributed virtualized small datacenters, by applying the following three principle technologies; SDN, NFV and virtualized Cloud. CORD platform is built with a commodity hardware (e.g. commodity servers and white-box switches) and an open-source software. Therefore, we could combine all the above sentences and define CORD as an edge service delivery platform that supports the current known services and the unknown future services, it supports different access devices and technologies and it has a programmable APIs, therefore, CORD enables datacenter economics into the CO.

Historically, CORD seen the light in 2015, it was developed by ON. Lab along with global community of partners and collaborators consists of technology vendors, network owners and individual volunteers, some of them can be viewed on the following link (https://opencord.org/members/).

CORD covers three service portfolios, Residential, Enterprise and mobile. This thesis will cover the residential track or (R-CORD), it will discuss its aspects and review an R-CORD demo installation.

To give an idea how big CORD platform's potential. Michael Howard, a co-founder of IHS markets, one of the industry's leading experts says in a study that a "seventy percent of operators worldwide are planning to deploy CORD". Another important declaration by Roz Roseboro, a Heavy Reading principal analyst, wrote that "Nearly forty percent of all end-customers will have service provided by COs or their equivalents using CORD by mid 2021". [31] [23]

8.4 CORD platform architecture for the network edge

CORD is an edge service delivery platform, focuses on the access edge. As shown in Figure 8-3, in a multi-tier cloud, CORD is positioned at the access network edge that connects end-users (e.g. FTTX or Mobile subscribers).

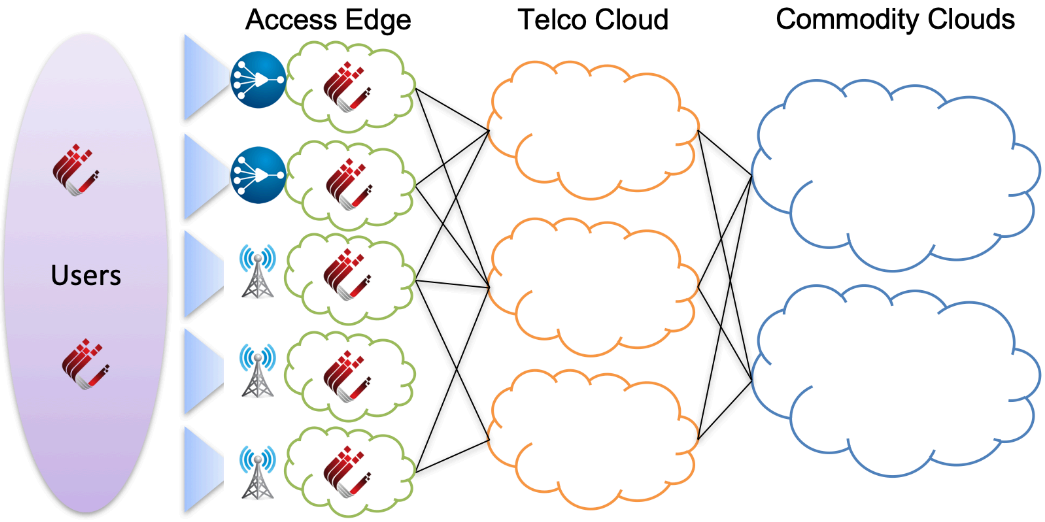


Figure 8-3 [32] CORD an edge service delivery platform

To understand CORD architecture, let's first discuss its architectural requirements, as per the following:

1) Uses commodity servers and white-box switches built with merchant silicon. Bare metal hardware will enable the installation of an open source software, also it's possible, for an example, to use a programmable hardware to provision new

services if needed. Plus, it's cheaper to acquire than vendor proprietary products.

2) Support different services, both legacy and yet to be created, where services disaggregated and deployed as a VNFs that live on commodity servers, and a switching fabric that's controlled by an open source SDN controller.
3) Use of an open source software and open standards.
4) The platform has to be extensible (i.e. adjustable performance and customizable services). For an example, you may scale hardware resources up or down as needed, which will help optimize the performance while smart utilize the resources, hence reduce costs and customize services per a subscriber.
5) The platform should be shared across multi-tenants and subscribers, where each tenant platform and every user service traffic is isolated.
6) Embraces robustness, scalability and automation of a cloud DC. E.g. uses same state-of-art cloud DC enabling technologies.

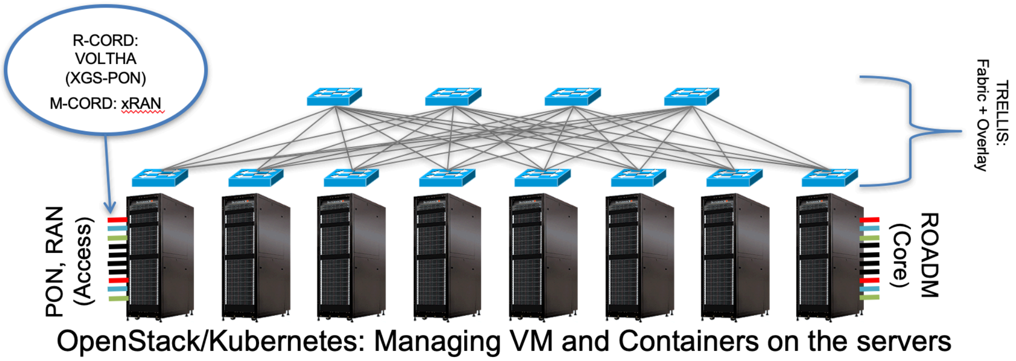


Figure 8-4 [32] CORD platform as a datacenter

As shown in Figure 8-4, the hardware consists of a switching fabric and commodity servers. The switching fabric consists of white boxes organized in a spine-leaf fabric, the leaves are used as a top of rack (TOR) switches that connect the servers (and VNFs), later we will talk about the switching fabric in more details. In CORD, there is no traditional north-south traffic of a data center, instead it's replaced with an east-west access network traffic (i.e. from the left), where the access network is connected to a commodity server (e.g. merchant silicon pizza boxes). Another important design aspect, is that the access network traffic might travel across the switching fabric to the upstream backbone (at the right side) through a disaggregated router without touching a single server (unless a service that lives on a server is required). [33] [32]

Figure 8-5, shows full CORD architecture from a software point of view. Again, all the software platform lives on a merchant built hardware (shown at the bottom) which consists of a commodity servers, fabric white-boxes and a stack of access network white-boxes. The middle layer contains the SDN controller of the switching fabric (ONOS), which is managing both the underlay fabric that consists of spine leaf white-boxes and the overlay tenant networks consists of virtual network (i.e. virtual switches), at ONOS northbound interface is where the network services run as fabric management and control applications (e.g. vRouter, AAA…etc.). Also in the same layer, virtualized network functions (VNFs) are running as a virtualized micro services in the form of containers and VMs, where OpenStack and/or Kubernetes (K8) are used for VM/container life cycle management, where OpenStack is responsible for instantiating containers (VNF) on the switching fabric in the form of control applications above ONOS, and the Kubernetes is responsible for instantiating containers (VNFs) lives on computing nodes, (Please note that in later versions of CORD OpenStack VM are replaced with K8 containers). [33] [32]

At the top of the architecture where the CORD controller, its main function is to provision services, where it enforces various policies to control what the services are permitted to do or what resources they are allowed to use. It also determines traffic paths through the switching fabric as per the provisioned services dependencies.

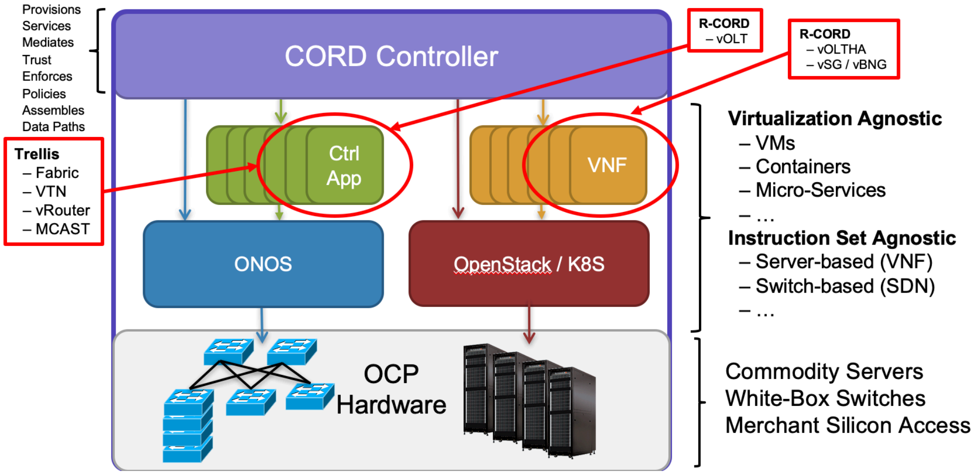


Figure 8-5 [32] CORD architecture

Looking at the big picture, CORD is applying same SDN concept of splitting control plane from the data plane but on the services architecture, as it ultimately splits the services control plane from the services data plane as shown in Figure 8-6.

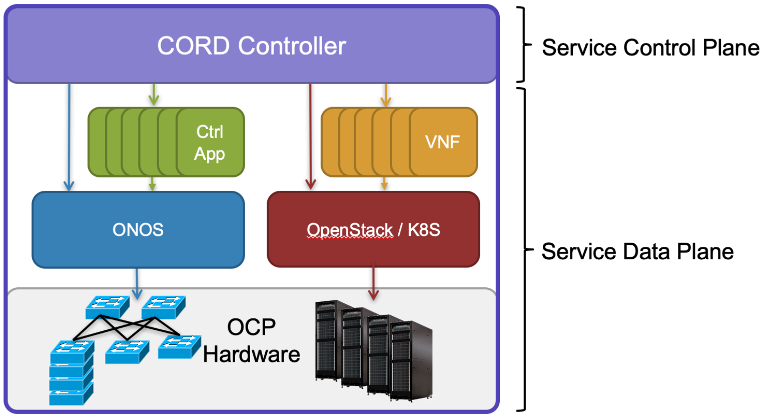


Figure 8-6 [32] CORD architecture, split service control plane from data plane

8.4.1 CORD controller (XOS) structure

CORD controller is called XOS, which stands for everything as a service (XaaS) Operating System (OS). XOS is constructed from a set of micro services conceptually organized in a three layers' diagram, as per Figure 8-7, the essential element of this structure is the XOS core inside the data model layer. To understand the rule of XOS, let's think about the structure of a tradition monolithic application running on a computer, for an example Microsoft Word, which consists of hundreds of embedded small functions, for convenience let's refer to them as services, these services will share the program memory space, therefore when you lunch Microsoft Word, you just run its executable file and the computer's OS will take care of running the whole application with all of its built-in services as a one-piece process. In CORD this is not the case, because all the services/functions of a monolithic application have been disaggregated into individual entities that could be initiated, executed independent from other services, where each service has its own sandbox. The micro services disaggregated architecture adds many benefits with some tradeoffs in complexity, where it's more complicated to combine and lunch several micro services in order to provide a "Service". Therefore, we will need a special tool (a software) to control the process of launching a group of interconnected micro services that will provide a "Service". XOS just does that, it's a self-contained software that takes data models as an input, these data models describe several micro services with their dependencies, and by running them in the XOS will create several interconnected micro services (i.e. VMs and containers connected with the network fabric).

XOS core has a generative-tool-chain called (xosgen), it takes an input of a set of predefined data models (or a service templates) along with a set of policies, then through running a descriptive language called "Xproto" (which is an extended version of Google's protobufs) inside the service templates, it automatically generates a code depends on the xproto translator tool, the generated code which is used to perform different tasks such as, implement interfaces, execute synchronizers, enforce security and initiate end-to-end tests over CORD in order to validate the system consistency. Xproto translating tool called “Jinja2”, it produces different codes depends on the Jinja2 template, more info about Jinja2 is available on this link (http://jinja.pocoo.org/docs/2.10/), for an example using “Django.xtarget” template will generate a Django model (a python code) which will be executed by the synchronizers, more templates are available on this link (https://github.com/opencord/xos/tree/master/lib/xos-genx/xosgenx/targets). [33] [32]

Policies are applied on the data models before they get translated into a code, therefore when the model is finally executed and becomes a group of instances, then the policies are already applied on the whole CORD cloud. (I.e. applying the policy directly on the data model, will ultimately, apply it on the generated services).

XOS has a database storage to make it persistence, also there is an event bus to capture XOS event messages in order for the other listening components to process the XOS events. Overall, XOS core function is to implement data models which are a form of relational representation of services (CORD uses TOSCA models), into a cloud topology consists of VNF instances.

At the top layer, or user interfaces layer, there are the following interfaces:

- Tosca API: TOSCA is a modeling language for NFV service models. It refers to Topology and Orchestration Specification for Cloud Applications. It basically describes cloud topologies, where every single blueprint in TOSCA represent a service template which turns into a service instance when instantiated by an orchestrator, in this case CORD orchestrator. By using TOSCA service templates, Tosca API interface helps automate the configuration of applications (VNFs) in CORD where the workflow “recipes” are supplied to CORD controller in order to form the services .[34] , more info about TOSCA on the following link, (https://www.oasis-open.org/committees/tosca/faq.php)
- REST API (Chameleon): is used to implement CORD graphical user interface.
- VIM, reserved for future implementation to offer infrastructure as a service (IaaS).

The synchronizers layer synchronizes backend services with the controller, in other words, it synchronizes ONOS, Control applications on top of ONOS, OpenStack and/or K8 and the (VNFs) images implemented in containers or VMs on top of OpenStack and K8 as shown in Figure 8-6. In other words, synchronizers are used to synchronize the services consistency to reflect database state (i.e. service model state). [33] [32]

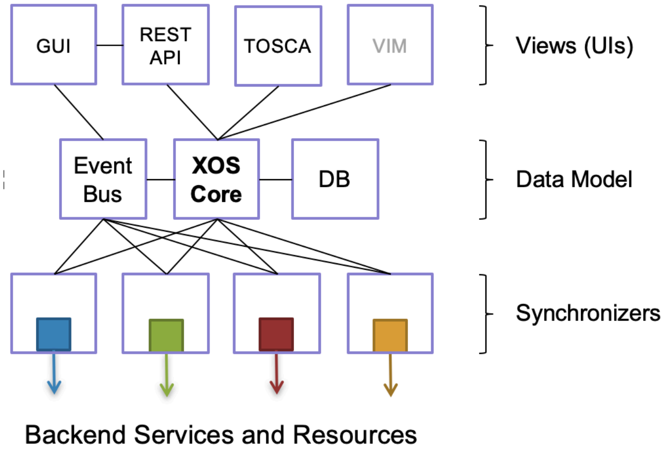


Figure 8-7 [32] Cord Controller XOS structure

8.4.2 CORD models and frameworks

So far, in 8.4.1, we discussed XOS tool mechanism that evaluates models and apply policies to those models to generate different kinds of code depending on a targeted XOS component.

In this part, we will discuss the modeling concept of services in CORD, which is the essence of how CORD provides services innovatively and completely different than any traditional CO implementation. [33] [32]

CORD consists of set of core models, in other words, when these core models are combined together, then they become the CORD itself. Pretty confusing right!

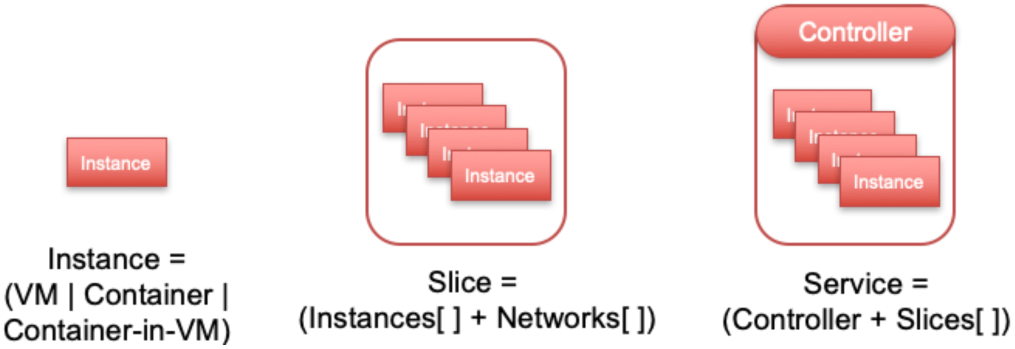

Instance
Instance = (VM | Container | Container-in-VM)
Instance
Slice = (Instances[ ] + Networks[ ])
Controller
Instance
Service = (Controller + Slices[ ])

Figure 8-8 [32] core models

Let's take it one step at a time. The first model is the basic building block of a service, which is either an instance or a network, these instances could be containers, VMs or containers running inside VMs. Let's go one step further to the next model, which is a combination of instances and networks or what we can refer to as a slice, not to be confusing, the slice is referred to as a virtualization agnostic infrastructure model, but for now let's stick with only a "slice". Please also note that, until this point we still don't have a "Service". Now if we consider a group of slices controlled, configure and manage by a controller, we will end up with a "Service", one example of a service is a vRouter or vSG, where each one of these services is a group of different instances managed by a controller. As described in Figure 8-8. [33] [32]

Our next model is referred to as "instruction set architecture agnostic service graph", for convenience let's call it only a "service graph". Service graph, is defined as a set of services logically linked together with a set of dependencies. Figure 8-9 shows an example of Residential CORD (R-CORD) simple (but not full) service graph, which consists of three services, they are; virtual OLT (vOLT) which implemented as a distributed flow rules in the underlying hardware (i.e. pizza boxes and switching fabric), later we will explain it in more details. A virtual subscriber gateway (vSG) service, which is implemented by a set of VMs and containers, last service is the virtual router (vRouter) which will be discussed later under the switching fabric. [33] [32]

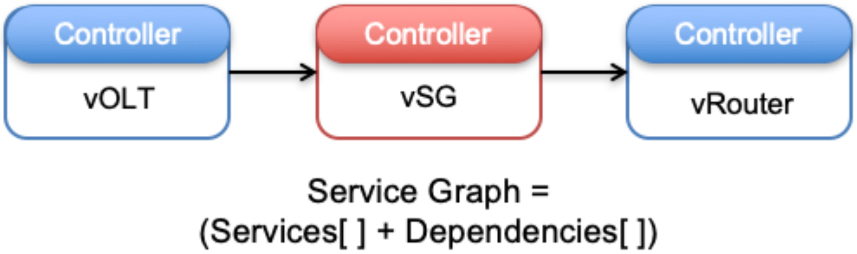


Figure 8-9 [32] R-CORD Service graph

Therefore, to create a service graph in CORD that consists of services X, Y and Z, first we need to define each service's model (i.e. service templates we discussed in 9.4.1) and define its dependencies with other services (how they relate to each other), regardless of their implementation methodology, when executed by the XOS they became implemented services. [33] [32]

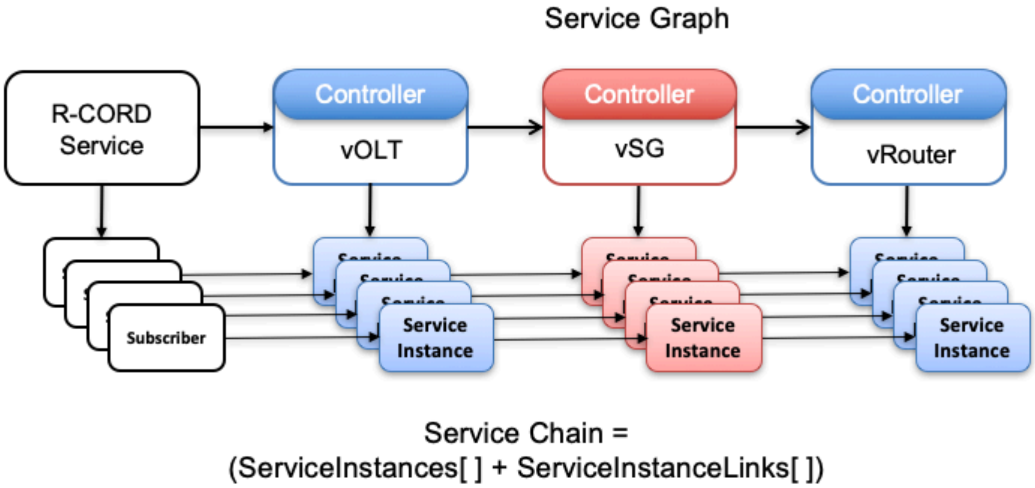


Figure 8-10 [32] R-CORD Service chain

Now let's glue all these concepts together. First, the telecom operator defines the service track, each service track should include a pre-defined service graphs (service models + dependencies). In CORD, there are predefined service tracks, Residential CORD, Mobile CORD and Enterprise CORD, built with collaboration and contribution from Telecom providers in order to provide a real use cases. For an example, for the R-CORD track, after the service graph of R-CORD is on boarded into CORD platform, when a new subscriber connects to the R-CORD platform, he or she will be assigned a logical subscriber object, which gets connected to an instance of vOLT, a vSG instance and finally a vRouter instance. Same steps apply either for a single or a thousand subscriber. This process is called a "service chain" and it's shown in Figure 8-10. [33] [32]

In another different scenario, a group of services could be nested to form one service, in other words this service will contain a service graph that consists of multiple services. Figure 8-11 shows an example of EPC-as-a-service, in this case it consists of group of nested services that could be included as a part of a larger service graph that could have multiple instances. [33] [32]

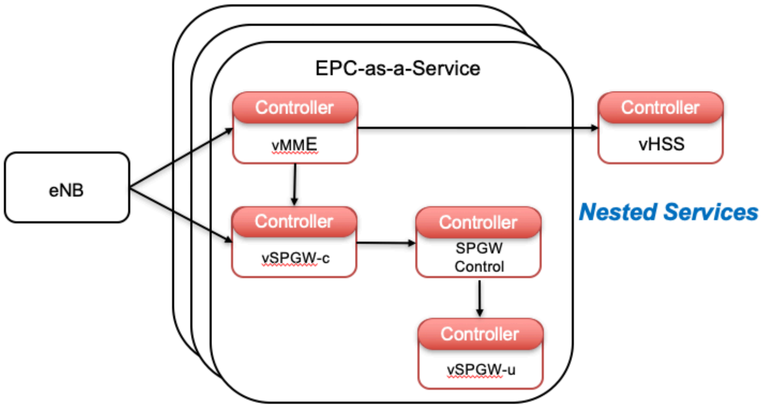


Figure 8-11 [32] Example of a Nested Services

You might ask, what the Telecom provider will achieve by implementing the CORD platform besides having it built from commodity hardware and disaggregated software services. What are the benefits that the service provider will get?

I will save the full answer to the end of this report, however the quick answer is that the CORD platform will offer Telecom service providers, the advantage of providing verity of old traditional and new services that could be built by mixing different combination of building blocks (services), since CORD modeling mechanism enables tunable sizing and mixture of hardware and software components to be implemented through new service graphs over the CORD platform.

Now let's compare that with an old proprietary hardware platform, how many different combination of services the Telecom provider could build out of it? (Considering it's practically doable in the first place), is it scalable and how fast it could be modified? [33] [32]

8.4.3 CORD service portfolio

CORD platform has a quite large service portfolio as per Table 8-1, new services expected to be added with new releases.

Table 8-1 [32] CORD V4.1, Service Portfolio

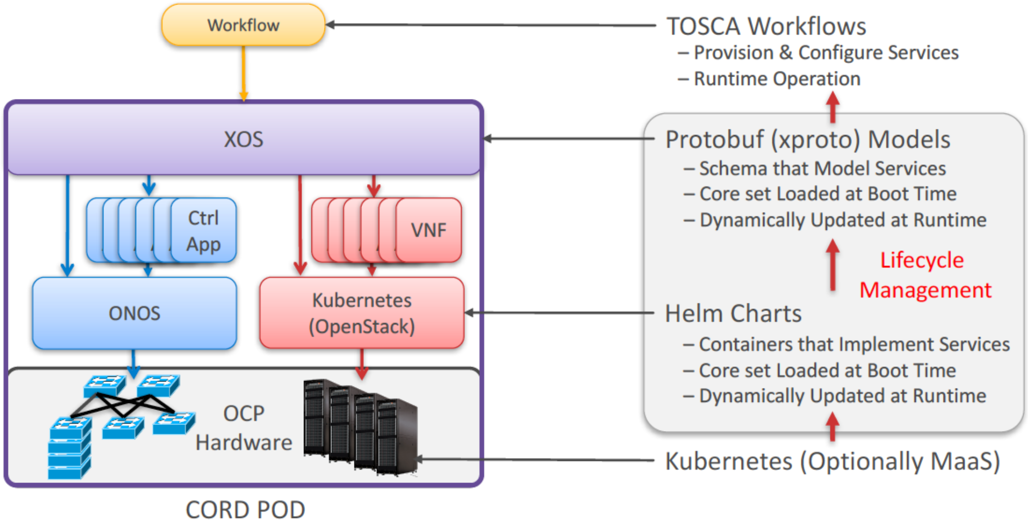
Service	Description
vSG	Virtual Subscriber Gateway
vOLT	Virtual OLT
vRouter	Virtual Router
vEG	Virtual Enterprise Gateway
vEE	Virtual Enterprise Ethernet
vHSS	Virtual Home Subscriber Server
vMME	Virtual Mobility Management Entity
vEPC	Virtual Evolved Packet Core
vTR	Virtual Truck Roll
HyperCache	Akamai CDN
SGW	Virtual Serving Gateway (User)
vSGWc	Virtual Serving Gateway (Control)
vPGW	Virtual Packet Gateway (User)
vPGWc	Virtual Packet Gateway (Control)
vBBU	Virtual Broadband Base Unit
xRAN	Virtual Radio Access Network
ONOS	Network OS
OpenStack	Infrastructure-as-a-Service
Swarm	Container Management Service
Fabric	Fabric Management Service
VTN	Virtual Tenant Network
A-CORD	Monitoring-as-a-Service
LBaaS	Load Balancer-as-a-Service
VNaaS	Virtual Network-as-a-Service
AAA	Access Control
Address Manager	Allocate IP Addresses
IGMP	Multicast Signaling
MCAST	Multicast

8.5 CORD automated configuration

To facilitate automation, CORD uses different tools and models as described in Figure 8-12, let's consider starting from a bare metal hardware with nothing installed on it, as shown in the bottom, first we will use MaaS (https://maas.io/) to mount CORD core set at boot time, which includes Kubernetes (k8), then K8 will spin up the CORD controller XOS, its interfaces and ONOS. At this point, the POD is up and running and ready to implement the services.

To implement the service graph, such as R-CORD, we begin by giving CORD a workflow through TOSCA API, inside the workflow file a provision (resources allocation), configuration and dependencies of services, which could be reconfigured at any time during run time by supplying a new workflow recipes or you can do it per a single service model at a time.

So, TOSCA workflow tells the CORD controller what services it should implement via passing services templates in the form of data models programmed in Xproto. In other words, XOS will understand what services to create, how they relate to each other through Xproto. Then Kubernetes (k8), which is an orchestration tool for Docker containers, the K8 will use "Helm charts" to spin up containers images in order to instantiate services. In other words, the k8 will containers to implement services instances which were modeled using Xproto that configured into CORD through TOSCA API.



Workflow
TOSCA Workflows
– Provision & Configure Services
– Runtime Operation
XOS
Ctrl App
VNF
ONOS
Kubernetes (OpenStack)
OCP Hardware
CORD POD
Protobuf (xproto) Models
– Schema that Model Services
– Core set Loaded at Boot Time
– Dynamically Updated at Runtime
Lifecycle Management
Helm Charts
– Containers that Implement Services
– Core set Loaded at Boot Time
– Dynamically Updated at Runtime
Kubernetes (Optionally MaaS)

Figure 8-12 Automated configuration process in CORD

8.6 Trellis the switching fabric

Trellis is the name used for the switching fabric in CORD, it is defined by CORD as a multipurpose leaf-spine fabric designed for NFV. It's implemented based on SDN and NFV on a leaf-spine architecture. Trellis is built from an open source software installed on commodity servers, and bare metal white boxes made from a merchant silicon and controlled by an open source SDN controller which manages both, the underlay fabric and the overlay virtual networks. Figure 8-13 shows Trellis architecture. ONOS and VNF applications are running on the commodity servers (head and compute nodes).[35]

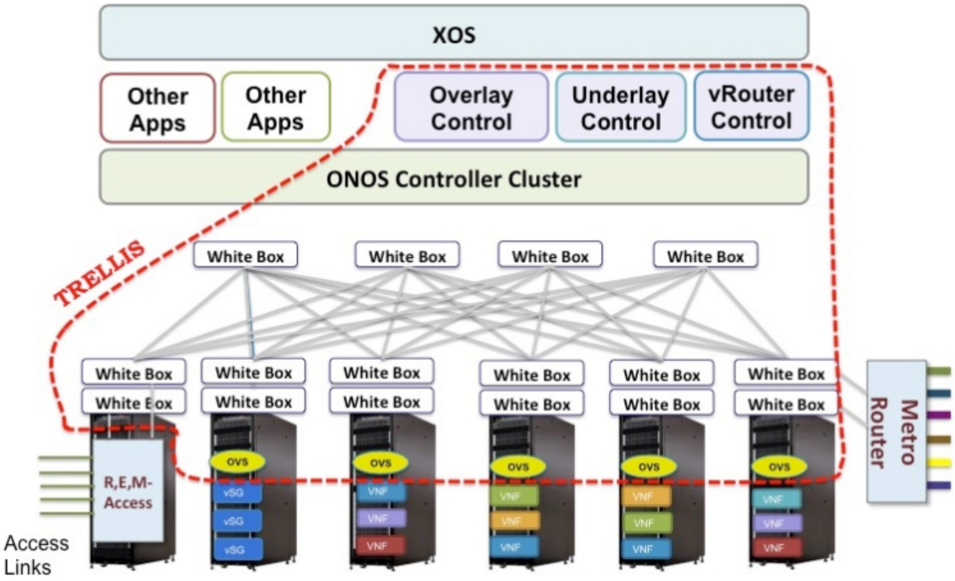


Figure 8-13 [35] Trellis Architecture within CORD

8.6.1 SDN controller (ONOS)

Open Network Operating System (ONOS) used as a unified SDN controller in Trellis to manage and control the underlay (leaf-spine switching fabric) and the virtual network overlay, as shown in Figure 8-14. Therefore, fabric configuration, traffic control, policy implementations will be managed through ONOS.

Having a unified SDN controller for both underlay and overlay networks simplifies the implementation of distributed virtual routing for tenant networks and facilitate optimized delivery of multicast traffic streams.

In current CORD release, ONOS is split into two clusters to achieve better isolation between components responsibilities and eases development process. There are two clusters of ONOS controllers as per the following:

- Onos-cord, it manages the overlay network and service components. Also, it hosts some control applications such as a virtual OLT (vOLT).
- onos-fabric, it controls the switching fabric and interfaces with the upstream router that routes CO traffic to external networks. Also, it hosts the Fabric control and virtual router control application (vRouter) VNFs.

In Trellis, some NF services were disaggregated into smaller components that have been distributed over Trellis, for an example, in Trellis the multicast service is provided by two applications, IGMP which runs on onos-cord and PIM-SSM which runs on onos-fabric. [35]

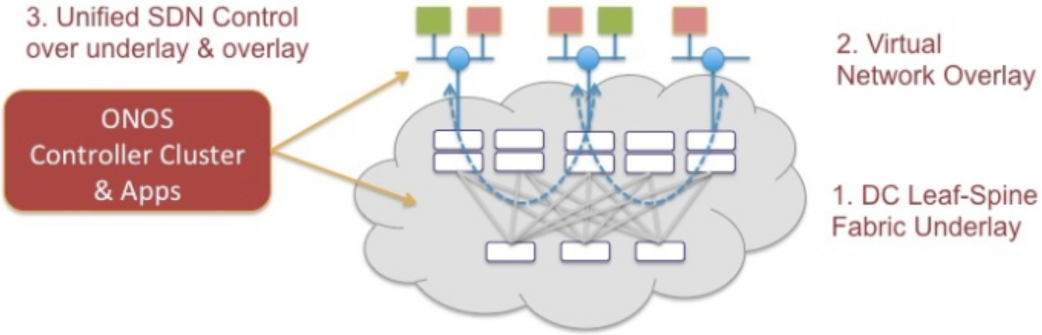


Figure 8-14 [35] ONOS controls overlay and underlay network

More details about ONOS are provided in developer's homepage on the following link (https://wiki.onosproject.org/).

8.6.2 Trellis underlay fabric

Trellis underlay fabric is a hardware bare metal white box switches controlled by an application named (segment-routing) inside the SDN controller (ONOS) and works with other VNFs such as vOLT and vRouter to implement the CORD networking services.

As shown in Figure 8-15, the white boxes are organized in a spine-leaf architecture, where spines at the top and leaves at the bottom. The leaf switch act as the TOR switch that is directly connected with the computing nodes. All the fabric is controlled by a cluster of multiples instances of ONOS that share the load. OpenFlow protocol is utilized to provide the communication channel between the control plan (ONOS) and the data plane (fabric). All network protocols are exclusively run in the control plane (ONOS and control applications) such as ARP, MAC, VLAN, MPLS, VXLAN, STP, MSTP, PVSTP, OSPF, IS-IS, LDP and BGP.

Each rack is placed in one network domain, therefore computing nodes (i.e. pizza boxes) on the same rack are assigned IP address from the same network subnet, thus traffic between nodes on the same rack uses L2 Ethernet switching provided by leaf switch acting as a TOR, however VLANS could be configured to split traffic if needed. On the other hand, Internal traffic across racks will use L3 routing with MPLS (e.g. ECMP hashing and MPLS). Therefore, nodes from different racks will use fast MPLS label switching between leaves and spines, thus leaves and spines will behave as an IP/MPLS network, where MPLS label is pushed and popped at leaves (MPLS network's ingress and egress), spine switches forward packets between leaves solely based on labels switching. The separation of forwarding between spines and leaves, where the spines do only MPLS fast label switching, is perfectly suits multi-purpose fabric concept, where the fabric will be able to serve different types of traffic (PON, Mobile or Enterprise) very efficient, where the spines (the network core) will perform the same traffic forwarding based on MPLS for all use-cases regardless of traffic source. In addition, this also benefits hardware optimization, for an example the spine white boxes that perform only MPLS label switching could have more switching processing power where leaf switches that handle larger tables of IP rules could include more forwarding table capacity, more QoS capability and less processing power than spines. [36] [37]

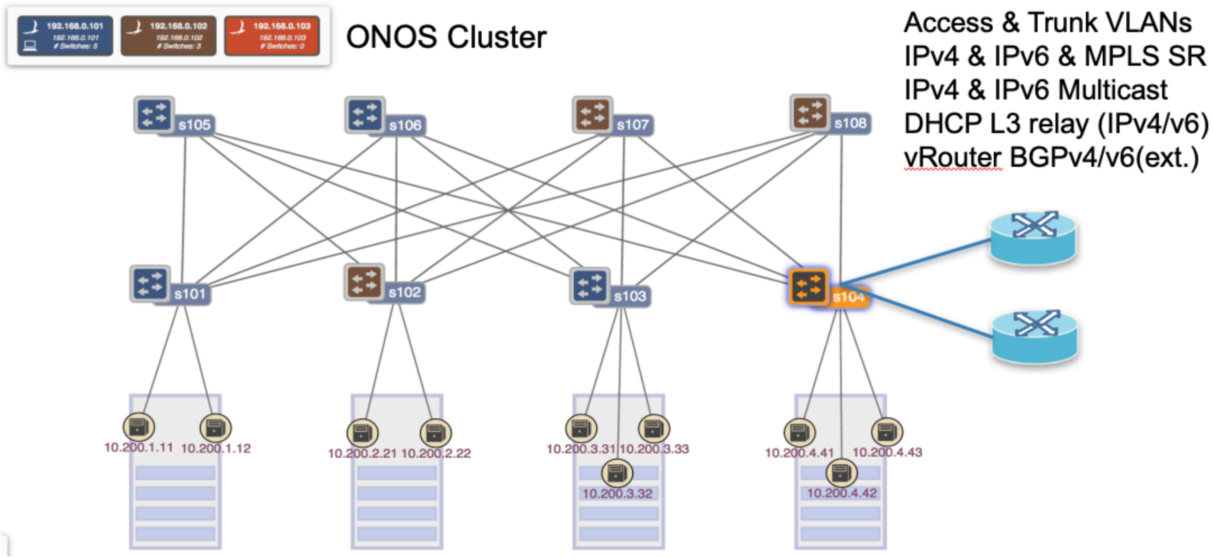


Figure 8-15 [37] CORD Switching fabric (Trellis), ONOS GUIi view

Multicast (IPv4 & IPv6) is also supported, where ONOS could configure replication at spines and leaves level to form multicast trees, this is particularly important for IPTV streaming from an upstream (source) to a residential subscriber (destination). In Trellis, the underlay fabric transports VXLAN encapsulated traffic between services. Full list of services that Trellis fabric supports can be found on (https://wiki.opencord.org/).

The separation between overlay virtual network and underlay hardware fabric is important, it keeps the underlay hardware simple and multi-purpose, and it doesn't have to deal with implementing complex virtual networks. The overlay network is realized through software switches applications such as (VoS). Furthermore, having a unified SDN controller that manages both the underlay and overlay networks, enables them to interact with each other. [36] [37]

8.6.2.1 White box build

White boxes are made from a merchant silicon, even though it's described as a bare metal which indicate that they don't run any kind of logic, however this is not entirely true, since a small software stack is required to perform the forwarding plane interaction with the SDN controller.

First let's examine the hardware, as explained earlier there could be quite different hardware designs for the white boxes depends on the network design requirements, but the general rule is that any hardware is acceptable as long as it can support the switch software stack. To demonstrate that we will consider using the same white box (Accton 6712) for the spine and the leaf, Accton 6712 switch is based on Open Compute Project (OCP) certified design as shown in Figure 8-16, it has 32 40Gig ports, depends on the usage, the leaf will use 24 ports to connect servers, access network pizza boxes and metro router, the remaining 8 ports will be used to connect with the spine white boxes. On the other hand, the spine switches will only connect with leaves. Different switches with a different port capacity could be used, there is no fixed hardware as long as it runs the switch software stack. [36] [37]

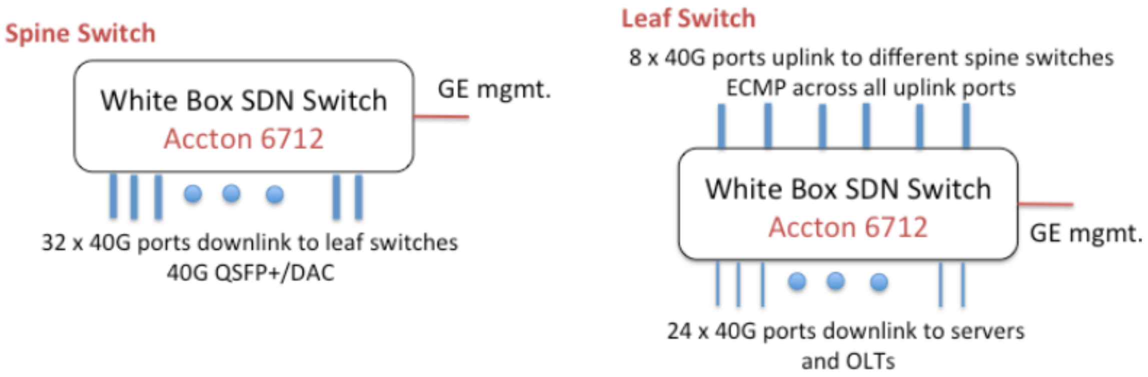
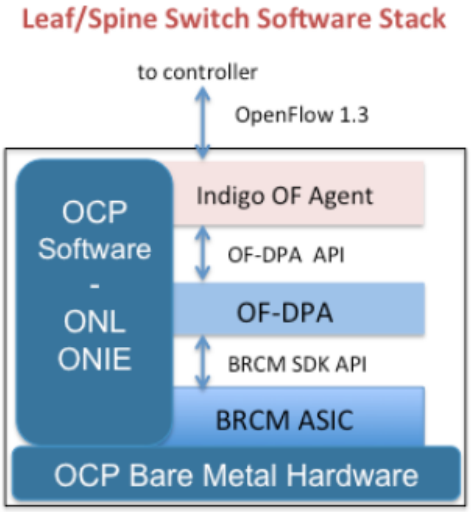

Spine Switch
White Box SDN Switch
Accton 6712
GE mgmt.
32 x 40G ports downlink to leaf switches
40G QSFP+/DAC
Leaf Switch
8 x 40G ports uplink to different spine switches
ECMP across all uplink ports
White Box SDN Switch
Accton 6712
GE mgmt.
24 x 40G ports downlink to servers
and OLTs

Figure 8-16 [36] White boxes hardware

The spine and leaf runs an identical software stack, shown in Figure 8-17. Open network Linux (ONL) and Open Linux install environment (ONIE) is used as the switching operating system. Also because of the use of Broadcom ASIC as the merchant silicon to build the white box, hence software stack uses Broadcom's Open flow data plane abstraction (OF-DPA) and Broadcom SDK API to communicate between an OpenFlow API agent (Indigo-OF agent is used in this example) and the Broadcom ASIC chip. By using OpenFlow, the software stack enables ONOS to program the ASIC chip's forwarding tables that belong to any switch on the fabric. [36] [37]



Leaf/Spine Switch Software Stack

OCP: Open Compute Project
ONL: Open Network Linux
ONIE: Open Network Install Environment
BRCM: Broadcom Merchant Silicon ASICs
OF-DPA: OpenFlow Datapath Abstraction

Figure 8-17 [36] leaf/spine Software stack

8.6.3 Trellis network overlay

The network overlay, in simple words is a logical virtual network of switches that connects service instances in a service chain, these service instances could be a VM or a container and could exist in any computing nodes, on same rack or different racks. The traffic between services is encapsulated using VxLAN where VxLAN tunnels logically connects service instances.

The virtual network traffic is carried by the physical underlay fabric, in order to understand how that happen, let's review it in steps. First, CORD will instantiate the service graph through XOS, then service instances (in a service chain) are created for the specific subscriber including the virtual network that connects these services, where the service graph's virtual network is translated by a Virtual Tenant Network (VTN) inside ONOS into flow rules entries, these flow rule entries are pushed down to the underlay fabric through an OpenFlow channel into every switch's ASIIC chip forwarding table. This means if the service instances where on the same rack then their traffic will be carried using Ethernet switching where MAC addresses of the computing nodes NIC will be used to perform L2 switching. Another example if two service instances belongs to the same service chain wants to communicate, but the two instances were living in different computing nodes on different racks, in this case their traffic will be carried by an IP/MPLS leaf spine fabric. The third example, if the service's traffic wants to access the Internet, then it has to go through the virtual subscriber gateway (vSG) then to the virtual router (vRouter) which acts as BGP router, then to the internet. Figure 8-18 describes this process. Please note that, in all the three scenarios, the traffic is always logically tagged with VxLAN regardless of where the service instance lives on servers.

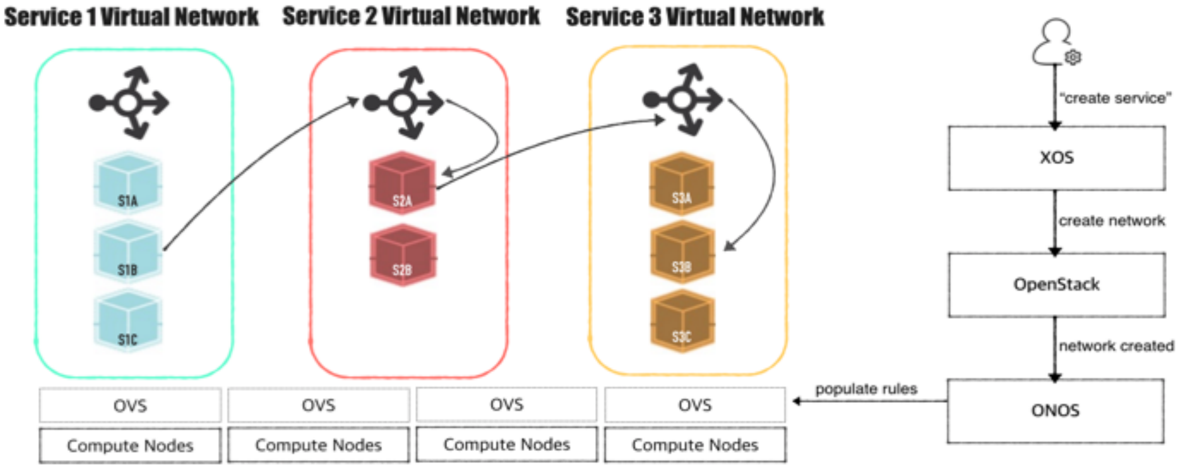


Figure 8-18 [38] Services connected together with a virtual overlay network to form a service chain

Overall, the beauty of splitting the logical overlay from the underlay networks, is that each one of them will communicate independently with no awareness of the other fabric. What makes it even more compelling is that the multiple overlay logical networks that connects many service chains belongs to different subscribers and handles different types of traffic from a different use-case (mobile, FTTH or enterprise) that belongs to different tenant are sharing the same physical underlay network fabric, where the underlay fabric (forwarding plane) is not aware of the overlay network existence and it only sees and gets flow rules that are pushed down by the SDN controller. [38] [37]

The overlay network is crucial for CORD scalability, because when the service instances are dynamically scaled up or down, the associated virtual network should follow. In addition, every virtual switch (OvS) within a service chain will have a load balancer, in order to balance the load between services instances within a service chain, consequently balancing the load on the underlay fabric. [38] [37]

8.6.4 vRouter service

vRouter service in CORD replaces the broadband network gateway (BNG) that exists in the traditional CO networking infrastructure. The vRouter service runs above ONOS, it uses Trellis fabric to provide the upstream communication and Internet access to CORD subscribers and services.

As a gateway for the CORD platform, the vRouter is required to provide an IP routing between CORD and the external metro router, therefore it has to support IGP and BGP routing protocols stack, also it should support Multicast protocols such as PIM-SSM.

The vRouter service architecture includes a control plane and a data plane. The data plane will consist of Trellis fabric (both overlay and underlay). The control plane architecture will consist of a 3rd party protocol stack software (Quagga), note that any other 3rd party software that implements routing stack could be used here not necessary Quagga. The vRouter app and ONOS diagram is shown in Figure 8-19. [39] [37]

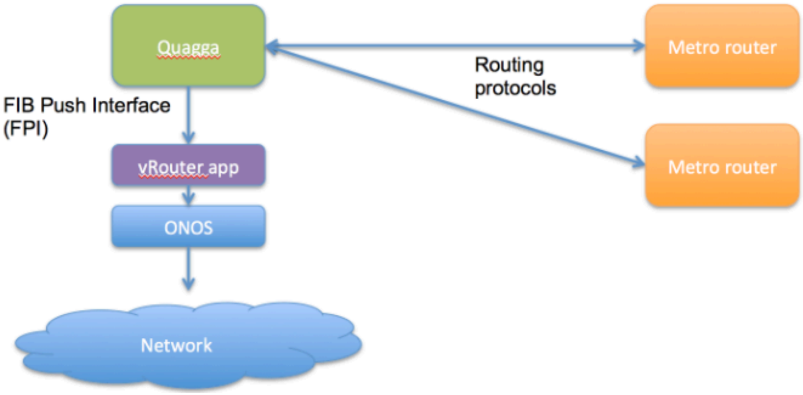


Figure 8-19 [39] Control Plane of vRouter service

Let’s discuss the vRouter service in action as shown in Figure 8-20. The control plane uses (Quagga) to implement the routing protocol stack (e.g. OSPF, IS-IS, BGP...etc.). Therefore, Quagga will talk with external upstream metro router by exchanging compatible routing protocol control messages (in this example OSPF, iBGP), *shown in Blue dashed line*, through a dedicated port on the leaf switch is configured by the vRouter to forward the routing protocol traffic to Quagga on the control plane. Quagga will use a “Forwarding Information Base Push Interface” (FPI) API to send all the external IP routes received from the upstream to the vRouter application, *shown in purple dashed line*, then the vRouter app that acts as the forwarding plane manager, through ONOS and OpenFlow, will configure the Trellis fabric to forward the traffic correctly to the outside world, and it does so by configuring the edge leaves that have a direct links with the upstream metro routers by pushing down forwarding tables contains the external IP addresses, *shown in red dashed line*, whereas rest of the fabric (spines and leaves) will be configured with a default route toward the edge leaves in order to forward their traffic to the upstream/Internet. In addition to the underlay fabric, ONOS will also configure the overlay’s OvS switches to correctly route traffic if, for an example, a service wants to connect to Internet, *shown in dashed green line*. [39] [37]

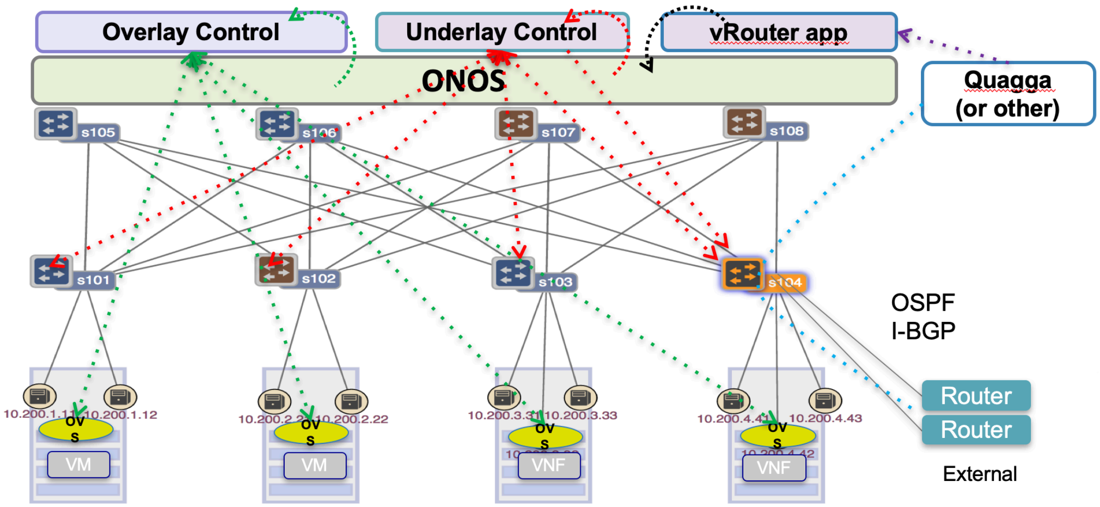


Figure 8-20 [40] vRouter service in operation.

8.6.5 Redundancy in Trellis

Redundancy in Trellis is applied on both the overlay and the underlay fabric, to avoid a single point of failure, thus provides a high service availability.

The control plane redundancy in Trellis is achieved through the same way that the scalability is achieved, as in Figure 8-21 (A), ONOS will always run multiple instances that are spread over compute nodes for the reason of scalability and also redundancy, where every switch in the fabric is connected to multiple instances of ONOS, however only one instance will act as the master controller for that specific switch at a time, and the other instances act as backup controllers. In the case of losing one of the master ONOS instances, as shown in Figure 8-21 (B), then one of the backup ONOS instances will take over and become a master controller, therefore services on the fabric won't be affected by the failure, in the meanwhile the system will recover the failed ONOS instance. [40] [37]

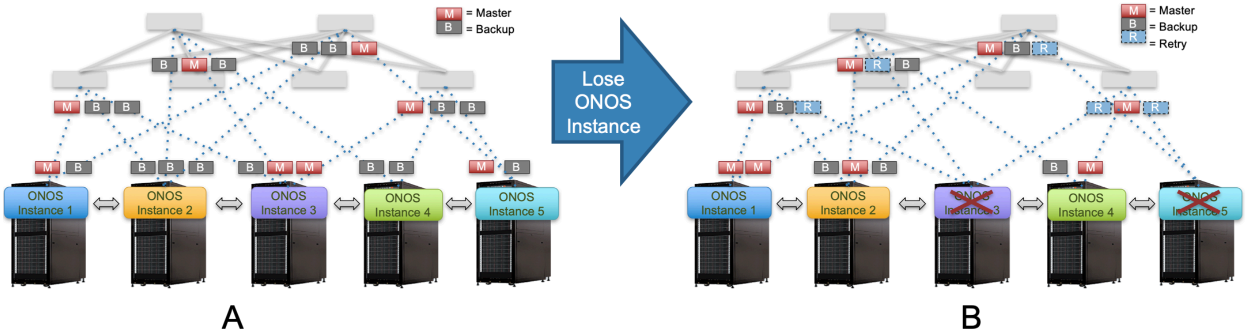


Figure 8-21 [40] Control Plane redundancy

Redundancy is also implemented in the underlay fabric as shown in Figure 8-22, there are multiple connections between the leaf and spine switches, and also there are two TOR switches on each POD rack. In addition, the edge leaf also has redundant multiple links to the upstream metro routers. The access network will have redundant OLTs implementation (active, standby) which is still under development.

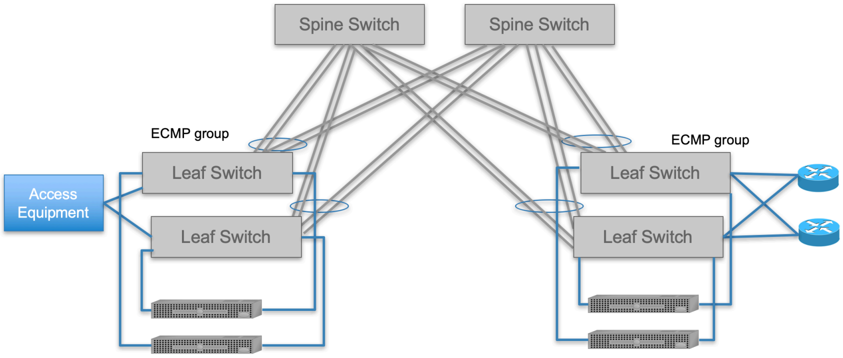


Figure 8-22 [40] Redundancy in underlay fabric

8.7 Residential CORD (R-CORD)

The residential CORD track, is designed as a built-in CORD solution for the FTTH access network, currently it's available for G-PON and XGS-PON access networks (as a trial), but it still growing to include other PON technologies such as 10G-EPON, and NGPON2. Furthermore, there is one big advantage that the Telecom companies have which most of cloud service providers lack, which is that the Telecom companies already own existing widely distributed COs that are fairly close from the end-user, which naturally give the Telecom providers an extra incentive toward implementing the new Edge computing technologies such as R-CORD.

As mentioned in the CORD definition, it is a service delivery platform aims to bring the agility, efficiency, and economies of the cloud into Telecom provider's central office. Well, implementing R-CORD will just do that, it will bring the cloud benefits and business model to the CO. Not only that, but because the CORD platform is shared between access networks, it could serve residential, mobile and enterprise customers on the same platform.

To understand the advantage that R-CORD has over the traditional COs infrastructure, let's take a closer look on the traditional CO architecture, for an example in order for the traditional CO to be able to serve mobile and residential subscribers, then it has to have two different infrastructures to support each use case, including a different switching fabric for both, plus a separate proprietary hardware to support the network functions. In addition, it is a very expensive and time consuming to manage and operate these heterogeneous infrastructures. No need to mention, if you want to scale up your resources and/or add a new service, then you will have to purchase a new hardware and get a new software license then reconfigure the network which takes a significant amount of time and money to do. Let's compare that with R-CORD, first you will have a one common platform to serve all your FTTX, Mobile and Enterprise customers, the only hardware difference is in the last mile access equipment, which is also an affordable cheap white box. Also, it's far easier to deploy new services in R-CORD, where all you need to do is to define a new service graph, and run it through the CORD controller, Bang! You have a new service, no new hardware or a major reconfiguration is needed, that might affect the service availability. Furthermore, in R-CORD has the advantage of using Trellis switching fabric with all discussed advantages over traditional old network fabric. Another advantage is that R-CORD hardware is much cheaper to acquire, manage, and to reprogram. The list goes on and on, R-CORD by far is the winner, it will significantly reduce Capex and Opex plus enable new service to be provided. Also, the fast service deployment and higher availability will provide better user experience. [41] [42]

8.7.1 R-CORD architecture

There is a common thing between R-CORD and the guy shown in Figure 8-23, his name was Philip II, he used to be a Greek king and was the father of the famous Alexander the great, but what this have to do with CORD any way!!, just keep him in mind and we will come back to him later.

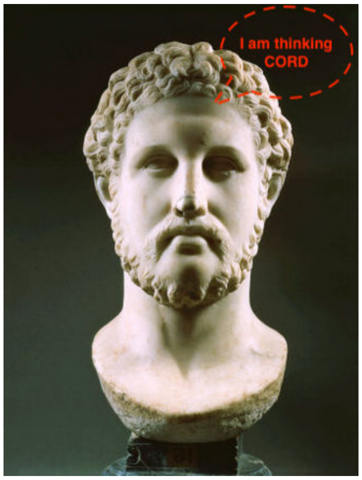
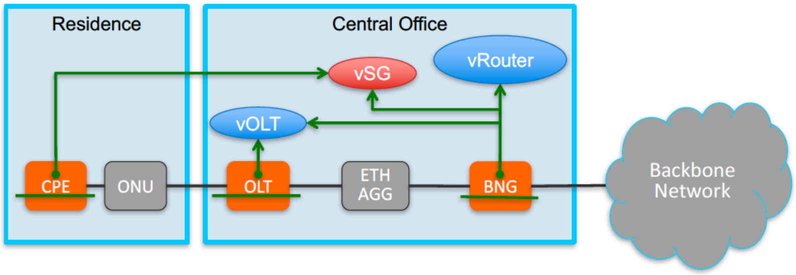


Figure 8-23 Philip II of Macedon

To better understand R-CORD service chain building blocks, let's first review disaggregation principles that were used to develop R-CORD segregated services. As shown in Figure 8-24, the legacy CO main hardware components have been disaggregated into an OCP hardware (data plane) and a virtualized control plane.

Therefore, the OLT becomes a white box OLT bare metal performs only forwarding and a control plan that lives in CORD cloud which consists of a virtual OLT agent (vOLT) runs on top of ONOS and a vOLT Hardware abstraction (vOLTHA) VNF runs in a computing node. The customer home router (CPE) control plane was taken away from the subscriber premises to run inside CORD platform as a virtual subscriber gateway (vSG) container, what left is a white box forwarding device with a minimal software to run configurations received from the control plane. The BNG disaggregation was discussed more in 8.6.4, BNG functions was disaggregated into many small services implemented inside CORD, for example vRouter, vSG and AAA above CORD.



CPE – Customer Premises Equipment
OLT – Optical Line Termination
BNG – Broadband Network Gateway

Figure 8-24[43] Traditional CO hardware disaggregation in CORD

This will bring us to R-CORD service graph shown in Figure 8-25, if you look close enough to each building block you will think there is something familiar, yes you are right! These are the same disaggregated services from the legacy FTTH CO explained in the previous paragraph. By the way, you may apply this disaggregation exercise on other CORD service graphs. This will bring us back to our friend Phillip of Macedonia, the author of the strategy “Divide and Conquer”!!, I’m pretty sure he didn’t mean CORD by then, however this is how services are implemented in CORD, first disaggregate them “Divide” then control them in small sandboxes “Conquer”. Thanks Phil you may rest in peace now.

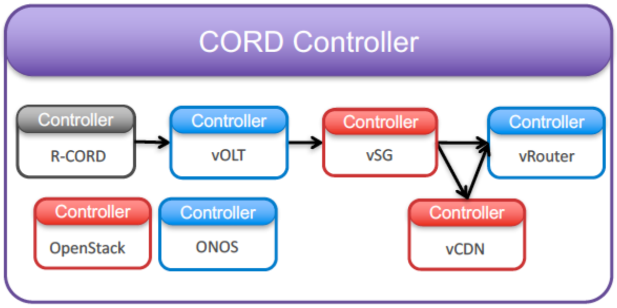


Figure 8-25 R-CORD service graph

R-CORD will include the same common (CORD) infrastructure of open source software, merchant silicon hardware and leaf spine white boxes switching fabric that connects the whole infrastructure of computing and access network hardware that serves home, mobile and enterprise customers, also metro routers are present to connect the CO to other COs or to the Core network, as shown in Figure 8-26

The FTTH access hardware is shown in the left of Figure 8-26, it represents a passive optical network that reach out to the residential subscribers (ONUs), where OLTs are installed inside the CO. An R-CORD OLT is a bare metal hardware that's made of a merchant silicon and managed by an open source software that lives outside the white box and inside the CORD computing nodes. Currently there are two available OLT white boxes, GPON and XG-PON both uses a merchant silicon from Broadcom with a small stack of an open source software installed inside the OLT as a bare metal operating system, the OLT's control plane have been taken away to run as a VNF inside the computing nodes to manage the OLT white boxes. The VNF that manages the OLT is referred to as a virtual OLT hardware abstraction (VOLTHA), which also interacts through its northbound interface with the SDN controller (ONOS) and the control applications running on top of it such as vOLT service, as well as other VNFs in CORD's data plane which live physically inside the computing nodes as VMs or containers such as a virtual subscriber gateway (vSG). [41] [42]

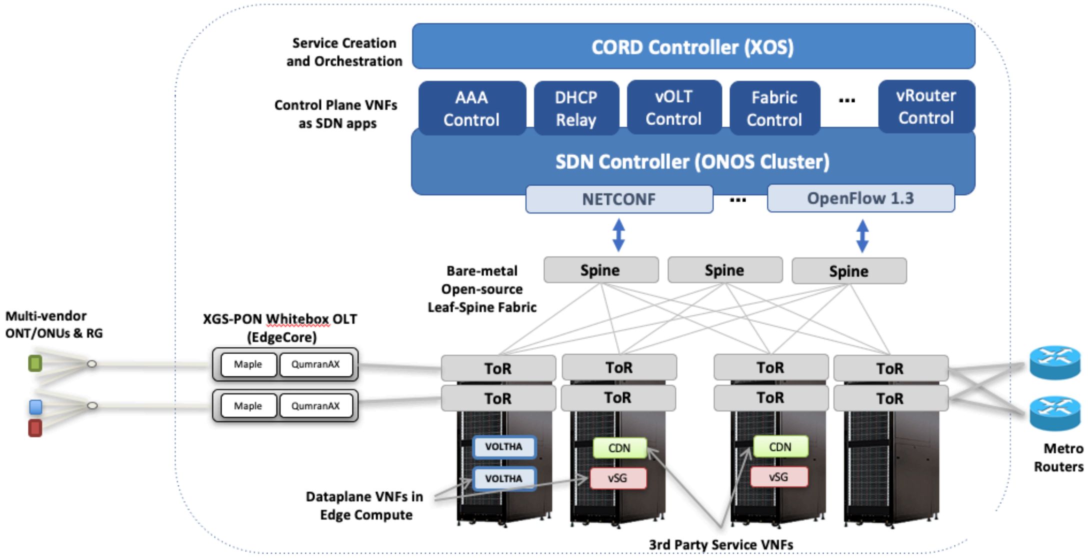


Figure 8-26 [41] R-CORD architecture

The PON's control plane VOLTHA is a VM that manages the OLTs and the ONUs, where it performs all PON standard operations. The main reason for VOLTHA is to provide a hardware abstraction in CORD, in order that a common platform controls all OLT types regardless if they are white boxes or vendor related products, in fact the SDN controller and all VNF apps running on top of it are neither aware of the PON nor its associated GPON/XGS-PON TC layer functions such as T-Count mapping and XGEM Ports assignment, also OMCI massaging and ONUs discovery processes…etc., all these PON associated processes are hidden and abstracted away from the SDN controller. Therefore, the SDN doesn't see any PON details, and it only sees a pseudo Ethernet switch and controls it similar to any other white box switch inside the switching fabric. Later we will discuss VOLTHA in more details. [41] [42]

8.7.2 Virtual subscriber gateway (vSG)

vSG service runs as a container inside a computing node to provide the residential subscriber functions such as (DHCP, NAT, Firewall, Parent control, VoIP…etc.), it could be programmed by the subscriber or the operator through a special GUI which passes control instructions through XOS, ONOS to the vSG.

8.7.2.1 VOLTHA Architecture

As explained in 8.7.1, VOLTHA abstracts the PON access network as a pseudo Ethernet switch to be controlled by one SDN, but how it does that?

Figure 8-27 shows VOLTHA architecture, where the bottom layer has southbound device adapters, these adapters enable configuring different types of OLTs and ONUs either a white box bare metal or a vendor related. Currently the following collaborated vendor's equipment are supported (Calix, Ciena, Tibit, Nokia and Adtran).

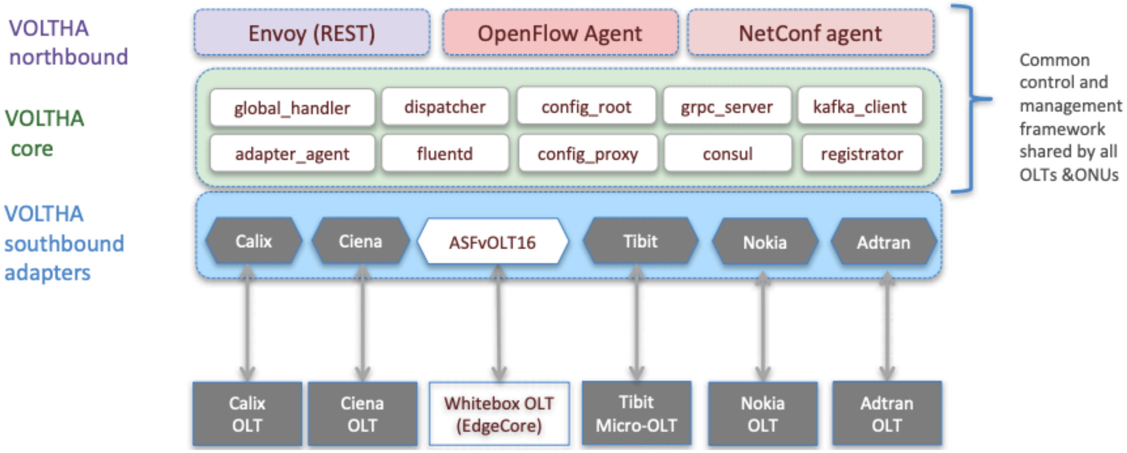


Figure 8-27 [42] VOLTHA acrchitecture

VOLTHA's northbound interfaces and VOLTHA core are one common management frame work that is shared by all OLTs and ONUs to realize the PON services. [41] [42]

8.7.2.2 VOLTHA components

VOLTHA uses Docker in Swarm mode as a platform to spin up services, either independently or in a cluster, a single service could be scaled individually if needed by spinning new instances for an example to handle more load or to optimize performance. More info about Docker and its modes, provided at the following link (https://docs.docker.com/). [44][45]

In VOLTHA all services run in a load-balancing cluster. For a failover propose there are 2N+1 servers (i.e. v-core containers) run simultaneously and handles the load equally, for any reason if one server has failed then the other instances will continue to provide services while Docker will restart the failed instance and recover all its failed services. [44][45]

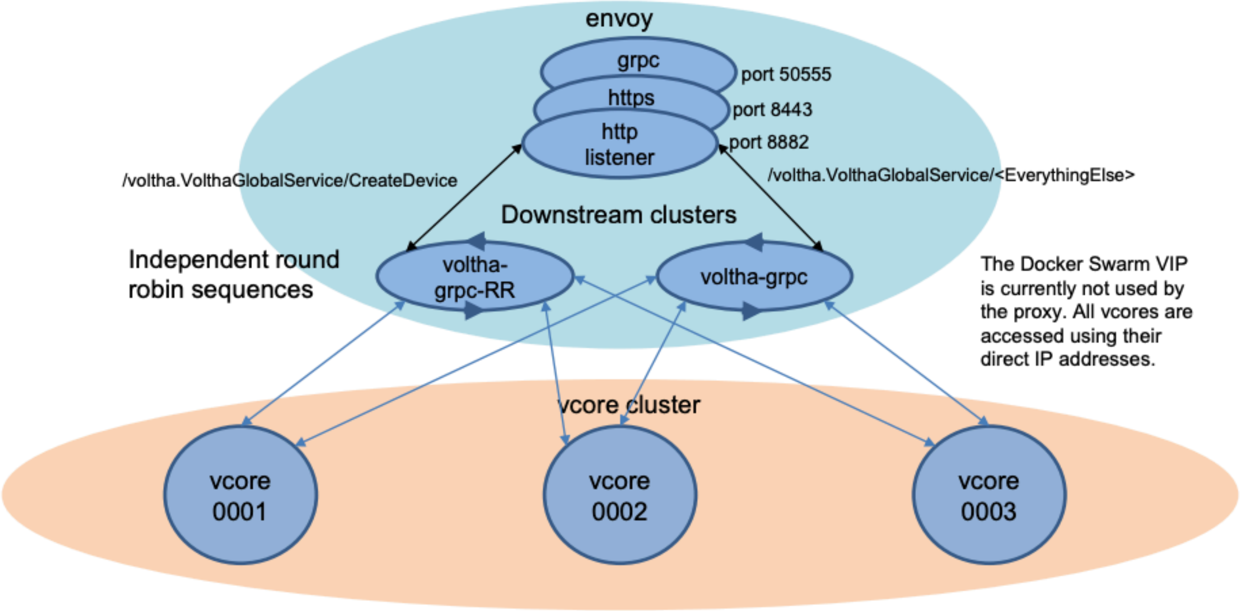

envoy
grpc
port 50555
https
port 8443
http listener
port 8882
/voltha.VolthaGlobalService/CreateDevice
/voltha.VolthaGlobalService/<EverythingElse>
Downstream clusters
Independent round robin sequences
voltha-grpc-RR
voltha-grpc
The Docker Swarm VIP is currently not used by the proxy. All vcores are accessed using their direct IP addresses.
vcore cluster
vcore 0001
vcore 0002
vcore 0003

Figure 8-28 [44] Load balancing by Envoy proxy

The load-balancing in VOLTHA is provided by running only one “Envoy proxy” instance, the reason for choosing to use Envoy proxy, because it features a hitless reconfiguration capability, means it is able to hand over smoothly between envoy proxy instances while the new instance can take over completely before the older instance is shut down without causing any service interruption. It also supports a gRPC protocol to communicate between services, where protobufs are exchanged. The load balancing mechanism inside the Envoy service is accomplished through two round robin (RR) sequences (this is the reason behind running only one Envoy instance), one RR is used to create a new device instance (OLTs/ONUs), the other RR is used to handle other service instantiating requests which are spread equally among v-core instances, as shown in Figure 8-28. More info about Envoy is available on the following link (https://www.envoyproxy.io/docs/envoy/latest/intro/what_is_envoy). [44][45]

The SDN controller (ONOS) keeps a one to one direct communication with all created logical devices (ONUs and OLTs) through an OpenFlow (OF) adapter, where it manages and configures them, therefore their associated hardware devices. [44][45]

V-core service (container) contains three main functions, a dispatcher to direct upcoming requests from the CORD controller to the correct logical device within a v-core cluster. The coordinator manages v-core instances and their assigned work. For an example, in case of a v-core crash, the coordinator will fetch all saved service data models from the consul cluster and spin them up on a new recovered v-core instance. [44][45]

The consul cluster is used to persist v-core cluster state, stored models within the consul cluster could be restored when needed, more info about Consul is available at the following link (https://www.consul.io/). Figure 8-29 shows v-core architecture. [44][45]

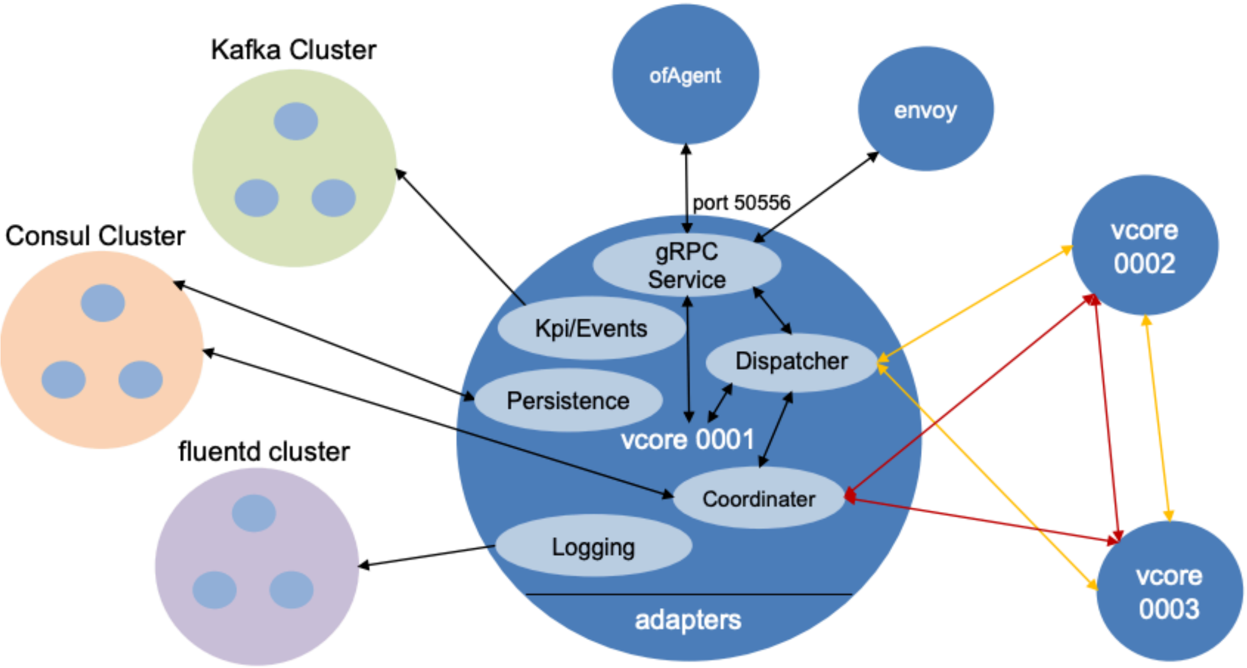


Figure 8-29 [44] v-core cluster architecture

Kafka cluster is used as events bus, where the upstream services (VNFs) are able to request KPIs, therefore, issue back the appropriate controls. More info about Kafka are available at (https://www.confluent.io/what-is-apache-kafka/).

Flaunted cluster is used to collect and store v-core logging info, it's an open source data collector tool that unifies data collection, also facilitate data access through various filters. More info about flaunted is available at the developer home page (https://www.fluentd.org/architecture).

Grafana is used to visualize KPIs analysis, it connects directly with the Kafka to fetch the KPIs then creates a dashboard to visualize them, thus displays VOLTHA's real time performance. More info about Grafana is available at its creator's website here (https://grafana.com/). [44][45]

8.8 Summary

This chapter started with an introduction about today's challenges facing traditional centralized cloud DC architecture and the burden placed on networks infrastructure due to the large amount of collected data to be analyzed in the cloud, and the high latency in responses due to long distances separating edges from the network core. Then we discussed the edge computing concept as a solution. Later, we reviewed a new revolutionary solution CORD, an edge service delivery platform that brings cloud agility, economics and best practices into the Telecom's CO. Then we described CORD architecture and main building blocks, and how micro services are disaggregated and implemented inside CORD, we also spoke about CORD controller, service models, services portfolio and the automated configuration. In Addition, we discussed Trellis as CORD switching fabric and its main components, then clarified R-CORD track's architecture and its main building blocks.

Section 3: R-CORD implementation

Chapter 9 R-CORD implementation

9.1 R-CORD real-case implementation (Demo)

A compact R-CORD system was run at Berlin Broadband World Forum (BBWF) to demonstrate a multi-vendor capable platform. Figure [44] 9-1 shows the implemented setup.

CORD
Central Office Re-architected as a Datacenter
R-CORD and VOLTHA Demonstration
urce SDN/NFV Platform for Delivering Multi-vendor Residential
OPEN ACCESS
ONF
Open Networking Foundation

Figure [44] 9-1 BBWF R-CORD demo

This is a full functional CORD platform in its smallest configuration. However, CORD platform could be as large as sixteen racks or more, built with bare metal pizza boxes, similar to the one shown in Figure [44] 9-1. This implementation includes the following components:

- Head-node: it's a commodity server, a single pizza box in this implementation that runs the SDN controller (ONOS), CORD controller and orchestrator (XOS), K8/Docker, MaaS and control apps above CORD (e.g. vRouter).
- Compute node, is another commodity hardware server used to run data plane VNFs, which could be VMs or Containers. Also, it runs the VOLTHA and vSG.
- White box switch, represents the switching fabric controlled by ONOS. In this small demo, a single switch was used to connect the upstream, the compute nodes and

the access devices.

- Two OLTs were used, one white box OLT, and another Nokia OLT to demonstrate the multivendor capability of R-CORD platform, both OLTs are connected through the VOLTHA southbound adapters. Each OLT is connected to one ONU in this example.
- Two ONUs were used, to represent a residential unit, both ONUs were connected to a residential gateway that provides a Wi-Fi access for the end users devices (tablets, smartphones, smart TV…etc.)

Figure 9-2, shows logical building blocks of used R-CORD system.

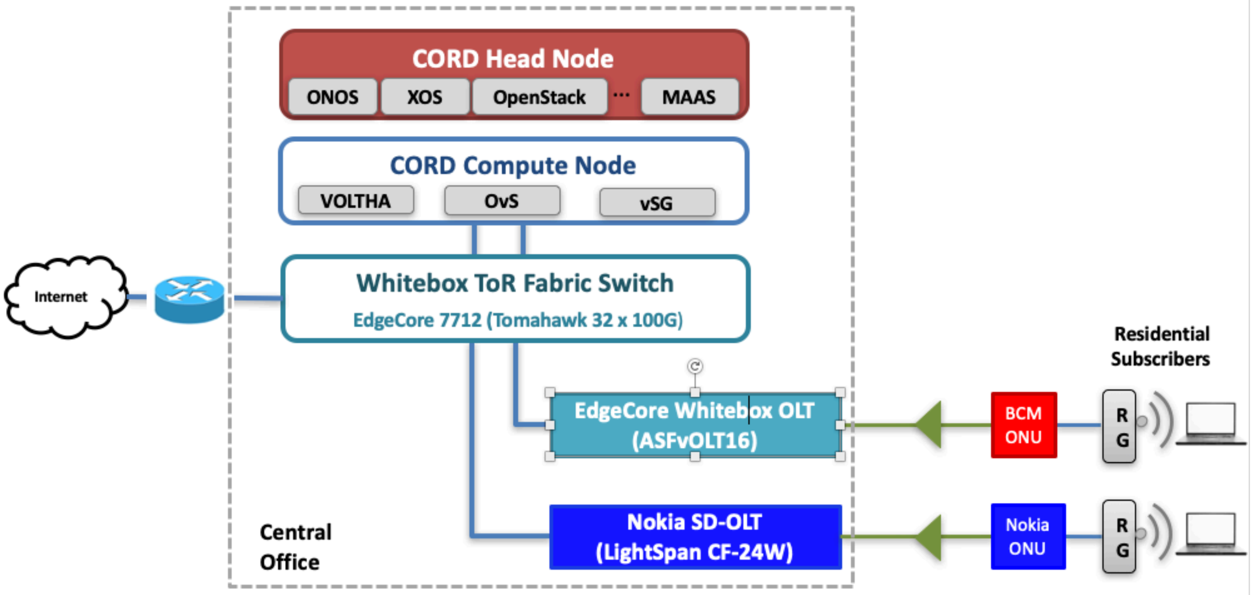


Figure 9-2 [45] BBWF R-CORD Demo, Logical architecture

In order to install and configure the platform we need to follow same steps discussed in 8.58.7. After CORD core set is installed and running, the vOLT service is created by running Tosca Yaml models on XoS, by running several (xProto) service models, therefore service components will get instantiating, including configuring a new OLT and ONUs in the PON network through the VOLTHA.

The following will briefly explain what happen when a user is connected to the R-CORD access network:

- When a new user connects to the residential gateway, the ONU will send its authentication info to the OLT which forwards it to VOLTHA at the computing node up to AAA authentication service running on top of ONOS. If the authentication fails then the traffic of this specific user gets blocked.

- IT’s also important to mention that ONOS assigns a VLAN per customer (C-VLAN) and per a subscriber (S-VLAN) to isolate the traffic per client and per subscriber. To achieve that, the ONOS request VOLTHA to create the required VLANs, VOLTHA then configure C-VLAN on the OLT white box and the S-VLAN on the ONU, where VLAN creation only happen after a successful subscriber authentication. Simultaneously the ONOS tells the switching fabric through the virtual switch (OvS) to forward those VLANS to the virtual subscriber gateway instance (vSG) therefore it will learn about them.
- For an example, if the user wants to connect to the Internet, then the data traffic flows in the following order; through the residential gateway (RG), the ONU, the OLT white box/Nokia, it doesn’t need to go to VOLTHA as VOLTHA there only for controlling, so the traffic then goes directly through the switching fabric to the computing node to the OvS, then vSG, again it goes back to the OvS and from there to the switching fabric (i.e. edge leaf), to the Internet.
- vSG could be used to provide additional services, for an example it could be controlled through a subscriber portal to configure parental control to restricts, or through a service operator portal to block some payment overdue subscribers. The portal is just a graphical API interface to the XoS which will communicate the configuration with the correct vSG instance for that specific subscriber to perform the required action.
- Using same configurations method, variety of different services could be configured by the CORD operator to perform traffic control functions, for an example, they could direct the traffic to a 3rd party content delivery network (CDN) to provide better performance. [44][45]

9.2 Comparison between R-CORD and legacy CO

In this section, we will make a fair comparison between R-CORD solution and the legacy CO implementation, I will try not to advocate a specific solution, all I will do is to perform a reasonable comparison and let the reader decide the winner.

Here is how it works, first I will mention an aspect then I'll explain how it's applied within each technology starting with legacy CO and moving to CORD. Let the games begin!!

9.3 Hardware

Description	Legacy CO	R-CORD
Acquisition	Expensive proprietary hardware components.	Cheap open source hardware built on merchant silicon.
Software Stack	Full stack of software is installed on every hardware device in order to implement both the control and the forwarding planes. Uses licensed vendor proprietary software stack, therefore you can't change any software aspect unless you go through vendor and most likely will be expensive.	Only light white software stack is needed to communicate with the controller. Uses an open source license-free software stack. The Service providers have complete freedom to customize services that fit their use-cases, plus they own intellectuality of any new software solutions, which could become an alternative source of revenue.
Configuration syntax	You need to use a special configuration syntax to configure and manage every network device individually. For an example: Cisco CLI is different from Juniper, then Nokia...etc.	Only high level automated configuration, based on simple easy to read data models which is common for all hardware devises, where hardware is abstracted by VOLTHA, therefore same configuration is used for all access devices regardless if they are white boxes or vendor supported proprietary devices.
Configuration simplicity	Devices have to be configured individually, which takes a long time and needs more redundancy to make sure services continue when a device	Full automated platform configuration is done centrally by the CORD controller which configures the whole platform at once. Regardless of its size

	is off grid. Also, different expertise is needed to perform the configuration.	and hardware type. Also, the SDN controller (ONOS) is able to configure the network fabric simultaneously in minutes.
Interoperability	Interoperability is uncertain between different vendor equipment where some standards may be implemented differently from vendor to vendor.	Interoperability is not an issue as all hardware is abstracted.
Hardware programmability	Hardware is not programmable, the ASIC pipeline is fixed, for example if your switch doesn't know PPP, then you can't or it's a tremendously expensive and complicated to reprogram it to understand PPP.	It's possible to use a special programmable hardware, such as using P4 (programming protocol independent packet processor), it's easy to reprogram your hardware to perform new functions that didn't exist before, such as basic policy functions that are implemented directly on the hardware.

9.4 Services

Description	**Legacy CO**	**R-CORD**
Services Structure	Services are built-in inside appliances in a proprietary hardware. Traditional services are fixed inside the hardware, also it's not possible to combine different services to form a service for an example.	Disaggregated micro services lives in CORD cloud as a VMs or containers. In CORD, everything is implemented as a service (XaaS), not just the traditional network services such as firewall for an example, in CORD you could construct a new service by combining one or more service in a service graph,

		also you can provide the platform components as a service such as Network fabric (NaaS), or the infrastructure as a service (IaaS).
Implementation	To implement a new service, you need to acquire an expensive proprietary hardware appliance, and the service you buy is what you get, you can’t for an example modify it or build a new service. To configure the network appliances, you need to do it individually per device, using a vendor-specific CLI syntax, which takes a long time and needs more resources and expertise to do it safely.	Services implementation is automated, they are built very easy by running predefined data models. CORD uses service graph that contains different micro services connected together, it’s very easy to define and implement a new service, only by defining a new service graph. Service configuration is fairly straight forward simple process and could be performed in minutes.
Add new feature or new service	Will require a new hardware in most cases.	You only need to add the new feature by updating the service container. New services could be added to the platform by adding a new software containers, and create new service graphs.
Scalability	You need to buy more hardware, install it, and make necessary configuration to the device and all associate networking infrastructure.	Services are dynamically scaled up and down according to load condition, by spinning up or terminating services instances.
Development	It’s not possible to do it alone, you have to request	Services are working independently from one

	any changes from the vendor and wait until it's available. Some hardware appliances are combining some services together, therefore it's not possible to develop these services individually.	another, you can develop a service without affecting other services. It's done fairly quickly since teams can focus on individual services.
Service portfolio	Services in the legacy CO are mainly based on transmission of end user's traffic and provide connectivity to the Internet, IPTV, PTSN…etc.	Besides supporting all traditional services in a legacy CO, a new Cloud-based services are possible to implement, such as Infrastructure as a service (IaaS), Networking as a service (NaaS)…etc.
Edge computing services	Legacy CO mainly provides connectivity to centralized cloud DC, can't support edge computing unless with costly solutions built on proprietary hardware for a specific service.	CORD is a realization of Edge computing, the platform is ready not to just implement current services, but also for future services yet to be innovated for IoT, G5, NGPON2, big data analysis, real time low latency futuristic applications and so on.

9.5 Network fabric

Description	Legacy CO	R-CORD
Fabric components	Built from proprietary equipment. Every network device (switch/router) independently implements OSI protocol stack.	Forwarding plane is split from the control plane. Forwarding plane (underlay fabric) consists of bare metal white boxes built from merchant silicon organized in a leaf spin architecture. The control plan

	Every device has a control and forwarding plane built-in.	is an SDN controller lives inside the CORD cloud as an open source software implemented as a container.
Main function	Provide connectivity between CO components and outside network, it's used only for transmission. Cannot be configured by Subscriber, only done by the operator.	Connects service components, in other words, the overlay virtual network connects the containers where underlay fabric carries the traffic. Fabric in CORD could be provided as a service or network as a service (NaaS).
Configuration	Every network component has to be configured individually using a specific syntax per vendor, which takes a lot of time and resources.	Configuration is done by the SDN controller in a matter of minutes you can configure the whole network.
Performance	Slow convergence especially in large network. Vulnerable to traffic peaks, where congestion will cause performance to deteriorate quickly. The only solution is to overprovision network resources, which reduce network utilization and adds cost. Heavy network management overhead. Duplication of routing protocol stack processing.	Very fast network convergence, as the underlay fabric will receive forwarding entries pushed down from the centralized SDN controller at once. Dynamically react to traffic peaks by scaling recourses up through instantiating virtual network services. High utilization of the shared fabric, where resources are scaled up or down as needed. There is no network overhead as underlay fabric only talk to the ONOS controller.

		Routing protocol stack is executed only once at the controller then forwarding tables are pushed down to the underlay fabric.

9.6 Other aspects

Description	Legacy CO	R-CORD
Monitoring and management	- Legacy Mechanisms to collect traffic statistics by using specialized protocols such as NetFlow and sFlow. - Uses royalty 3rd party software agents to collect information. - Information can be obtained only from ports of network devices, can only be distinguished by traffic type (e.g. unicast, broadcast, multicast, drop ...etc.). - Monitoring can't be fine-grained thus very inefficient to make critical decisions such as optimize network resources. - Monitor system NMS can't directly manage traffic flow.[24]	- Monitoring in CORD is architected as multi-tenant, scalable service. - High granularity, you can monitor everything to any granularity, for an example you can monitor each service instance (container). - Different open source tools are used. - Real time monitoring, therefore used as a loopback control mechanism to reconfigure the platform, for an example to scale a service up or down. - In CORD Monitoring is provided as a service, MaaS.[46]
Power saving	Hardware power consumption is very difficult to optimize, there might be a vendor hardware that supports power saving techniques, however it will remain very limited at the	CORD platform is dynamically scalable, hardware could be optimized to condensate running containers and fabric inside some POD and place ideal

	device level but not the whole infrastructure. Generally, lack of dynamic scalability required to have overprovisioned hardware to support unexpected peaks, which means more unnecessary power consumption. It’s not an energy aware, it can’t reduce energy consumption according to load status. Overprovisioned hardware is not utilized efficiently. More equipment, needs larger supporting facilities (e.g. Cooling, UPS, Generator...etc.) Thereby it consumes more power.	hardware into power safe modes to save power. Same CORD platform is used to handle different tenant’s traffic types, which minimize hardware footprint thus power consumption. Energy-aware, it’s able to scale down and turns off unnecessary resources to save power. The common infrastructure for all use-cases means less equipment and less power consumption. Less equipment means less supporting facilities, less cooling…etc. Thus, consumes less power.
Supporting multi tenant’s traffic (use cases)	In legacy CO, there should be a dedicated racks of vendor proprietary equipment to support certain traffic such as traffic produced by PON, or a Mobile network, which uses different protocols and requires very different class of service.	CORD platform is common for all use cases, same platform will work for Residential, Mobile and enterprise customer, it only uses a different access equipment in the last mile, the rest is shared, and near future roadmap, CORD will have a multi-access edge cloud solution with one access network supports both Mobile and Residential customers.

Environment (Greenhouse emission)	Not energy aware, consumes more power.	Energy-aware, consumes less power and more efficient, therefore more environmentally friendly.
Cost (Opex and Capex)	-Expensive proprietary equipment -Expensive Licensed software stacks -Expensive upgrade and costly to customize solutions (software, hardware or both). -Expensive to operate and maintain. -Overprovisioned hardware to deal with peaks.	- Unified common multitenant platform for all use-cases, means less equipment. - Cheap open source Commodity servers, white boxes built form merchant silicon. - License-free open source software, cheap to acquire and upgrade. - Cheaper Centralized management. - Save power there for reduce energy cost.
Economy	- Only traditional CO services. - Can’t provide cloud services. - Very hard to support new services. - Very slow convergence, un-scalable, overprovisioning of resources, traditional business model, and lack of programmability.	- Support for old services. - Programmable platform can introduce new services. - Provides cloud services (e.g. PaaS, NaaS, IaaS, etc.) - Very easy to implement new services. - Support future to be developed service, All of that makes it a very successful investment, and its ability to support multitenant on the same platform provides high ROI.

9.7 Conclusion

As a telecom service provider, what will you get by implementing CORD in your CO?

- You will be able to provide new revolutionary services similar to cloud successful services model while supporting legacy CO services with the advantage of cloud agility and scalability in your CO.
- You will have the advantage of being closer to end user over the cloud, which means better interaction and lower latency that enables you to provide futuristic real time-based services, which is not possible with today's cloud DC architecture.
- Your CORD platform is designed to provide everything as a service (XaaS), this opens new possibility to develop and provide newer services (for example new service graphs) the only limitation is developer's imagination.
- You will be able to support multi-tenant traffic on the same platform, your FTTH, mobile, enterprise customer can use the same platform, and for you this means better return on investment (ROI).
- You will own a smart platform that dynamically scale up or down according to traffic condition, this means better service quality, efficient resources utilization, less energy consumption, therefore more environmental friendly infrastructure, and less cost.
- You will have a platform that is open source both software and hardware, no more expensive vendor royalties, proprietary hardware and licensed software. You will have a full ownership of your hardware and software, therefore any customized solution you develop, it will be owned by you and can be an alternative source of revenue.
- You will have a programmable hardware that enables implementation of new features.
- You will have a platform that implements services and configurations in matter of minutes instead of weeks. Therefore, a better user experience, and services will be quickly presented to the market, which give you an advantage over

competition.

- You will have a better monitoring tools that wasn't available in the legacy CO, therefore you will be able to control your platform on real-time and achieve even better performance.

9.8 Summary

In this study, I started with explaining various FTTH PON technologies, some already existed in the traditional CO such as GPON and XG-PON, and some will be implemented in the near future, such as XGS-PON and NG-PON2. Then I explained SDN and NFV as the main technologies used to deploy network fabric in a virtualized DC which revolutionized network implementation, configuration, and management with high efficiency and flexibility. Next I talked about the new concept of edge computing as a new solution to solve future challenges facing the tradition model of cloud-based services which does not fulfill the requirement of low latency real time services and IOT real time control systems, which expected to keep growing fast, especially with the introduction of the new broadband networks such as 5G and NGPON2. In the last two chapters, I spoke about CORD as a new solution for an Edge service delivery platform that is based on commodity servers and a white boxes fabric built from merchant silicon and uses an open software to implement and manage services using state of art tools, and explained how CORD will bring cloud economics, agility, flexibility and scalability into Telecom CO. Finally, I made a comparison between the traditional CO and the CORD platform where I tried to help the reader to see for himself the advantages that the Telecom operator will achieve through implementing CORD.

At the end, I hope that you agree that CORD will be a game changer, and will have a profound influence on what and how services will be provided in the future. CORD was only a concept in 2015, amazingly it's a very living platform today, thanks to ONF and all the generous collaborators. Currently there are many CORD trials going on worldwide, besides that CORD still under development and keeps constantly changing, it's expected to be available as a product in a short window frame as many service providers are generously contributing into its development.

I hope that you enjoyed reading this thesis report as much as I did when I wrote it, and I will be happy to receive any question, comment or a suggestion. You may get in touch with me through my email (elragaie@ualberta.ca).

References

[1] J. Farmer, B. Lane, K. Bourg, and W. Wang, *FTTx networks : technology implementation and operation*. 2017.

[2] C. F. Lam, Passive optical networks : principles and practice. Elsevier/Academic Press, 2007.

[3] E. S. Quellmalz and J. W. Pellegrino, *FTTx PON Technology and testing*, vol. 323, no. 5910. 2005.

[4] D&O COMMITTEE of FTTH Council Europe, “FTTH Handbook Edition 8,” *FTTH Counc. Eur.*, no. 5, pp. 1–161.

[5] I.-T. S. G. 15, “Broadband optical access systems based on Passive Optical Networks (PON) G.983.1,” *ITU-T Recomm. G.983.1*, 2005.

[6] I.-T. S. G. 15, “G.984.1 Gigabit-capable passive optical networks (GPON): General characteristics,” *Networks*, pp. 1–43, 2008.

[7] SIEPON Working Group, “IEEE P1904.1 SIEPON Working Group.” [Online]. Available: http://www.ieee1904.org/1/index.html. [Accessed: 17-Sep-2018].

[8] IEEE P802.3bn EPON Protocol over Coax (EPoC) Task Force, “IEEE P802.3bn EPON Protocol over Coax (EPoC) Task Force, Home Page.” [Online]. Available: http://www.ieee802.org/3/bn/. [Accessed: 17-Sep-2018].

[9] ITU-T, “ITU: Committed to connecting the world.” [Online]. Available: https://www.itu.int/en/Pages/default.aspx. [Accessed: 26-Sep-2018].

[10] D. Hood, E. Trojer, and Wiley InterScience (Online service), *Gigabit-capable passive optical networks*. Wiley, 2012.

[11] I.-T. S. G. 15, “G.984.3, Gigabit-capable passive optical networks (G-PON): Transmission convergence layer specification,” pp. 1–170, 2014.

[12] ITU-T Study Group 15, “G.987: 10-Gigabit-capable passive optical network (XG-PON) systems: Definitions, abbreviations and acronyms,” *ITU-T Recommendation G.987*. pp. 1–26, 2012.

[13] ITU-T, “XG-PON booklet by ITU-T,” 2010.

[14] ITU-T Study Group 15, “G.987.1 : 10-Gigabit-capable passive optical networks (XG-PON): General requirements,” 2016.

[15] ITU-T Study Group 15, “G.9807.1: 10-Gigabit-capable symmetric passive optical network (XGS-PON),” *Itu-T G-Series Recomm.*, 2016.

[16] ITU-T Study Group 15, “G.987.2: 10-Gigabit-capable passive optical networks (XG-PON): Physical media dependent (PMD) layer specification,” 2016.

[17] ITU-T Study Group 15, “G.987.3: 10-Gigabit-capable passive optical networks (XG-PON): Transmission convergence (TC) layer specification,” vol. 2.0, pp. 1–146, 2014.

[18] ITU-T Study Group 15, “G.989 : 40-Gigabit-capable passive optical networks (NG-PON2): Definitions, abbreviations and acronyms,” 2015.

[19] ITU-T Study Group 15, “G.989.1 : 40-Gigabit-capable passive optical networks (NG-PON2): General requirements,” 2013.

[20] NG-PON2 council, “NG-PON2 track – Las Vegas Oct 2017 , broadband access summit event,” 2017.

[21] ITU-T Study Group 15, “G.989.2 : 40-Gigabit-capable passive optical networks 2 (NG-PON2): Physical media dependent (PMD) layer specification,” pp. 1–108, 2014.

[22] ITU-T Study Group 15, “G.989.3: 40-Gigabit-capable passive optical networks (NG-PON2): Transmission convergence layer specification,” 2015.

[23] B. Agerbeck, “Day1 session2 CORD build Keynote Nov 2017,” 2012.

[24] J. Taheri, *Big Data and Software Defined Networks - Knovel*. Institution of Engineering and Technology, 2018.

[25] Rene Molenaar, “CEF (Cisco Express Forwarding) | NetworkLessons.com,” 2013. [Online]. Available: https://networklessons.com/cisco/ccnp-route/cef-cisco-express-forwarding/. [Accessed: 22-Dec-2018].

[26] Y. Zhang, Network function virtualization : concepts and applicability in 5G networks. .

[27] O. N. L. Larry Peterson, “CORD: Central Office Re-Architected as a Datacenter - IEEE Software Defined Networks,” 2018. [Online]. Available: https://sdn.ieee.org/newsletter/november-2015/cord-central-office-re-architected-as-a-datacenter. [Accessed: 24-Dec-2018].

[28] Nippon Telegraph and Telephone Corporation Press release., “NTT HOME &gt; NTT Press Releases &gt; Announcing the &quot;Edge computing&quot; concept and the &quot;Edge accelerated Web platform&quot; prototype to improve response time of cloud applications.,” 2014. [Online]. Available: http://www.ntt.co.jp/news2014/1401e/140123a.html. [Accessed: 24-Dec-2018].

[29] Philip Adams, “When will Edge Computing Live Up to its Hype?,” *KNect365*, 2017. [Online]. Available: https://knect365.com/cloud-enterprise-tech/article/3157ea5f-c11c-49c3-9350-74363829ecf2/when-will-edge-computing-live-up-to-its-hype. [Accessed: 24-Dec-2018].

[30] P. Garcia Lopez *et al.*, “Edge-centric Computing,” *ACM SIGCOMM Comput. Commun. Rev.*, vol. 45, no. 5, pp. 37–42, Sep. 2015.

[31] C. by Aj, “What Is Central Office Re-architected as a Data Center (CORD)? - Ciena,” 2018. [Online]. Available: https://www.ciena.com/insights/what-is/What-Is-CORD.html. [Accessed: 01-Jan-2019].

[32] O. N. L. Larry Peterson, “CORD build 2017: Day1_Session8_CORD-Build.” 2017.

[33] O. Larry Peterson, “CORD Technical Overview - YouTube,” *CORD Build, OpenCORD*, 2017. [Online]. Available: https://www.youtube.com/watch?v=eEjb9zXbuEQ&list=PLCnPGaNt7C5f5XJTM8hhxyaxlskYZWRDm&index=7. [Accessed: 01-Jan-2019].

[34] Cloudify.co, “What is TOSCA &amp; Why is it important? | Cloudify,” 2018. [Online]. Available: https://cloudify.co/tosca/what-is-tosca/. [Accessed: 01-Jan-2019].

[35] Larry Peterson, “Trellis: CORD Network Infrastructure - CORD - CORD Wiki,” 2017. [Online]. Available: https://wiki.opencord.org/display/CORD/Trellis%3A+CORD+Network+Infrastructure. [Accessed: 09-Jan-2019].

[36] Saurav Das, “Trellis Underlay Fabric - CORD - CORD Wiki,” *OpenCORD*, 2017. [Online]. Available: https://wiki.opencord.org/display/CORD/Trellis+Underlay+Fabric. [Accessed: 09-Jan-2019].

[37] OpenCORD, “Trellis Introduction - YouTube,” 2017. [Online]. Available: https://www.youtube.com/watch?v=DNpgTkFEzw0&list=PLCnPGaNt7C5clVWthmBdS1j-HK-oh-yhG&index=2. [Accessed: 09-Jan-2019].

[38] Saurav Das, “Virtual Network Overlay - CORD - CORD Wiki,” *OpenCORD*, 2017. [Online]. Available: https://wiki.opencord.org/display/CORD/Virtual+Network+Overlay. [Accessed: 09-Jan-2019].

[39] Larry Peterson, “Virtual Router (vRouter) - CORD - CORD Wiki,” *OpenCORD*, 2017. [Online]. Available: https://wiki.opencord.org/pages/viewpage.action?pageId=1278093. [Accessed: 09-Jan-2019].

[40] Saurav Das, “Trellis_Intro-CORDBuild-Day3.” CORD Build 2017, 2017.

[41] S. Das, “R-CORD_VOLTHA_Intro-CORDBuild-Day1 (1).” 2017.

[42] S. Y. Saurav Das, “VOLTHA - CORD - CORD Wiki,” 2018. [Online]. Available: https://wiki.opencord.org/display/CORD/VOLTHA. [Accessed: 10-Jan-2019].

[43] C. M. E. Cloud and L. Peterson, “CORD: Multi-Access Edge Cloud.”

[44] C. Sr. Director Open Source, “CORD Build 2017: VOLTHA Architecture in a clustered HA co...,” *CORD build 2017*, 2017. [Online]. Available: https://cordbuild2017.sched.com/event/CFGi/voltha-architecture-in-a-clustered-ha-configuration. [Accessed: 10-Jan-2019].

[45] Saurav Das, “CORD Build 2017: Introduction to R-CORD and VOLTHA,” *CORD build*, 2017. [Online]. Available: https://cordbuild2017.sched.com/event/CFGh/introduction-to-r-cord-and-voltha. [Accessed: 10-Jan-2019].

[46] Srikanth Vavilapalli, “Monitoring Service - Software Components - CORD - CORD Wiki,” 2017. [Online]. Available: https://wiki.opencord.org/display/CORD/Monitoring+Service+-+Software+Components. [Accessed: 11-Jan-2019].