ABSTRACTION LAYER FOR IMPLEMENTATION OF EXTENSIONS IN PROGRAMMABLE NETWORKS

Collaborative project co-funded by the European Commission within the Seventh Framework Programme

Grant agreement no:	317880
Project acronym:	ALIEN
Project full title:	"Abstraction Layer for Implementation of Extensions in programmable Networks"
Project start date:	01/10/12
Project duration:	24 months

Deliverable D3.1
Hardware platforms and switching constraints

Version 1.0

Due date:	28/02/2013
Submission date:	28/02/2013
Deliverable leader:	UCL
Author list:	Richard G. Clegg, Raul Landa (UCL), Remigiusz Rajewski, Marek Michalski (PUT), Artur Binczewski, Bartosz Belter, Damian Parniewicz, Iwo Olszewski, Łukasz Ogrodowczyk, Krzysztof Dombek, Artur Juszczyk (PSNC), Matteo Gerola (Create-Net), Mehdi Rashidfard (UBristol), Marc Bruyere (Dell/Force 10), Jon Matias (UPV/EHU), Hagen Woesner, Andreas Koespel (EICT)

Dissemination Level

☒	**PU:**	Public
□	**PP:**	Restricted to other programme participants (including the Commission Services)
□	**RE:**	Restricted to a group specified by the consortium (including the Commission Services)
□	**CO:**	Confidential, only for members of the consortium (including the Commission Services)

Abstract

The main purpose of this document is to describe in detail the equipment (ALIEN hardware) at each partner site. This description is sufficiently detailed as to include the exposed protocols for data and control and the transmission mechanisms used by the hardware. The aim of the document is to present the information in such a way as to gather common themes which will be useful in the design of the Hardware Abstraction Layer (HAL).

Table of Contents

Figure Summary

Table Summary

Executive Summary

This deliverable describes a wide range of equipment available in the ALIEN consortium in local test-beds. All this hardware is not yet OpenFlow capable. The goal of ALIEN is to allow the use of hardware capabilities through a unified access interface for control and management. The project will introduce a new layer, the Hardware Adaptation Layer (HAL), which is intended to hide the complexity of underlying hardware and provide an abstraction for OpenFlow agents.

The following equipment is considered in ALIEN and will be referred to as “alien hardware”:

- NetFPGA cards,
- EZappliance based on EZChip NP-3 network processor,
- Cavium OCTEON Plus AMC network processor module in an ATCA system,
- Optical switches,
- EPON OLT and ONU units,
- DOCSIS hardware.

This report is considered as a database and source of information for features and capabilities of particular hardware components available in the project. The report provides all relevant information about each piece of equipment, including a physical box overview, specific data plane details (e.g. transmission technology or switching management) and control/management plane analysis (e.g. protocols exposed by equipment or configuration requirements). This document also reports some work already done in other research projects. Specifically, the document describes experiences with introducing OpenFlow on ADVA, optical ROADM equipment, performed by University of Bristol as a part of its contribution to OFELIA. This work is considered as a starting point for ALIEN and the HAL specifically in the optical domain.

The description of the alien hardware available in the consortium is preceded by an overall description of the partners’ local test-beds. This information will be used in other project work packages while planning scenarios for an integration with OFELIA facilities and scheduling specific experiments (e.g. CCN application on an OpenFlow environment).

The document introduces the concept of common hardware themes. This is the first attempt to create thematic groupings of alien hardware which can be treated in in a similar way by while designing the HAL.

1 Introduction

This document contains descriptions of the partner testbeds and, more importantly the target hardware for ALIEN. A major outcome for the ALIEN project is to implement OpenFlow on new hardware platforms via a common “Hardware Abstraction Layer” or HAL. This hardware will be described throughout this document as ALIEN hardware. The document provides the descriptions for each piece of hardware where OpenFlow will be implemented in such detail as to aid the design of the HAL. For this reason the document contains precise details of the hardware in terms of data and control plane both in terms of capabilities and in terms of protocols used.

The structure of the document is as follows: in section 2 the testbed at each partner site is described at a high level. This is to provide context for the later description of the ALIEN hardware. In sections 3 through to 9 the specific pieces of hardware are described in detail. Section 10 brings together common threads from the equipment description to create the building blocks for the HAL. Section 11 provides conclusions for this deliverable and, in particular, brings out those points most important for HAL design which can be learned from detailed analysis of the hardware.

2 Testbeds

This section describes briefly the testbed available at each ALIEN partner site. The specific description of the ALIEN hardware available is deferred to a subsequent section.

2.1 PUT testbed

The PUT testbed consists of multiple network hardware entities delivering functionalities of all layers of ISO/OSI network model. The equipment is as follows:

Router Juniper MX240 3D Universal Edge Router – an Ethernet-optimized edge router that provides both switching and carrier-class Ethernet routing, with a capacity of up to 240 gigabits per second (Gbps), full duplex. The MX240 router enables a wide range of business and residential applications and services, including high-speed transport and VPN services, next-generation broadband multiply services, and high-volume Internet data center internetworking. All ports on this equipment have a speed of 1Gbps.

Alcatel-Lucent SR 1 7750 Service Routers – two powerful routers of service provider class. Both are equipped with 10 SFP Ethernet ports with the speed of 1Gbps each.

- About 20 Cisco Enterprise router and L2/L3 switches (mostly series C2800, C3560) – this equipment is used as a referenced or access network, it is also used with CCNA/CCNP Cisco Networking Academy.
- 3 high performance virtualization servers from HP – each of them has two multicore processors (Xeon), 24 GB RAM, 2 TB HDD space, 6 Ethernet ports with the speed of 1Gbps each. It makes these servers a powerful tool for any experiments with a lot of virtual machines on it.
- The Network processor NP3 – EZappliance – two applications of NP3 network processor from EZchip. Each device has 24 Ethernet ports with the speed of 1 Gbps. This device can be programmed with dedicated environment and language. It can process Ethernet frames in almost any way.
- Demo VoIP network – PUT's educational and research laboratory network contains 3 dedicated VoIP protocol servers, 15 hardware and a lot of software IP phones. It is connected to the Internet and cooperate with the real VoIP operator.
- The Spirent Network Test Center – it is a powerful network analyzer with a very wide range of possible tests and protocols. It can generate artificial network traffic and allows measurement of many parameters, including delays with using a GPS clock. This model contains 4 Ethernet ports with the speed of 1 Gbps.

- Optical Switch 4 x 1 x 2 for fiber with 1550 nm.

All of this equipment is connected into one network with two Juniper EX 3200 switches (see Figure 2.1). They offer almost 100 ports with the speed of 1 Gbps. We also use smaller switches from Cisco and Dell. In addition we can use a lot of PCs as a workstations or clients for this network.

The PUT testbed has a direct connection to the Polish Optical Internet (PIONIER) which is also a gateway to pan-European GEANT network. Therefore, PUT testbed can be reached by a research community from all Europe and even from around the world.

An important part of the PUT input is a laboratory of NetFPGA cards. It consists of 12 NetFPGA cards, that allow the testing of any new protocol within the hardware structures of FPGA Chip. These cards are programmable in VHDL or Verilog code. They realize programmed operations directly in the hardware structure of the main chip. Each such card consists of 4 Ethernet ports with a speed of 1 Gbps. They are hosted by high performance PCs which can stand for experimental network nodes. In addition to that an FPGA laboratory is deployed with FPGA cards based on 5 demo boards with Virtex 5 chips from XIlinx. All these cards are equipped with at least 2 SFP Ethernet ports with the speed of 1 Gbps and many additional elements as VGA/DVI drivers, different memories, interfaces as SATA, and so on. We use also a lot of Spartans – smaller FPGA chips from Xilinx. More description of the NetFPGA cards can be found in section 5.

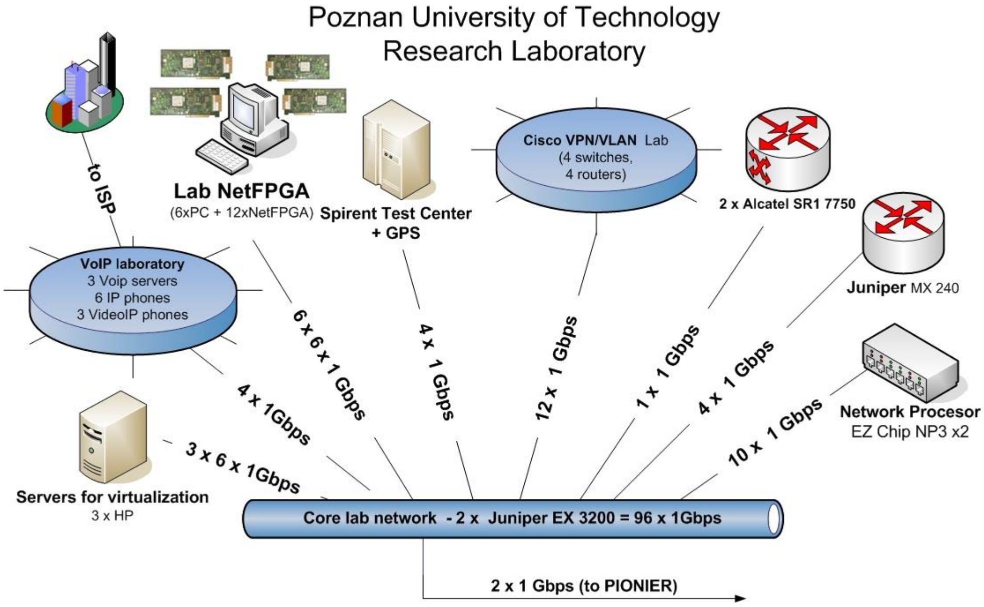


Figure 2.1 PUT research laboratory

PUT is planning to extend this environment with next generation NetFPGA cards with 4x10 Gbps Ethernet interfaces, and servers where these cards will be installed.

2.2 UCL testbed

The centre of the UCL testbed is the Gigabit Ethernet Passive Optical Network (GEPON) which is described in section 3. The UCL Converged Networks Laboratory (CONNET) was built to enable collaborative and cross-layer research. It consists of an IP testbed running over core and access optical (EPON) equipment and providing a joint platform for cross-layer interoperability and convergence studies. The PON access network testbed contains all the necessary equipment to implement, test and integrate innovative ideas in PON media distribution. Equipment includes lengths of fibre from 100m to 100km, 2 x 1Gigabit Ethernet OLTs, a 16 way splitter, a split emulator, 20 ONUs, 3 1Gigabit NetFPGAs, 2 10 Gigabit NetFPGAs, a 24 port Cisco router with 10GE capability and a range of optical and electronic measurement equipment. The CONNET laboratory can be directly connected to the OFELIA's UNIVBRIS island through the Aurora link. Aurora is a dark-fibre network to support research on photonics and optical systems. It interconnects with research groups at five universities, and provides access to intermediate locations along each fibre path where additional equipment can be sited.

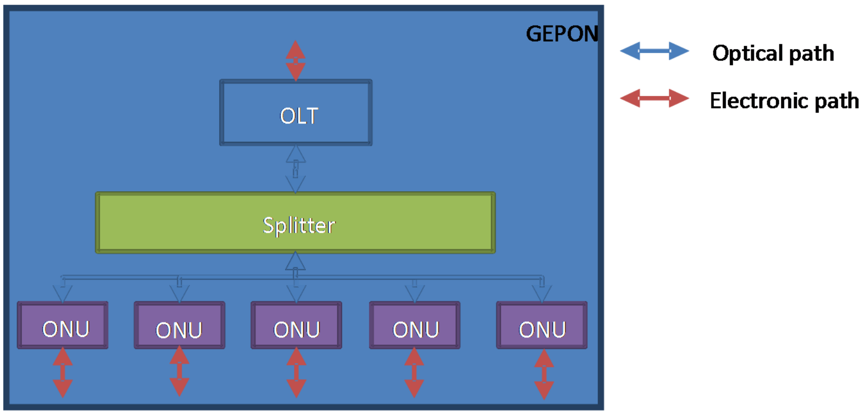


Figure 2.2 GEPON system

Figure 2.2 shows the GEPON which is the heart of the UCL testbed. The GEPON is described more fully in section 3. Ethernet input at the Optical Line terminal (OLT) is converted to an optical signal in a proprietary format. This is fed through a splitter which multiplexes the optical signal to all of the Optical Network Units (ONUs) which receive identical copies of the optical domain signal. Each ONU uses time dimension multiplexing to select which part of the signal is destined for that particular ONU. The signal is then converted back into the electronic domain by the ONU. In the UCL testbed the connections between the splitter and OLT and between the OLT and each ONU can be replaced by various lengths of optical fibre from a metre up to a kilometre. The ONU themselves can be terminated by various pieces of test equipment. The most likely experimental set up for the ALIEN project will be to use NetFPGA cards to provide test loads for the GEPON.

2.3 PSNC testbed

The PSNC network processor laboratory is designed for performing tests with network processors devices which could be fully programmed to perform various network roles. It contains one EZappliance device based on NP-3 network processor, three HP servers and 1GbE Ethernet Juniper switch implementing tagged connections. Currently there are 12 ports of EZappliance and 18 ports of servers connected to the Ethernet switch which allows changes to the network topology and traffic passed by EZappliance depending the tests or demonstrations to be performed. PSNC rack servers are powerful machines (e.g.: 2x6 core processor with hyper-trading, 24GB RAM, 2TB disk space, 6x 1GbE ports) for deploying virtual machines with any required software like network control plane or management plane components, network applications, software traffic generators, etc.

The PSNC network processor laboratory is a part of PL-LAB infrastructure. PL-LAB is a Polish nationwide network testbed designed to facilitate research in the area of Future Internet carried out within the Future Internet projects [PL-LAB]. It allows researchers to create virtual laboratories, with guaranteed resources and direct physical access to laboratory equipment. Depending on requirements, PSNC may reserve more network processor (e.g.: up to eight EZappliance boxes) devices using PL-LAB for a period of time which will allow to perform more complex test scenarios and demonstrations.

The PSNC network processor laboratory can be connected to pan-European GÉANT network using PIONIER network hub in Poznan (PIONIER is a Polish NREN network operated by PSNC).

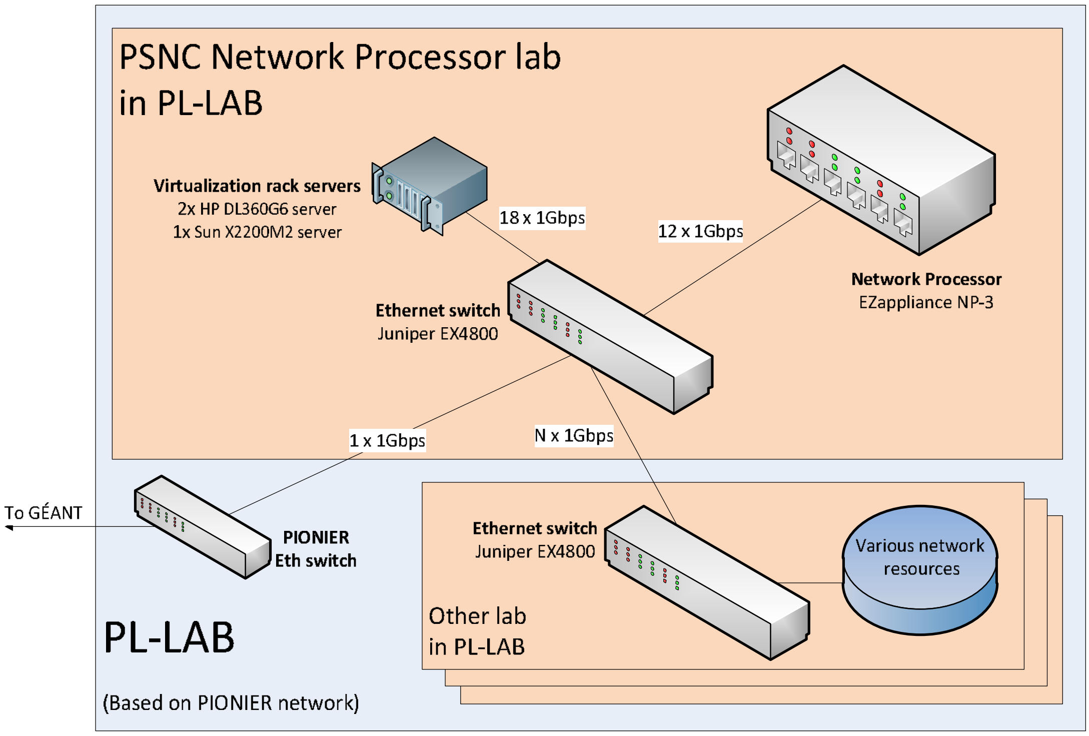


Figure 2.3 PSNC network processor laboratory

2.4 UPV/EHU testbed

The core of the UPV/EHU testbed is the EHU OpenFlow Enabled Facility (EHU-OEF). The EHU-OEF provides the researchers of the UPV/EHU with a shared infrastructure to test and validate their proposals in the area of networking. Several different experiments can be run simultaneously.

The OpenFlow enabled devices available at the EHU-OEF are seven NEC switches IP8800/S3640 OpenFlow enabled, two NetFPGA 1G cards and four OpenFlow-ready Wireless LAN Access Points (Pantou firmware). There are also two powerful machines devoted to deploying virtual machines (e.g. Sun Fire X4450 4xQuad-Core Intel Xeon, 64GB RAM, 1TB disk space, 4x 1GbE ports). Additionally, there is also measurement and analysis equipment with line-rate support for 1 Gbps, such as two Spirent Smartbit network analysers with GPS synchronization and a DAG capture board.

Regarding network processors devices, the UPV/EHU has recently acquired an EZappliance with a NP-3 network processor from EZchip and is currently involved in the purchase process of an ATCA-F140 40G AdvancedTCA Switch Blade with a Cavium Blade with two OCTEON CN5860 network processors.

Finally, there is an operational DOCSIS platform deployed in the access networks laboratory of the UPV/EHU. This equipment is connected to the EHU-OEF. The DOCSIS equipment is fully described in section 6. In brief, there are two different elements, the CMTS (1x Cisco uBR7246VXR Universal Broadband Router) and the cablemodem (12x Cisco Modem/EPC3825 GW EU-DOCSIS 3.0 802.11n).

The EHU-OEF is directly connected at 10 Gbps to the Spanish NREN (RedIRIS) and the Basque NREN (i2basque). The direct connection with RedIRIS gives the possibility to connect this infrastructure with i2CAT's OFELIA Island at layer 2, thus enabling the connection of the EHU-OEF to OFELIA.

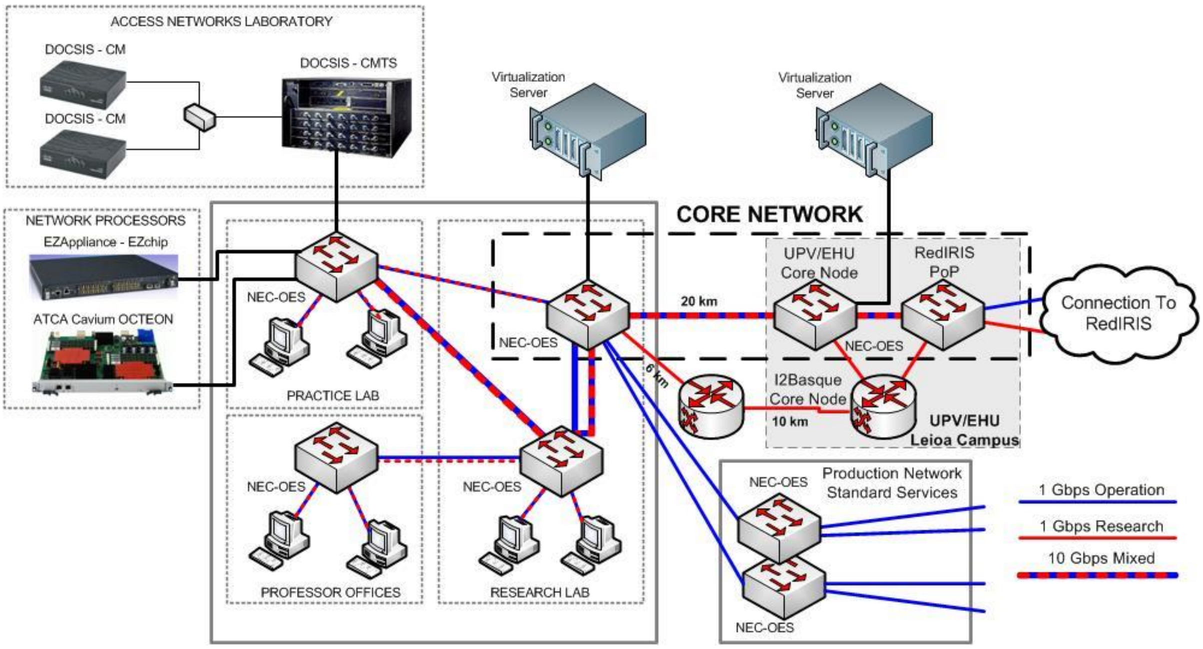


Figure 2.4 UPV/EHU OpenFlow Enabled Facility

2.5 EICT testbed

EICT's OpenFlow testbed serves two main purposes: (a) for testing upcoming releases of OFELIA's control framework, and (b) as a laboratory for hosting network hardware in the context of SDN/OpenFlow. Currently, the core is formed by a Kontron OM5080 ATCA compliant device and an Intel RS4000 server based on Intel's Crystal Forest platform. Based on the expedient framework initially developed at Stanford University, the OFELIA development team has significantly extended and improved the software resulting in the OFELIA control framework. All new major releases are tested within the EICT Island before entering production operation in other OFELIA islands.

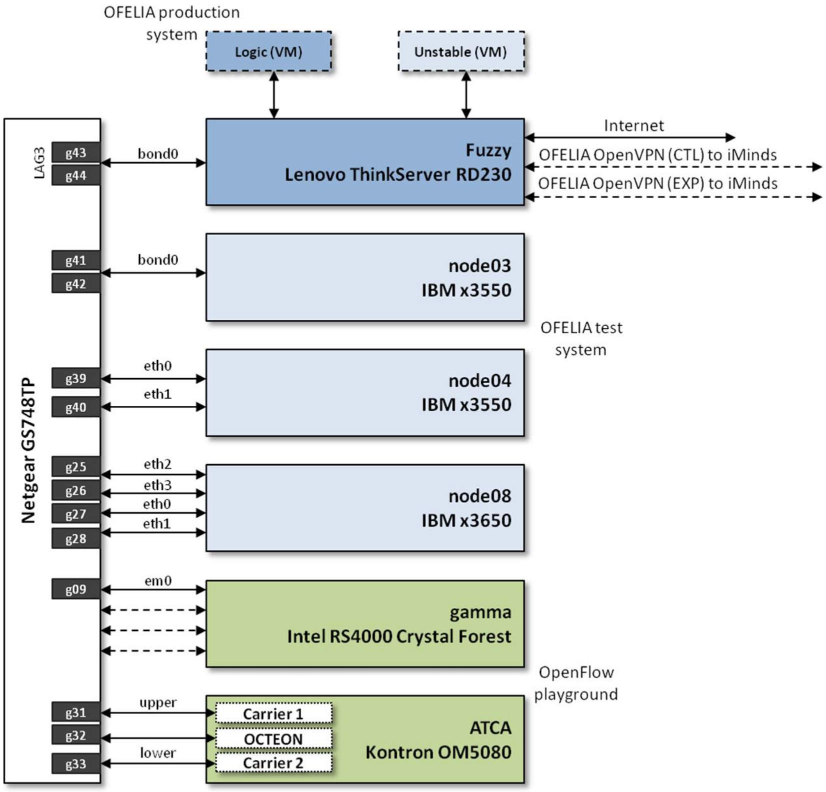


Figure 2.5 EICT's test environment

The current network consists of two evaluation boards of Broadcom, 48 port GbE switches with 4 HighGig uplink ports, several IBM x3650 servers for hosting virtual machines, and the ATCA platform described in section 7.

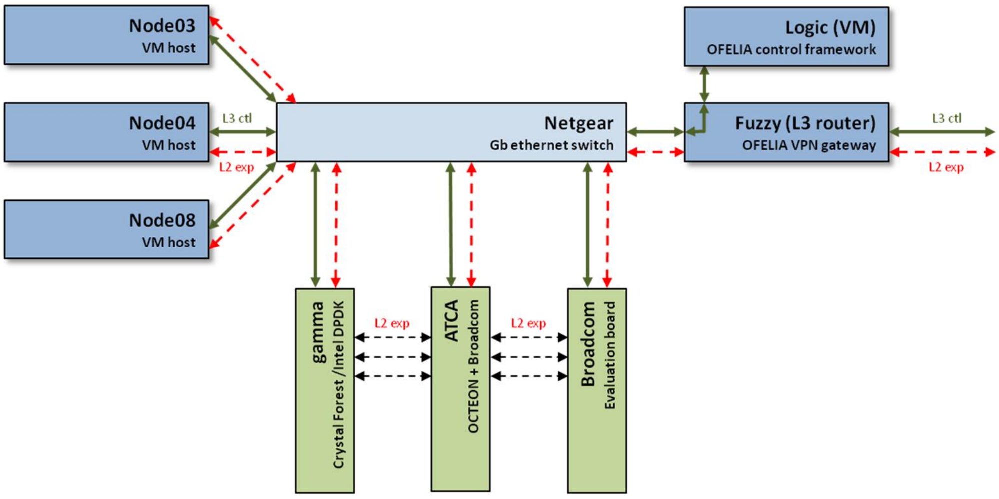


Figure 2.6 EICT testbed topology

2.6 Bristol testbed

University of Bristol has experimental facilities that cover several layers and technological domains of optical and high performance networking allowing for advanced experimentation on network infrastructure/services/ applications. The lab features diverse transmission capabilities up to 500 Gbit/s, full photonic elastic (space/frequency/time) switching, high-performance software/hardware FPGA network platforms, Carrier Grade Ethernet testbeds, IP routers, and OpenFlow switches. Analogously, feature-rich proprietary and standardized yet extended control/virtualization/management systems are developed to deliver network-wide intelligence. These include multi-technology enabled SDN (OpenFlow-based, FlowVisor) as well as GMPLS/PCE systems. Such systems incorporate protocols and algorithms to deliver optical infrastructure control and virtualization.

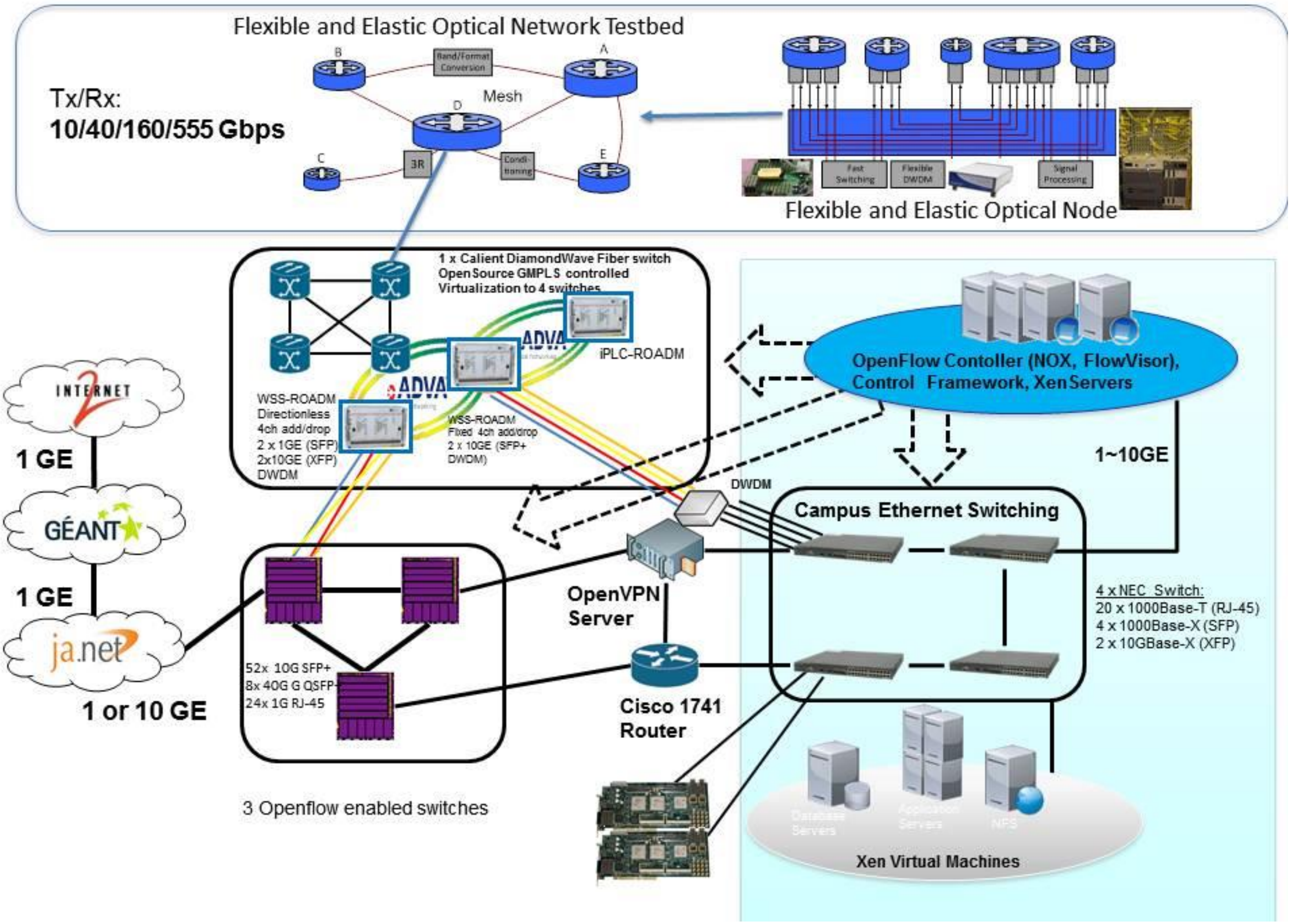


Figure 2.7 University of Bristol testbed

Architecture on demand and Multi-Dimensional Grid-less test-bed

- Architecture on-demand switching node
- Flexible Frequency/time/space transmission/switching
- High-speed Multi bit rate, multi format test-bed
- 160 GBPS OTDM, 74x10 GBPS, 15x40 GBPS RZ, NRZ
- Elastic bit rate, multi-format, and 555GBPS DMT, OFDM

Fixed/Flex Grid optical switching test-bed

- Four LambdaOpticalSystems LambdaNode2000 WDM switches
    - All optical wavelength/waveband switching
    - GE ports, Alien Lambda ports, SDH 1G & 10G Ports
- Three ADVA FSP-3000 ROADMs (GMPLS controlled, OpenFlow enabled)
- GMPLS controlled Calient FibreConnect DiamondWave switch
- 2x Spectrum Selective Switches (C Band, C+L Band), OpenFlow enabled

L2 OpenFlow enabled Carrier Grade and campus network

- 3x Carrier Grade Ethernet OpenFlow switches (ARISTA 7050S 64, Brocade CES 2000 & One Extreme Summit X650 supporting 1/10/40 G interfaces)
- 4x NEC IP8800 switch,
- L3 IP test-bed
- Juniper router emulator (software router), 6 TB storage for media content

Software/Hardware Defined Network

- 5x high-performance servers with 10GE interfaces for Virtualization and OpenFlow-based control framework
- OF controllers
  - NOX, FloodLight, SNAC
- Applications
  - FlowVisor, Virtualization algorithms for multi-technology integration (packet-based, Fixed-Grid and Flex-Grid WDM)
  - OpenVSwitch (soft switch)
- Software/Hardware-defined FPGA development platform/testbed
  - 3x Virtex 6-HXT, Virtex IV, 2x Virtex II-Pro
  - Network-on-Chip supporting dual L2 Eth. Frame – L1 TSON and hitless switch-over
  - Architecture on demand network on-chip and off-chip

The UNIVBRIS lab infrastructure is enhanced by the underlying network connectivity facilities that include: a dark fibre network connectivity; 1 and 10Gbps dedicated wavelength services over the JANET network; high speed dedicated connectivity over JANET, GEANT, GLIF and Internet2 to many research institutions into Europe, North/South America and Japan and Asia.

2.7 Dell/Force10 testbed

The Dell Force10 testbed will consist of two PowerConnect 7024 with a Split Data Plane module (SDPM) each. The Split Data Plane module will run a Linux Debian MIPS64. An external OpenFlow controller (FlowVisor / NOX or POX) running on the Dell Server PowerEdge to control the SPDM and or the PC7024 OpenFlow. Each SDPM has a serial console connection and external management Ethernet interface to allow full debugging and control.

The testbed is already configured to test different use case (encrypting flow, specific flow matching…) and the development environment is provided by the Cavium Software Development Kit.

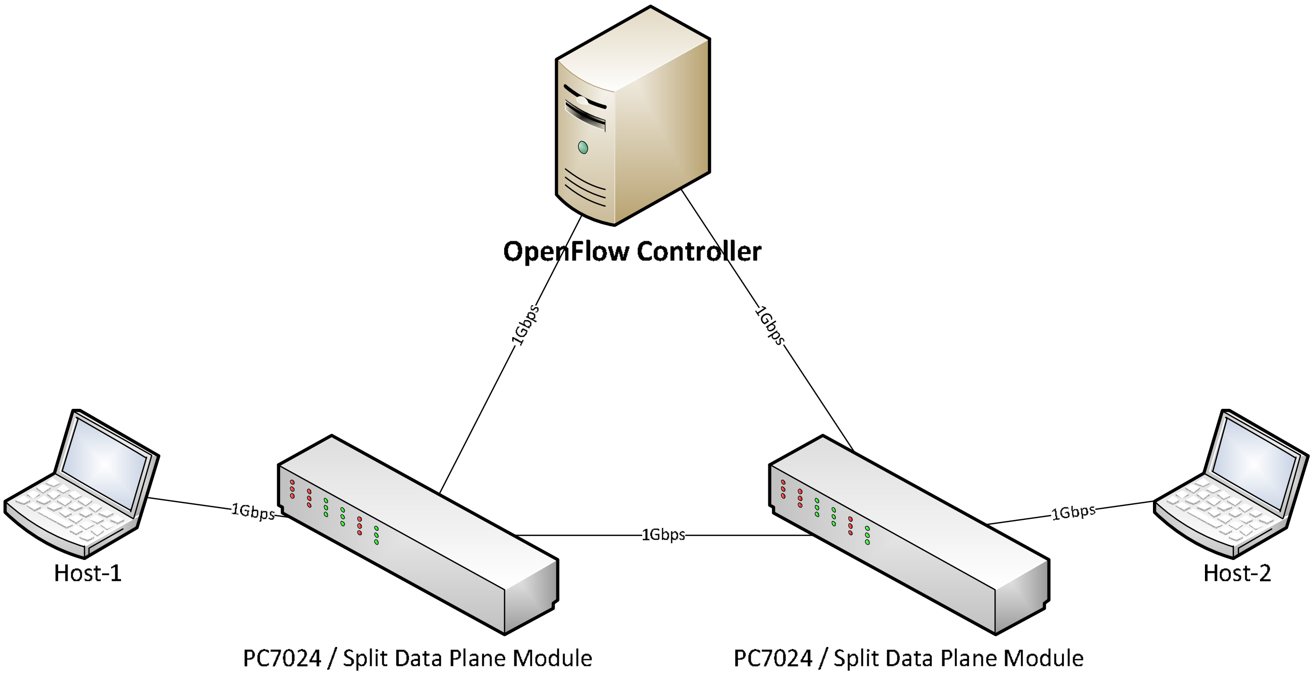

OpenFlow Controller
1Gbps
1Gbps
Host-1
1Gbps
1Gbps
1Gbps
Host-2
PC7024 / Split Data Plane Module
PC7024 / Split Data Plane Module

Figure 2.8 DELL/Force10 testbed

2.8 CREATE-NET testbed

The CREATE-NET OpenFlow Testbed is based on a geographically distributed facility located in the city of Trento. The facility is composed of three different locations interconnected through a 10Gbps dedicated fiber pair (max. link distance 8.6 km) in a ring topology. The three locations of the experimental facility are: CREATE-NET, University of Trento (Department of CSEE) and Trentino Network , a public regional network operator which is providing Internet connectivity and other telecommunication services to the Public Administration all over the region.

Figure 2.8 shows the physical topology of the CREATE-NET testbed. The island has three NEC switches model IP8800/S3640-24T2XW interconnected via XFP optical transceivers in a 10Gbit ring. The other equipment (two HP Procurve 3500 and four NetFPGA cards) are interconnected using 1Gbit Ethernet interfaces. All the OF-enabled switches are OF 1.0 compliant. As depicted, the island is split into three sub-islands each of them located at different sites of the testbed and including three OpenFlow switches and one server hosting users VMs.

The management servers (OFELIA Control Framework, FlowVisor, NFS, SMB, Monitoring, Backup, ProxMox Virtual Environment) are hosted by the CREATE-NET server farm. Each OpenFlow switch acts as a hybrid switch: half of the ports are OpenFlow Controller, and half are directly managed by the standard protocols programmed in the firmware. The sub-islands are interconnected both via a dedicated fiber link (for the OpenFlow traffic) and using QinQ over an MPLS network for management purposes (to allow out-of-bound connection between the switches and the controller). Besides the OpenFlow network, the island provides also these networks that follow the OFELIA addressing convention:

- Control network (VLAN 4094 – NET 10.216.32.0/22)
- Management network (VLAN 4093 – NET 10.216.160.0/22)

- Internet network (VLAN 4092 – NET 172.16.0.0/24)

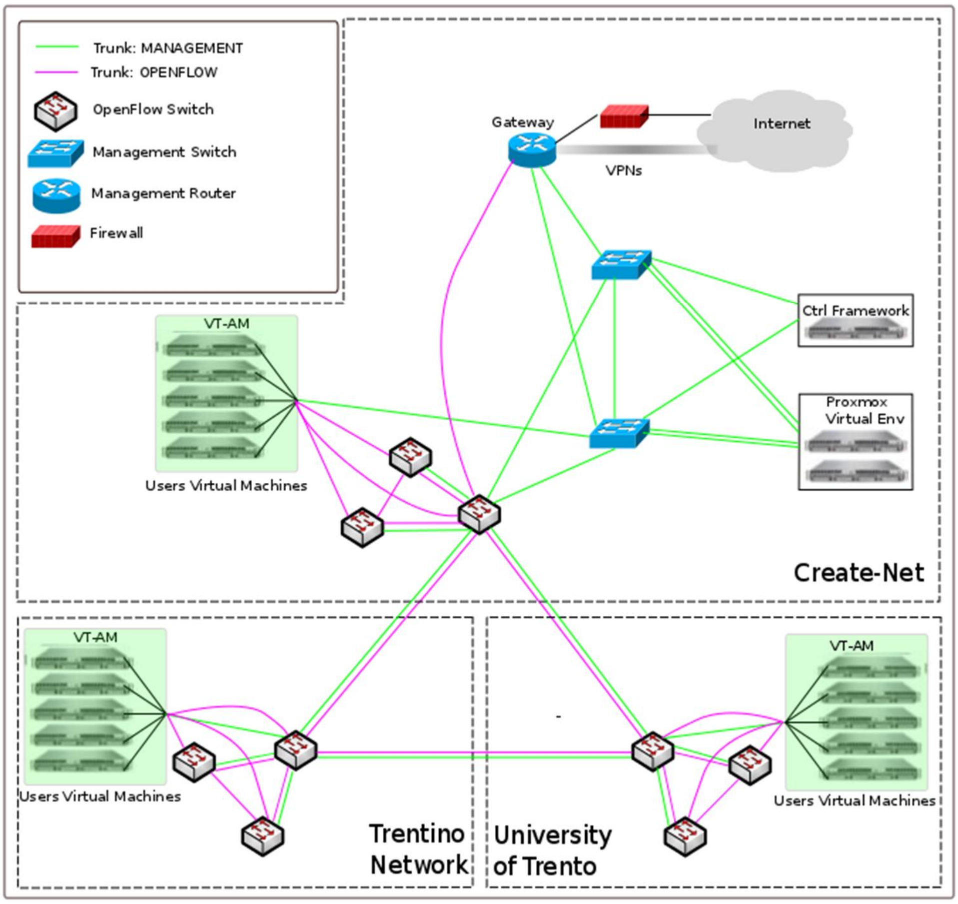


Figure 2.9 CREATE-NET island topology

Below the list of the entire inventory in the testbed, separated in OpenFlow hardware, non-OpenFlow (management) network equipment and Servers.

The Open Flow switches consist of:

- 3x NEC IP8800/S3640-24T2XW
- 2xHP ProCurve 3500
- 2xHost PC each with 2xNetFPGA cards (each NetFPGA has 4x1G ports)

The networking equipment includes:

- 3x Dell Switch power connect 5324
- 1x Dell Router (Vyatta Linux distro) working as a gateway, firewall, OSPF ABR and VPN endpoint

The servers include:

- 3x servers running Debian Squeeze 32-bit
- 1x server running Vyatta Core 6.4
- 3x servers running Debian Squeeze 64-bit

3 EZappliance

3.1 Overview

The EZappliance network device is designed and produced by EZchip Technologies [EZchip], a company from Israel. It is a compact hardware platform for deploying network applications that provides a complete data plane and control plane solution. This device is designed for easy development and deployment of new efficient network applications. EZchip's NP-3 EZappliance platform is based on EZchip's NP-3 30-Gigabit network processor [EZapp]. This specialized network processor is a fully-programmable entity which enables flexible processing, parsing, classification, packet header manipulation, switching and management of pass through packets. The NP-3 integrated traffic manager (TM) provides wire-speed QoS functionality.

EZchip Technologies provides a broad library featuring tested source code for a wide range of applications, such as L2 switching, Q-in-Q, PBT, T-MPLS, VPLS, MPLS, IPv4/IPv6 routing, firewall and IDS. Such libraries enable implementation of different network applications like carrier Ethernet access and metro switches/routers, traffic management, security, gateways, network monitoring and traffic analysis [EZapp].

3.1.1 Physical box overview

EZappliance is 1U system which offers twenty-four 1-Gigabit Ethernet and two 10-Gigabit Ethernet ports (Figure 3.1).

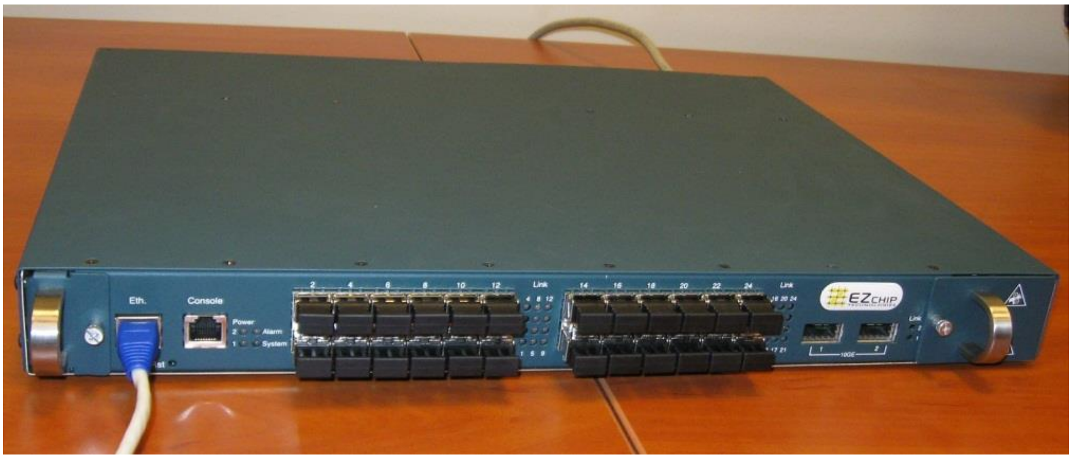


Figure 3.1 EZappliance network device from EZchip Technologies

Beside the main board with an NP-3 network processor, this device is equipped with a HostCPU embedded system with a pre-installed ELDK Linux system. HostCPU is based on the PowerQUICK™ III 800MHz general purpose processor with PowerPC architecture. The combination of the EZchip network processor and a general purpose CPU provides a complete data plane and control plane solution. NP-3 and Host CPU are interconnected by a PCI bus with throughput around 2Gbps. The NP-3 API routines and dedicated, pre-installed software on the HostCPU allow easy on-the-fly control of network processor actions. A data plane based on the NP-3 network processor with specialized Task Optimized Processors (TOPs) and a control plane based on HostCPU board with Linux OS are described in detail in sections 3.2 and 3.3 respectively.

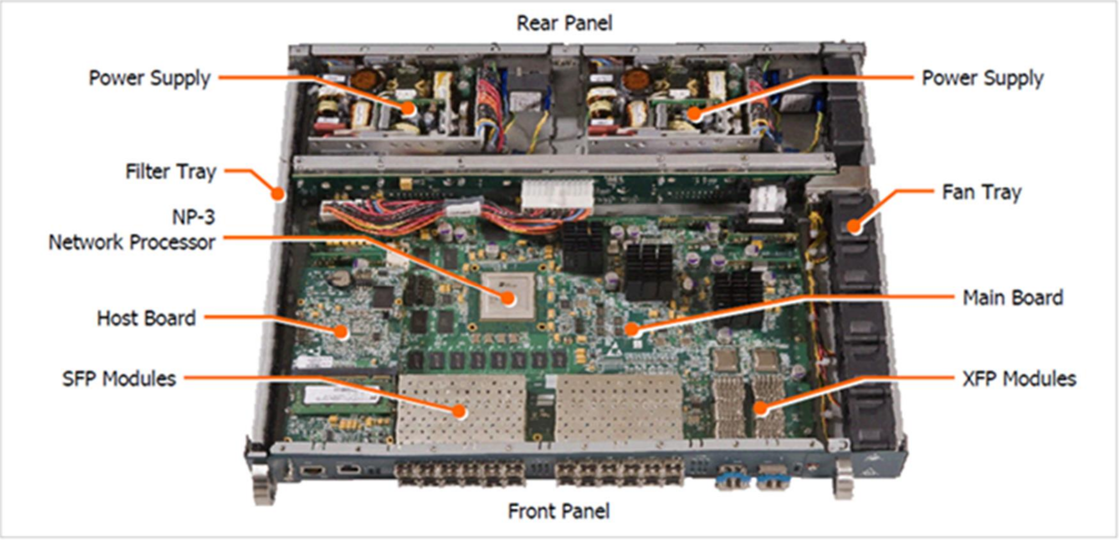
The EZappliance device houses the following external interfaces:

- 24 x 1GbE SFP ports
- 2 x 10GbE XFP ports
- 10/100/1000 Ethernet and RS232 management ports

The HostCPU board is equipped with:

- Freescale PowerPC 800MHz CPU
- Linux ELDK 2.6.24-EZ.1
- 512MB RAM
- 64MB + 192MB flash memory
- PCI bus to NP-3 processor (up to 2Gbps)
- One 10/100/1000 Ethernet port

The internal system layout is depicted in Figure 3.2. Main modules like the main board with NP-3 processor and SFP and XFP modules, Host Board as well as Fan Tray, Filter Tray and two hot-swappable Power Supplies are shown.



SFP - Small Factor Pluggable XFP - 10 Gigabit Small Factor Pluggable

Figure 3.2 EZappliance - Internal system layout [EZapp]

3.2 Data plane

3.2.1 Transmission technology

The data plane part of the EZappliance consists of the NP-3 network processor with associated memory and Ethernet MACs (Vitesse VSC7344) which means that the NP-3 is able to process Ethernet-based frames only. The data plane programmer is responsible for configuration and programming of all internal NP-3 entities to handle Ethernet frames together with all higher layers network headers and payloads.

Key components of NP-3 processor are five pipelined Task Optimized Processors (TOPs) and Traffic Manager (TM) as depicted on Figure 3.3. Each TOP processor is specialized in performing specific operations over processed Ethernet frames. All TOPs except TOPsearch II are programmable in EZchip assembler with different instruction sets for each TOP. The data plane programmer must implement four separate programs for four programmable TOPs and load them into the NP-3 memory.

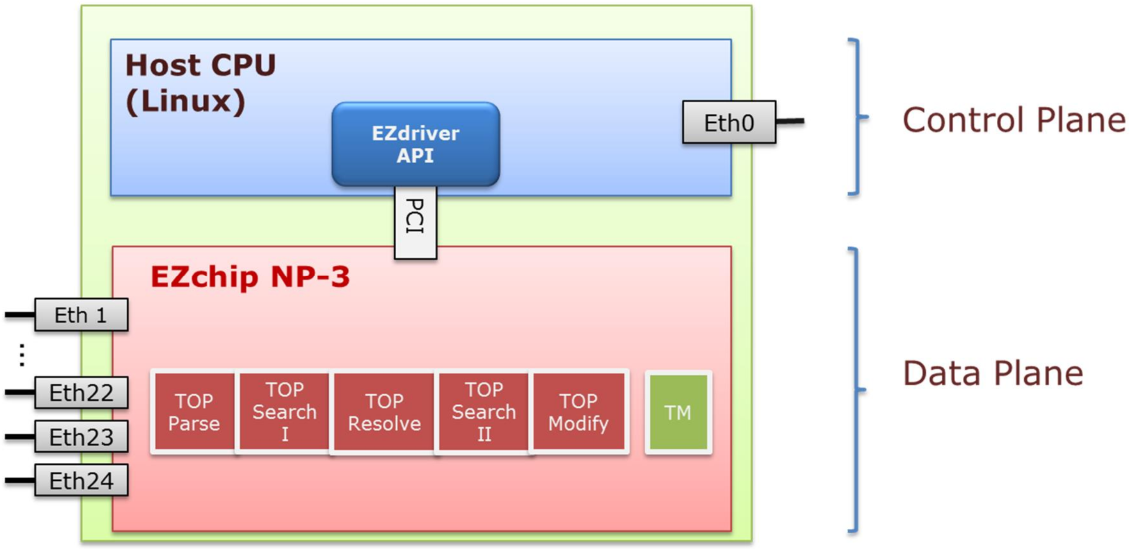


Figure 3.3 EZappliance – logical architecture [EZapp]

TOPparse is designed for decoding, parsing and analysing received frames. It prepares searching keys for TOPsearch I and can pass message with some additional data to TOPresolve. The TOPparse instruction memory can contain up to 9000 instructions. This TOP has access to the entire processed frame.

TOPsearch I is designed for performing search operations on defined search structures (described in 3.2.2). Searching keys are prepared by the previous TOP, but there is also a possibility to perform compound searches in multiple structures with optional assembler code. This TOP can be programmed with up to 256 instructions and have no access to a processed frame data. It is primarily a search TOP.

TOPresolve has the main task of taking routing/forwarding decisions based on TOPsearch I results and messages from TOPparse. It can be programmed with up to 9000 instructions and have additional logical instructions. The outputs from this TOP are search keys for TOPsearch II and messages to TOPmodify. It has no access to processed frame data.

TOPsearch II is a second search TOP. It has limited possibilities in relation to TOPsearch I. It has no instruction memory so it is the only non-programmable TOP in NP-3 pipeline.

TOPmodify is designed for performing operations on a frame body according to TOPsearch II results and messages from TOPresolve. Each byte in a frame can be inserted, cut or overwritten. After processing a prepared frame is sent to the proper output queue in the Traffic Manager.

Traffic Manager is a configurable entity responsible for traffic shaping and scheduling that is located outside the TOPs. Its tree structure allows configuration of up to 64K "leaf" queues. Frames from each queue are sent by proper physical or virtual ports. Virtual ports gives the possibility of sending frames to the Host CPU via PCI or to reprocessing via internal loopback.

3.2.2 Switching management rules

The primary ways to configure the behaviour of the data plane implementation on NP-3 are search structures. A data plane programmer defines which search structures should exist within NP-3 and specifies the purpose and size of each search structure. The usage of search structures is coded within NP-3 assembler programs. Search structures can be accessed from TOPparse, TOPsearch I and TOPsearch II. Entries in all defined structures can be managed by Host CPU. EZappliance have totally 128Mbytes of search memory which can be divided between all defined search structures.

EZChip NP-3 network processor has some constrains in assigning search structures to TOPs. Up to 4 Direct Tables and Hash Tables can be accessed by TOPparse processor but search structures are used generally by TOPsearch processors. TOPsearch I can have access to 64 defined structures and TOPsearch II to 8 structures. Each search structure can be also aliased to the same TOP or another one. The usage of Ternary Content Addressable Memory (TCAM) allows for the addition of key masking in tree structures. In each entry in a tree structure bits of key field can be one, zero or “don't care”. Detailed description of available search structures is presented in Table 3-1.

Structure type	**Key size [B]**	**Result size [B]**	**Structure type**	**Key size [B]**	**Result size [B]**
TOPsearch I structures			TOPsearch II structures		
Direct Access Tables	1-4	8, 16, 32, 64	Direct Access Tables	1-4	8, 16, 32
Hash Tables	1-48	8, 16, 32, 64	Hash Tables	1-32	8, 16, 32
Trees	1-16, 38	1, 3, 8, 16, 32, 64, 96	TOPparse structures		
Linked: Hash + Tree	1-38	8, 16, 32, 64, 96	Direct Access Tables	1-3	8, 16, 32
Linked: Tree + Hash	4	8, 16, 32, 64	Hash Tables	4, 8, 12	8, 16, 24, 28

Table 3-1 Search structures in NP-3

Thanks to freedom in the implementing frame processing mechanisms and defining all supporting data structures, in the NP-3 there can be implemented a very broad set of switching rules. Incoming Ethernet frames can be switched to any EZappliance data plane port or sent to HostCPU depending on any combinations of protocol headers fields values or offset bytes values in a processed frame. The NP-3 processor allows also for looping and multiple passing of the same frame through TOPs pipeline which give even more possibilities.

3.3 Control/management plane

3.3.1 Plane purpose

The control/management plane does not exist in an EZappliance device unless it is installed by a device programmer. The EZappliance device is designed in such way that control plane or management plane software should be installed within a HostCPU Linux system. In a host system, there is pre-installed by default special software for accessing the forwarding plane named EZdriver API library, which must be used by any developed software which will interact with the data plane part of the device (NP-3 processor). The EZdriver API library allows installation of assembler programs for each TOP processor, management of TOP programs execution environment and TM modules configuration.

A more detailed description of EZdriver API library is presented in Table 3-2.

EZdriver API functionality	Managed element of NP-3	Description
Data structures access	TopParse TopSearch	Operates on data structures initialized within NP-3 network processor. There is offered full access to read entries from structures, add new entries, modify existing entries or delete entries.
Memory/registers access	CREG	Allows to read from and write a new values to registers of NP-3 processor.
TM configuration	WFQ Shaping WRED Queue priority	Allows for configuration of Traffic Manager components. Various traffic management mechanism could be configured with proper parameterization.
Statistics access	Counters	Operates on building counters. State of counter can be set, reset and read.
Frame access	Frame memory	Allows for passing frames between NP-3 and HostCPU system. Frame generated with Host system can be sent via data plane port as well as frame received at data plane input port can be forwarded to HostCPU system.

Table 3-2 Most important NP-3 API functions available in EZdriver API library

Most fundamental NP-3 API functions are memory data structure operations within TopParse and TopSearch processors and operations within CREG and CAM registers. These data structures are used for the creation of routing or switching tables and storing some configuration values. Frame access functionality provided by EZdriver API library is very important for control plane components installed within Host system because allows to exchange control plane messages via data plane connections. A management agent installed within the host system can utilize statistics and counter access for any monitoring requirements. Both control plane and management plane systems can use TM configuration to implement different traffic conditioning and QoS mechanisms within the device.

The EZdriver API is provided in form of C headers files which must be included in any developed C program. In order to use the EZdriver API functionality, an EZdriver API object must be instantiated within application and this object will provide access to the NP-3 chip functionality. Instantiation of the EZdriver API object is quite complex. The workflow includes initialization of the NP-3 host environment, memory partition, loading the prepared code and running it with a valid configuration for the NP-3 chip.

3.3.2 Exposed protocols

By default EZappliance device doesn't have any protocols installed within the host system. The host system allows for installation of additionally developed software which can handle any communication, signalization or routing protocols required by the role performed by the device.

3.3.3 Configuration requirements

An IP address of eth0 interface within the host system must be configured in order to allow installation of forwarding and control plane software. This IP address can be used as an end-point of the signaling control network (SCN) or the management network which allows cooperation between various control/management plane modules.

Additional configuration requirements depends on the role performed when a concrete frame forwarding and control plane software is installed within EZappliance.

4 GEPON

4.1 Overview

The GEPON at UCL consists of several components as shown in Figure 2.2. The whole GEPON system consists of the OLT, a passive optical splitter and several ONU. The OLT is the “intelligent” part of the system and in the normal deployment is the part which is connected to the WAN. It is a point-to-multipoint device and in normal deployment is connected to a splitter which multiplexes the signal to the ONU which are typically situated in end user premises. The exact product numbers are as follows:

- OLT: Planet technologies EPL-1000 http://www.planet.com.tw/en/product/product.php?id=25817
- Splitter: Planet technologies EPL-SPT-32
- ONU: Planet technologies EPN-102 http://www.planet.com.tw/en/product/product.php?id=22826

The OLT can sustain 1.25Gbps both upstream and downstream as can the ONU.

4.1.1 Physical box overview

The OLT has dimensions 432 x 207 x 43mm, taking up 1U of rackspace. It houses the following ports:

- Uplink: 1 x Gigabit TP / SFP Combo Port (10/100/1000Base-T / SFP GbE)
- PON Port: 1 x PON Port (IEEE 802.3ah)
- Console Port: RS-232 Serial Port (9600, 8, N, 1)
- Management Port: 1 x RJ-45 (10/100Base-TX)

The main processor controlling the OLT is the Teknovus TK3721 ASIC System-on-chip MAC controller.

Figure 4.1 The EPL-1000 OLT

The ONU has dimensions 176 x 124 x 35 mm (it is end user equipment not designed for rack mounting). It houses the following ports:

- LAN: 1 x 10/100Base-TX, Auto-Negotiation, Auto MDI/MDI-X
- LAN: 1 x 10/100/1000Base-TX, Auto-Negotiation, Auto MDI/MDI-X
- WAN: 1 (1.25G EPON interface with SC type connector, 1000Base-PX-20)

Figure 4.2 EPN 102 ONU

4.2 Data plane

4.2.1 Transmission technology

The data plane for the GEPON presents externally as a layer 2 Ethernet switch with 1GbE (10/100/1000Base-T / SFP GbE IEEE 802.3u) at the OLT and ONUs. Internally it is compliant with IEEE 802.3z (Ethernet over Fiber-Optic at 1 Gbit/s) and IEEE 802.3ah (Ethernet for the last mile) including Forward Error Correction support and an Operations Administration and Maintenance (OAM) protocol based on IEEE 802.3ah. It includes Dynamic Bandwidth Allocation (DBA) support and SLA systems. The system supports IPv4 and IPv6 packets and up to 4,000 MAC addresses. The GPON can allocate up to 4,096 VLANs.

Because of the passive nature of the splitter, this is an important bottleneck in the system. Transmission from the OLT to the ONU is one-to-many with an ONU selecting the data intended for it using time-domain multiplexing (TDM) techniques. This means that, while all ONU are capable of 1Gbps transmission this is not full mesh and, in fact, one ONU transmitting at 1Gbps whether to the external network (beyond the OLT) or to another ONU will completely fill the capacity of the GEPON. All transmission from ONU to ONU must be via the OLT (Because of the nature of the normal deployment of the system, transmission from ONU to ONU is not the usual mode and the majority of data will be between ONU and the external network.)

Transmission between the OLT, splitter and ONU is optical and entirely passive in nature. The system supports a split of up to 1:32 (that is 32 ONU) and is specified to use fibre lengths of up to 20km (which have been tested in our testbed). The system uses wavelength of 1310nm upstream and 1490nm downstream. The optical encoding uses IEEE 802.3ah (Ethernet in the first mile) and IEEE 802.3z (Ethernet over fibre optic) which implies conformance to 1000Base-PX10 or 1000Base-PX20. In particular, for point-to-multipoint applications IEEE 802.3ah introduces a family of signalling systems derived from 1000Base-X but with extensions to the Reconciliation (RS), Physical Coding (PCS), and Physical Medium Attachment (PMA) sublayers. In addition, IEEE 802.3ah includes FEC capability (Clause 60). In very general terms, upstream and downstream channels are wavelength duplexed, and traffic between ONUs and the OLT uses time division duplexing.

The OLT controls an ONU's transmission by the assigning of grants. The transmitting window of an ONU is indicated in GATE message where start time and length are specified. The point-to-multipoint (P2MP) transmission from OLT to ONUs mean that the equipment might not be seen as a switch but the entire GEPON regarded together can be thought of as a distributed switch. Access is controlled by Multi-Point MAC Control Protocol (MPCP). A point-to-point emulation sub layer assigns a Logical Link Identifier (LLID). The system supports multiple MAC addresses with several at the OLT (at least one for each ONU) and one at each ONU. The purpose of the MPCP is to allow point-to-point emulation using transmission that is naturally broadcast in nature (via the splitter). Multiple MACs operate on a shared medium by allowing only a single MAC to transmit upstream (from ONU to OLT) at any given time across the network using a time-division multiple access (TDMA) method. An LLID associates a MAC at an ONU to a MAC at the OLT and they are prepended to the data frame. In addition to such unicast LLIDs, broadcast LLIDs are used. New ONUs are automatically discovered and assigned a MAC at the ONU and OLT and an LLID binding them.

4.2.2 Switching management rules

The switching management for the GEPON is controlled by the OLT. Rules for filtering (dropping packets) can be created which match against MAC, VLAN or LLID identifiers. These rules can be AND rules only although several are allowed. Rules can also be created for classifying traffic according to similar identifiers. These rules can be used to place traffic into different priority queues or to different links. The configuration and filtering rules are set at the OLT but can be pushed out to the ONUs.

VLANs can be configured following the IEEE 802.1q protocol. The VLANs can connect the OLT to one or more ONUs and this allows the possibility of multicast at the GEPON.

The OLT can configure filtering and classification rules based upon MAC addresses or VLAN identifiers. These rules can The OLT can also be configured as a layer 3 bridge and, hence, learn IPv4 destination addresses.

4.3 Control/management plane

4.3.1 Plane purpose

The GEPON is primarily managed at the OLT and the management system is based upon the Teknovus TK3721 chipset developed by Teknovus but now owned by Broadcom. The OLT is responsible for creating and destroying logical links between ONU, creating and maintaining SLAs (IEEE 802.1p), VLANs (802.1q) and other packet transmission rules. Traffic classification can involve layer 2, 3 and 4 rules. The operation of P2MP network is asymmetrical, with the OLT assuming the role of master, and the ONUs assuming the role of slaves.

The control plane can assign various algorithms to change the rate of the PON Traffic. The device uses a hierarchical weighted round robin algorithm for Dynamic Bandwidth Allocation. This allows, for example a shaper which restricts user traffic to enforce SLAs. Maximum and minimum bandwidths for an ONU to OLT connection can be specified. The DBA is implemented by changing the time slots granted to an LLID.

Port management can be used for several actions. Ports can be set to loop-back. VLANs can be allocated to single or multiple LLIDs (allowing multicast). ONUs can be cross connected to allow a logical connection directly between ONUs. A link from OLT to ONU can be set as a network bridge. Links can also be blocked, cutting an ONU off from the OLT. The OLT can use filtering rules to classify traffic. The OLT can also be configured as a layer 3 bridge.

4.3.2 Exposed protocols

The control channel between the OLT and the ONU is based on the standard IEEE 802.3ah Operation, Administration and Management (OAM) protocol (Clause 57). Architecturally, the OAM sublayer is an optional sublayer placed between the Media Access Control (MAC) and the Logical Link Control (LLC) sublayers within IEEE 802.3ah, and has the following objectives:

1. *Remote Failure Indication*. This allows the OLT to verify whether a particular ONU is operational.
2. *Remote Loopback*. This allows for traffic sent to the ONUs to be returned to the OLT for error testing.
3. *Link Monitoring*. This provides a channel for event notification that allows the inclusion of diagnostic information, as well as providing a mechanism for polling variables within the IEEE 203.3a MIB (Clause 30).
4. *Miscellaneous*. This allows the OLT to query the ONUs for capability discovery, as well as to allow proprietary higher layer management applications.

Control requests to the PON must be made via one of the two management ports of the OLT. A console port and a management port are available. The protocols are proprietary and specific to the TK3721 ASIC.

The console port is an RS-232 serial port and accepts text-style commands via a standard terminal application. A standard console terminal can send commands using a slash delimited command line interface. Commands must be sent “spaced out” to not overwhelm the send/receive rate of the serial port. Example commands are:

/pers/mgmtip <IP address> -- change the IP address of the management port.

/oam/dump <LLID> -- dump OAM statistics for the specified LLID.

The management port is an RJ-45 (10/100Base-TX). It accepts commands via a closed source proprietary GUI. This communicates the commands using UDP and a proprietary message format.

4.3.3 Configuration requirements

As previously stated, the GEPON can be configured via the console port (using the command line via a terminal window) or via a PC the management port using a GUI. The GUI is a windows based program and it must be run on a PC with an IP address matching that known for the management port by the OLT. This can be reset in the OLT using the serial port and a command of the form:

```
pers/hostip www.xxx.yyy.zzz
```

5 NetFPGA

5.1 The NetFPGA Card Overview

The NetFPGA card is an extension board for a PC (see Figure 5.1). It is connected with the PC via the PCI bus. It was designed and developed by researchers, students and engineers from Stanford University and the University of Cambridge. They created the NetFPGA groups and run the NetFPGA project [netfpga]. They prepared a lot of interesting examples and codes, most of them are available for free at website [netfpga].

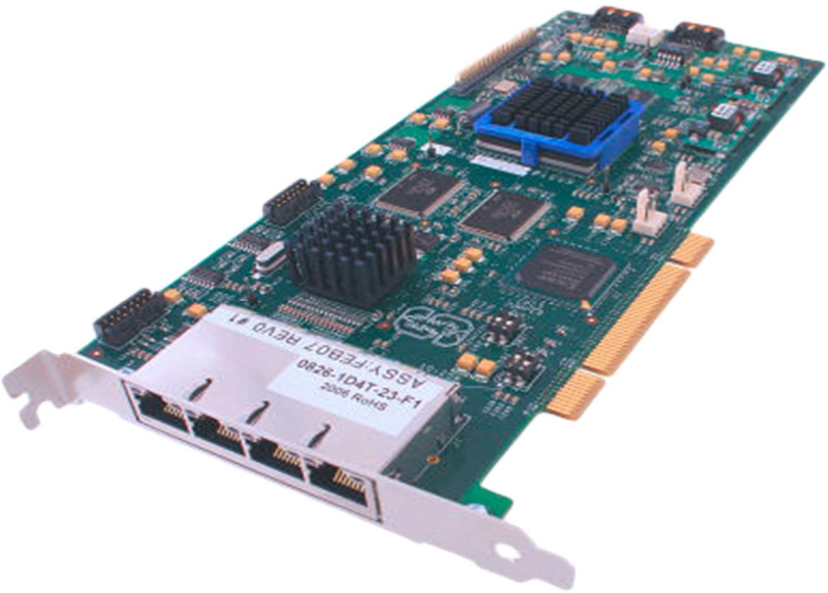


Figure 5.1 The NetFPGA card

The NetFPGA can be treated as a low-cost hardware platform for running experimental networking devices. It was primarily designed as a tool for teaching networking hardware and router design. It has also proved to be a useful tool for networking researchers. Through partnerships and donations from sponsors of the project, the NetFPGA is widely available to students, teachers, researchers, and anyone else who is interested in experimenting with new ideas in high-speed networking hardware. It allows use of reference projects (IPv4router, Ethernet switch or Network Interface Card) or creation of its own prototypes and projects.

A detailed description of the NetFPGA's operation is available in the official guide [netfpgaguide]. The NetFPGA codebase is open-source. Details about the license are described on-line. Researchers and students are free to use and modify the NetFPGA hardware and software code as they see fit. It is possible to directly use or modify prepared reference projects or prepare new ones.

5.1.1 Physical Box Overview

At a high level, the board contains four 1 Gigabit/second Ethernet (GigE) interfaces (they can be seen in Figure 5.1), a user programmable Field Programmable Gate Array (FPGA), and four banks of locally-attached Static and Dynamic Random Access Memory (SRAM and DRAM). It has a standard PCI interface allowing it to be connected to a desktop PC or a server. A reference design can be downloaded from the project website [netfpga] which contains a hardware-accelerated Network Interface Card (NIC) or an Internet Protocol Version 4 (IPv4) router that can be readily configured into the NetFPGA hardware. The router kit allows the NetFPGA to interoperate with other IPv4 routers.

The NetFPGA platform contains one large Xilinx Virtex2-Pro 50 FPGA. It is the main chip of this board. It should be programmed with user-defined logic, its core clock runs at 125MHz. The NetFPGA platform also contains next small Xilinx Spartan II FPGA holding the logic that implements the control logic for the PCI interface to the host processor.

The board contains also memory banks. They are realized by two 18 MBit external Cypress SRAMs, they are arranged in a configuration of 512k words by 36 bits (4.5 Mbytes total) and operate synchronously with the FPGA logic at the same frequency – 125 MHz. One bank of external Micron DDR2 SDRAM is arranged in a configuration of 16M words by 32 bits (64 MBytes total).

The Broadcom BCM5464SR Gigabit/second external physical-layer transceiver (PHY) sends/receives packets over/from standard category 5, 5e, or 6 twisted-pair cables. The four PHY interfaces with four Gigabit Ethernet Media Access Controllers (MACs) instantiated as a soft core on the FPGA. The NetFPGA also includes two interfaces with Serial ATA (SATA) connectors that enable multiple NetFPGA boards in a system to exchange traffic directly without use the PCI bus.

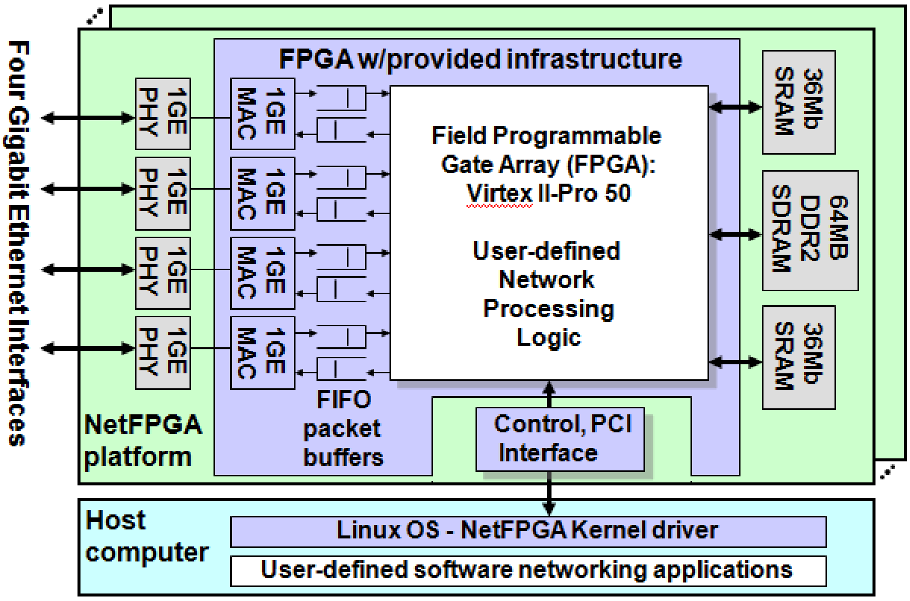

Four Gigabit Ethernet Interfaces
1GE PHY
1GE PHY
1GE PHY
1GE PHY
1GE MAC
1GE MAC
1GE MAC
1GE MAC
FPGA w/provided infrastructure
Field Programmable Gate Array (FPGA): Virtex II-Pro 50
User-defined Network Processing Logic
FIFO packet buffers
Control, PCI Interface
36Mb SRAM
64MB DDR2 SDRAM
36Mb SRAM
NetFPGA platform
Host computer
Linux OS - NetFPGA Kernel driver
User-defined software networking applications

Figure 5.2 Logical structure of the NetFPGA card

The project designated to be run on the NetFPGA card contains two main parts – the software part and the hardware part. The software is typical software which interacts with the operating system of a PC. It also contains a typical driver for hardware. The hardware part of code is run in Xilinx Virtex II FPGA chip, which is the main chip of the card. It processes all the data from input queues and implements the program for the hardware part. Both parts cooperate via PCI bus with kernel driver and registers [netfpgaregisters].

The NetFPGA offloads processing from a host processor. The host's CPU has access to main memory and can DMA to read and write registers and memories on the NetFPGA. Unlike other open-source projects, the NetFPGA provides a hardware-accelerated hardware datapath. The NetFPGA provides a direct hardware interface connected to four GigE ports and multiple banks of local memory installed on the card.

NetFPGA packages (NFPs) are available that contain source code (both for hardware and software) that implement networking functions. Using the reference router as an example, there are three main ways that a developer can use the NFP. In the first usage model, the default router hardware can be configured into the FPGA and the software can be modified to implement a custom protocol.

Another way to modify the NetFPGA is to start with the reference router and extend the design with a custom user module. Finally, it is also possible to implement a completely new design where the user can place their own logic and data processing functions directly into the FPGA.

The hardware is used as an accelerator and then the software must be used to implement new protocols. In this scenario, the NetFPGA board is programmed with IPv4 hardware and the Linux host uses the Router Kit Software distributed in the NFP. The Router Kit daemon mirrors the routing table and ARP cache from software to the tables in the hardware allowing for IPv4 routing at line rate. The user can modify Linux to implement new protocols and test them using the full system.

Starting with the provided hardware from the official NFP (or from a third-party NFP), this can be modified using modules from the NFP's library or by writing your own Verilog code. The source code is then compiled using industry standard design tools to a bit file. The implemented bitfile can then be downloaded to the FPGA. The new functionality can be complemented by additional software or modifications to the existing software. For the IPv4 router, an example of this would be implementing a Trie longest prefix match (LPM) lookup instead of the currently implemented CAM LPM lookup for the hardware routing table. Another example would be to modify the router to implement NAT or a firewall.

5.2 Data Plane

5.2.1 Transmission technology

The main assumption was to implement the data plane in the hardware part, i.e. the data will be processed only by the hardware chip, which is the FPGA. It is possible to use clean state design, i.e. the user has to develop their own logic and data path for internal structure of FPGA chip. But it is much better to use a prepared reference pipeline and yourself develop only the additional functionality. Figure 5.3 shows the difference between two approaches.

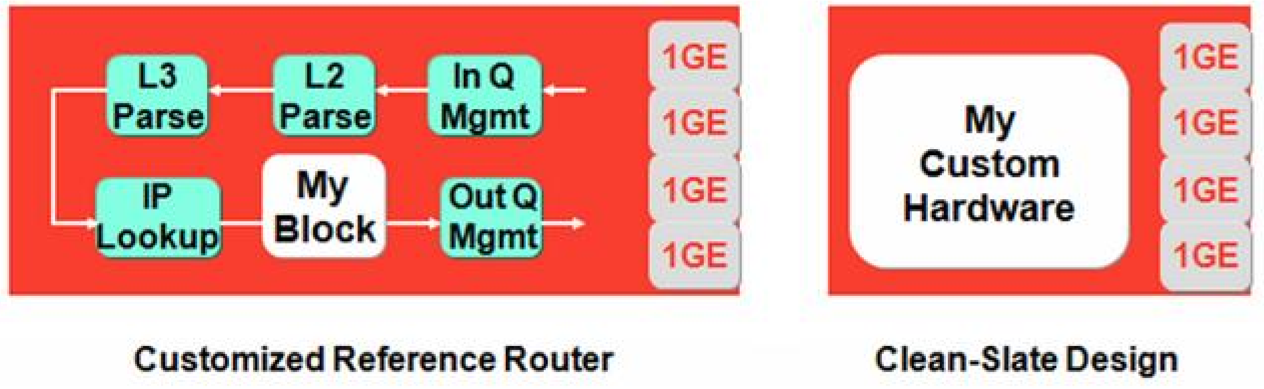


Figure 5.3 Two possible approaches to design project for NetFPGA card

When the reference structure is used, the user can use prepared blocks of code. They represent 4 input queues connected with each physical port (MAC RxQ) and 4 input queues form the driver from the PC side (CPU RxQ). Generally, there can be 8 input queues and all input data are store in them (according to the ingress port or driver). The next usable block is the input arbiter. It chooses which queue will be served in certain moment. When it chooses one nonempty queue, it reads packet data from this queue and processes the information. The simplest way to serve the packet is to make a decision about the output port. This control block can decide to send the packet to one of physical output ports or to the host (processor of PC). The decision is stored in a dedicated control structure and the data are sent to the next stage. After deciding about output direction the packet is placed in proper output queue and it is physically send to its next destination. Figure 5.4 shows the schema of the routing the packet inside the reference design. It is possible to add new blocks and extend the basic functionality of project.

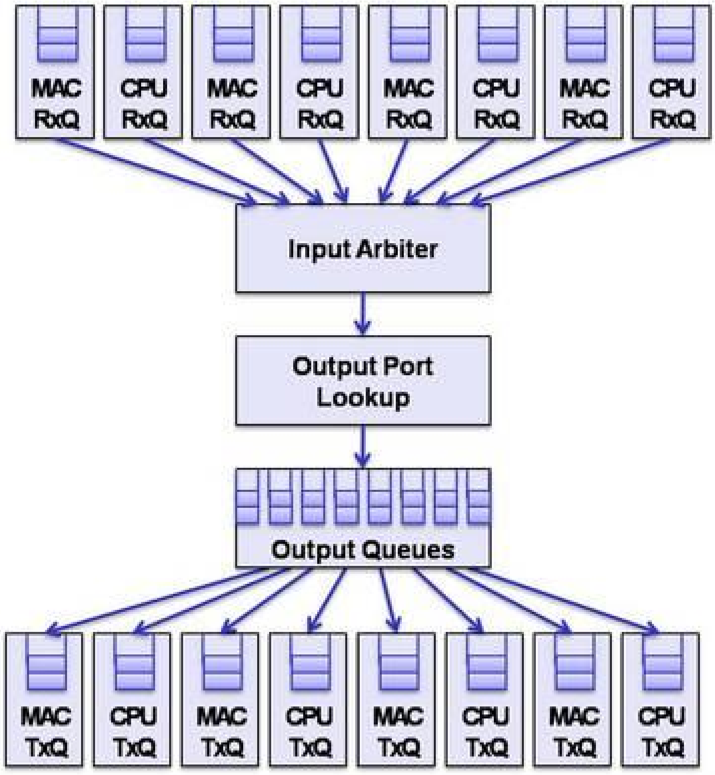


Figure 5.4 Diagram of the reference pipeline [netfpgareferenceNIC]

It can be said, that input packet (or Ethernet frame) is received on the physical port, it is placed in a sufficiently large input queue, next, in the correct time slot, the input arbiter chooses this packet for processing, it goes through the main pipeline, after the processing it is placed in proper output queue and physically sent out. This mechanism can be seen in Figure 5.5.

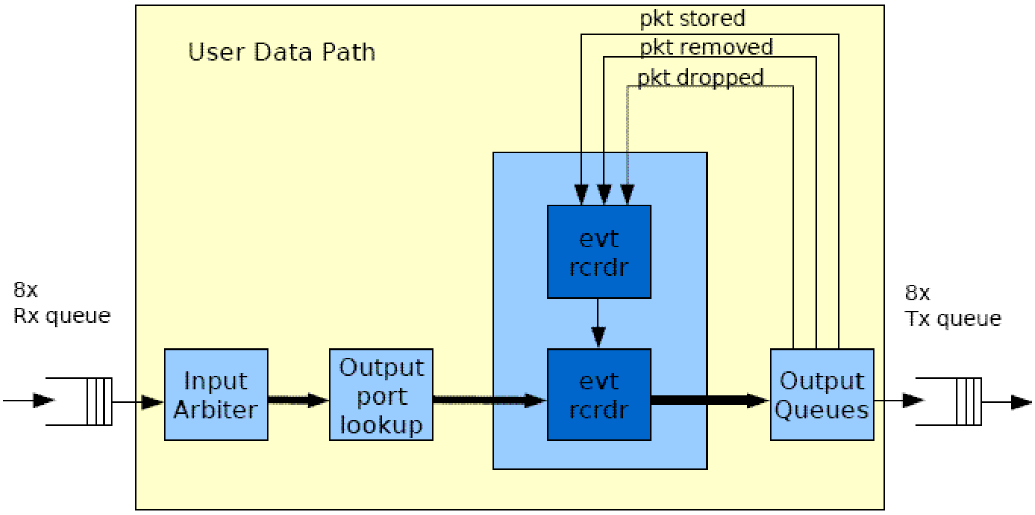
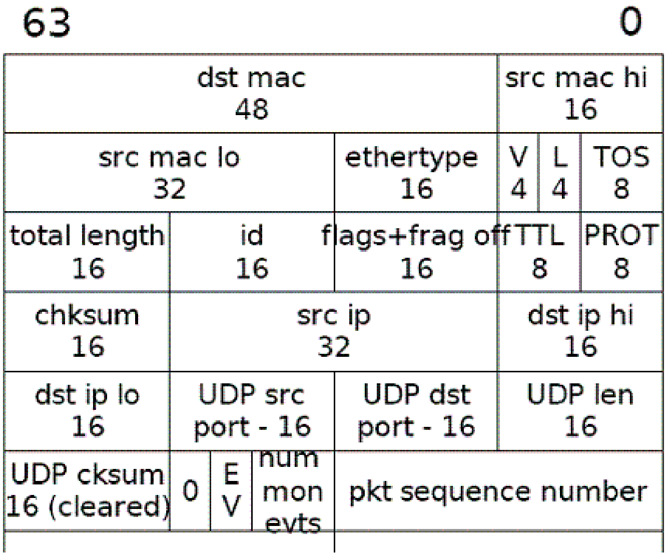


Figure 5.5 The reference user data path [netfpga-buffer]

5.2.2 Switching Management Rules

The Ethernet frames are stored in the input queues as 64 bit words. Each frame is divided into pieces with 8 bytes. So, the typical Ethernet frame with the IP packet in it is divided in parts, which are shown in Figure 5.6. This structure is very important for the designer, because in one clock cycle, only one such a slice is processed, and the dedicated finite state machines should be implemented to correctly find a state and process all the data. Additionally, each frame obtains the 64 bits control word at the beginning, which informs about incoming port (queue number), length, output ports etc.


63 0

dst mac 48				src mac hi 16		
src mac lo 32		ethertype 16		V 4	L 4	TOS 8
total length 16	id 16	flags+frag off 16		TTL 8		PROT 8
chksum 16	src ip 32			dst ip hi 16		
dst ip lo 16	UDP src port - 16	UDP dst port - 16		UDP len 16		
UDP cksum 16 (cleared)	0	E V	num mon evts	pkt sequence number		

Figure 5.6 Packet format inside the NetFPGA card

Basing on information from frame, control word and internal settings (such a ARP or routing table), the user/designer can prepare their own logic, which will control data switching in the hardware.

To obtain a typical Ethernet switch, the user has to download a reference project (Reference Switch) and run it on hardware. This is the learning switch, so, it can built its own ARP table and operate as a switch immediately without any configuration. It is possible (but not necessary) to monitor the state of its internal variables and data structures such as a counters of sent/received frames per port, learned MAC address on particular ports and so on.

To realize IP routing with the NetFPGA card, the user has to download the reference router project, run it on the NetFPGA chip and also run the software for the routing instance on the host. The proposed realization assumes that OSPF will be used. With the reference IPv4 router project, the user can run a router which can cooperate with other OSPF routers.

5.3 Control/Management Plane

5.3.1 Plane Purpose

Each realized project has its own functionality and specifications. So, the control and the management plane will be affected with its characteristics. The designers of the NetFPGA card assumed several possibilities for control the hardware part with the software part. The official way of performing control assumes use of the registry mechanisms [netfpgaregisters]. The registers allow writing some hardware variables from software and also reading them from software functions. With this mechanism the user/designer can realize even sophisticated mechanisms of control and monitoring (management) of software running in the NetFPGA card.

5.3.2 Exposed Protocols

The typical management application software run on PC. Based on provided functionality, examples of code and libraries, this software should be written in C or Java. With provided access mechanism, each project run on the NetFPGA card can have different management protocols implemented. There are no default management protocols, there is only a software interface for the card and applications which allow the downloading of the project to the card and monitor its activity in limited range. Wider scope should be provided by designers of particular projects.

5.3.3 Configuration Requirements

To be fully operational the NetFPGA card needs to be properly initialized in operating system. In order to do this dedicated drivers and software are provided. The kernel driver has to be compiled and installed in the OS. This needs to done only once. After each restart the card has to be initialized with small piece of code (to serve PCI). The main project also has to be downloaded into the main chip of the NetFPGA card (for example Ethernet switch). After this, proper software can be run and information can be written/read to/from hardware via registers and data are processed. Typical configuration of NetFPGA card consists on a setting up different MAC addresses for interfaces in different cards (initial MAC addresses in each card are the same, when more than one card is used they should be customized).

6 DOCSIS

Data Over Cable Service Interface Specification (DOCSIS) is a family of specifications developed by Cable Television Laboratories (CableLabs). DOCSIS has its origins in the cable industry of North America and enables high speed data transfer over an existing cable TV system (CATV). A coaxial-based broadband access network is typical in this scenario, which can be an all-coaxial or a hybrid-fiber-coaxial (HFC) network. The architecture of cable network is tree-and-branch with analog transmission. The frequency allocation bandwidth plans of CATV systems are different in United States and in Europe. Therefore, original standards have been modified for Europe, which are known as “EuroDOCSIS”.

There are three main elements in the DOCSIS system: the cable modem termination system (CMTS), the HFC infrastructure (previously mentioned), and the cable modem (CM). On the one hand, the CMTS is the head-end of the tree architecture and is the “intelligent” part of the system. The CMTS is located at the operator’s premises and connects the cable network to the core backbone of the operator. On the other hand, the cable modems are located at the customer’s premises, being the leaves of the architecture. Typically, the cable modems can be integrated in the set-top-box (video) or connected to a PC (data) with either Ethernet or USB.

The DOCSIS equipment available in the access networks laboratory is the following:

- CMTS: Cisco uBR7246VXR Universal Broadband Router
  http://www.cisco.com/en/US/prod/collateral/video/ps8806/ps5684/ps2217/product_data_sheet09186a0080092234.pdf
- CM: Cisco Modem/EPC3825 GW EU-DOCSIS 3.0 802.11n
  www.cisco.com/en/US/prod/collateral/video/ps8611/ps8675/ps8686/7018343.pdf

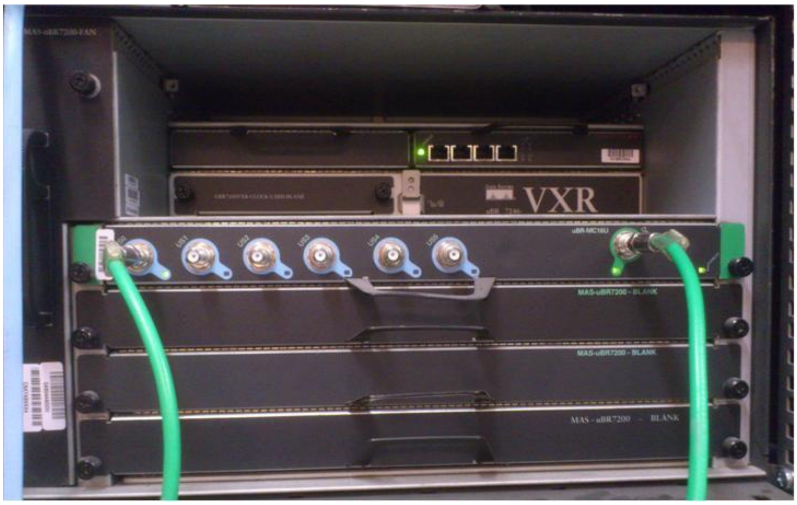


Figure 6.1 The Cisco uBR7246VXR Universal Broadband Router at the EHU-OEF

6.1 Physical box overview

As previously introduced, there are two different types of elements in the DOCSIS infrastructure deployed at the UPV/EHU laboratory: one Cisco uBR7246VXR Universal Broadband Router (CMTS) and twelve Cisco Modem/EPC3825 (CM).

Regarding the CMTS, the complete solution consists of several elements. The most relevant elements are described below:

Cisco uBR7246VXR Universal Broadband Router

This is the basic element which provides the chassis with the following dimensions (H x W x D): 26.67 x 43.18 x 53.98 cm, which means 6U of the rackspace. The Cisco uBR7246VXR supports up to four Cisco cable line cards, each featuring one or two downstream and six or eight upstream cable interfaces, for a total of up to eight downstream and 32 upstream interfaces in a single chassis. The main features are modular scalability and highest reliability, supporting up to 10,000 subscribers.

Figure 6.2 The Cisco uBR7246VXR chassis

Cisco uBR7200-NPE-GI Network Processing Engine

This NPE board takes advantages of microprocessors specifically designed for data networking applications. The major features of this element are: 3x 1GE (10/100/1000 Mbps) Copper/Fiber Ports, 1x console port (RS-232 9600, 8, N, 1, XON), 1x management port (RJ-45), Compact Flash, up to 1GB of DRAM and 256 MB of Flash memory.

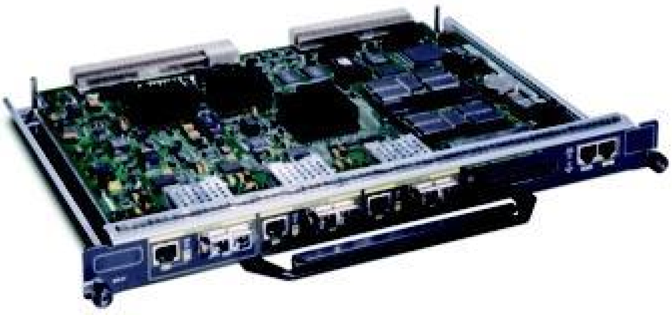


Figure 6.3 The Cisco uBR7200-NPE-GI

Cisco uBR-MC16U

The MC16U line card combines the highest level of integration with enhanced RF robustness with one downstream and six upstream cable interfaces. The RF modulation is 64-QAM or 256-QAM for downstream and QPSK 8-, 16-, 32-, and 64-QAM for upstream. The frequency range is 8MHz Annex A, 85-860 MHz (Euro-DOCSIS) for downstream and 8MHz Annex A, 5-65 MHz (EuroDOCSIS) for upstream.

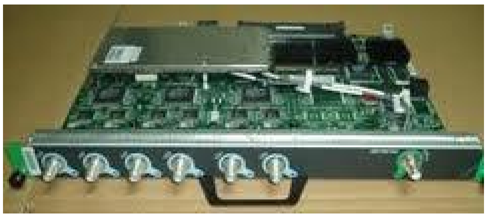


Figure 6.4 The Cisco uBR-MC16U

Regarding the CM, the Cisco Modem/EPC3825 8x4 EuroDOCSIS 3.0 is a high-performance home gateway that combines a cable modem, router and wireless access point in a single device. It provides a cost-effective solution for both small offices and residential users. Thanks to the capability of bounding channels, the device can deliver up to 440 Mbps downstream (eight bonded downstream channels) and 120 Mbps upstream (four bonded upstream channels).

Figure 6.5 The Cisco Modem/EPC3825

The cable modem has the following connections: four 10/100/1000BASE-T Ethernet ports, 2x2 802.11n wireless access point with four SSIDs, and 1x USB 2.0 Type 1 host port. The dimensions (W x D x H) are 14.5 cm x 18.6 cm x 5 cm.

6.2 Data plane

6.2.1 Transmission technology

There are differences between versions of DOCSIS specification (i.e. v1.0, v2.0 and v3.0) and between United States and European specifications. This section mainly covers EuroDOCSIS v3.0, although the equipment is backwards compatible and different versions can be used in the platform.

Regarding the most physical part of DOCSIS, there are several options for the RF modulation and frequency plan. Following the EuroDOCSIS specification, in the downstream direction, the RF modulation is 64-QAM or 256-QAM with a frequency range of 85-860 MHz. The cable system is assumed to have a pass band with a lower edge 47-87.5 MHz and upper edge typically in the range of 300 and 862 MHz. It is common to find PAL/SECAM analog television signals in 7/8 MHz channels, FM radio signals, and other narrowband and wideband digital signals. In the upstream direction, the RF modulation is QPSK, 8-QAM, 16-QAM, 32-QAM and 64-QAM with a frequency range of 5-65 MHz. EuroDOCSIS defines 8 MHz channels for data communication in both directions.

On top of the RF modulations, DOCSIS defines the Media Access Control specification. Some important details of the version 3.0 of the DOCSIS MAC protocol are described next. The bandwidth allocation is controlled by the CMTS, which is the intelligent part of the system. The upstream is shared by all the cable modems, being the most challenging part and a stream of mini-slots are defined in this upstream direction. The system allows the dynamic mix of contention-based and reservation-based upstream transmit opportunities. The efficiency of bandwidth allocation is obtained through support of variable-length packets. Regarding QoS support, DOCSIS provides bandwidth and latency guarantees, packet classification and dynamic service establishment. Extensions are provided for security at the data link layer. A wide range of data rates are supported, including the logical combination of multiple channels (bonding) for increasing the throughput.

The MAC-sublayer domain is a group of channels, both in upstream and downstream directions, operated by a single MAC Allocation and Management protocol. This includes one CMTS and several CMs. The CMTS provides all the upstream and downstream channels, and the CM may access one or more channels in the upstream and one or more channels in the downstream. Each upstream channel may use a different DOCSIS format (i.e. 1.x, 2.0 or 3.0). Any packet with a source MAC address that is not a unicast MAC address is discarded by the CMTS.

A MAC frame is the basic unit of data exchanged at the Data Link Layer between two entities at the CMTS and the cable modem. The same structure is used in the upstream and downstream directions. The MAC frame consists of a MAC Header and a variable-length data PDU. The variable-length PDU is used to encapsulate the 48-bit source and destination MAC addresses, the data payload and the CRC. The DOCSIS MAC Header uniquely identifies the contents of the MAC frame and can be used to encapsulate multiple MAC frames. Preceding the MAC frame is either the PMD sublayer overhead in the upstream direction or the MPEG transmission convergence header in the downstream direction.

The CM and CMTS MAC sublayer support a variable-length Ethernet-type Packet Data PDU MAC Frame. This Packet PDU can be used to send unicast, multicast and broadcast packets, which are passed across the network including its original CRC.

The Service Flow concept is a key element to the operation of the MAC protocol. It provides a mechanism to manage the QoS for both upstream and downstream traffic. More specifically, the Service Flows are integral to bandwidth allocation. The Service Flow ID defines a unidirectional mapping between a CM and the CMTS. Service IDs (SIDs) are associated to active upstream Service Flow IDs, and upstream bandwidth is allocated to SIDs by the CMTS. This means that the bandwidth allocated to CMs is related to the SIDs associated to each CM. Therefore, the QoS implemented in the upstream direction is provided by the Service IDs. Depending on the Service Flows required by the CM, the CMTS assigns one or more Service Flow IDs to each CM. This mapping can be negotiated between the CMTS and the CM during the CM registration or via dynamic service establishment.

In the simplest scenario, two Service Flows, one for upstream traffic and one for downstream traffic, can be used to provide best-effort IP service. However, a more complex scenario is also possible with support for multiple service classes, such as a maximum packet size limitation or a small fixed size. With independence of the specific QoS configuration, it is necessary to send certain packets upwards for MAC management, SNMP management or key management. For this, all CMs must support at least one Service Flow for upstream and another for downstream, which are referred to as the upstream and downstream Primary Service Flows. These Primary Service Flows should be provisioned always at registration time between the CM and the CMTS, and may be used for traffic. All unicast Service Flows use the security association defined for the Primary Service Flow.

The CMTS ensures that all Service Flow IDs (32 bits length) are unique within a single MAC-sublayer domain. An active service flow maps to one or more Service IDs. Service IDs (14 bits length) are unique per logical upstream channel. The CMTS may assign the same SID to two or more unicast flows on the same MAC-sublayer domain if they are attached to different logical upstreams.

6.2.2 Switching management rules

The cable modem is the device that connects the customer’s network (i.e. residential network or small office) to the operator’s HFC network, bridging packets between them. Basically, the CM can be configured at booting time (depending on the parameters of the configuration file) to run as a bridge/switch or as a router with NAT support. The CM can be connected to the customer’s CPE (i.e. PC) with an Ethernet interface or a USB connection. In other cases, the CM can be embedded with the CPE in a single device (i.e. set-top box). CPE devices typically use IPv4, IPv6 or both types of addresses.

The CMTS connects the HFC network with the operator’s core network. The main function of the CMTS is to forward packets between HFC and core network, and it is also possible to forward packets between upstream and downstream channels of the HFC network. The forwarding performed by the CMTS can be done at link layer or at network layer. The former means bridging, whereas the later implies routing. Therefore, the CMTS can act as a bridge or as a router.

The internal forwarding model of a CMTS defines two different sub-components: (1) the CMTS Forwarders, which forwards the packets with layer 2 bridging or layer 3 routing; and (2) the MAC Domains, which forward data to and from the cable modems (i.e. downstream and upstream channels). Therefore, the forwarding of packets between the MAC Domains and the network side interface (towards the core network) is performed by the CMTS Forwarder.

In DOCSIS 3.0, all the upstream data packets are delivered to the CMTS Forwarder, since the MAC Domain does not forward the data packets from its upstream to its own downstream channels. DOCSIS 3.0 also adds the requirement to manage CMs with IPv6, as well as to provide IPv6 connectivity to the CPE equipment. The previous specifications define IPv4 as the single alternative in both cases. However, DOCSIS does not specify if the CMTS should implement layer 2 or layer 3 forwarding of IPv4/IPv6 protocols; or if some protocols should not be bridged or routed. In addition, the DOCSIS Layer 2 Virtual Private Networking specification defines transparent layer 2 forwarding of CPE through the CMTS, which requires a new L2VPN CMTS Forwarder (different from the previous CMTS Forwarder).

A DOCSIS MAC Domain is a logical sub-component of a CMTS responsible for implementing all DOCSIS functions on the upstream and downstream channels. At least one downstream and one upstream channel is needed on a MAC Domain. All the MAC Management Messages (MMMs) are sent by the MAC Domain to the CMs. Each CM is registered to just a single MAC Domain. The layer 2 data transmission between the CMs and the CMTS Forwarders is provided by the MAC Domain.

The downstream packets are classified by the MAC Domain into service flows, which are distinguished based on layer 2, 3 and 4 information. Then, the scheduler of the MAC Domain assigns the packets to each downstream service flow and the corresponding downstream channel.

Based on the MAC Domain, the CMTS Forwarder identifies the CM which sends each upstream packet. Then, the CMTS Forwarder forwards the packets between the MAC Domains and the core network. DOCSIS specifies that the CMTS Forwarder is also responsible to forward packets between MAC Domains, from upstream to downstream channels. Therefore, the IPv4 ARP and IPv6 ND protocols need to be implemented within the CMTS Forwarder.

6.3 Control/management plane

6.3.1 Plane purpose

On a DOCSIS network, there are several applications that are needed to configure the devices. Both IPv4 and IPv6 applications are possibilities in DOCSIS v3.0. These applications include the following provisioning systems: (1) the DHCP server provides the initial configuration to the CM at booting time, such as the IP address; (2) the Configuration File server provides the configuration files (binary format with CM's parameters) to the CM when they boot; (3) the Software Download server is used to upgrade the CM's software; (4) the Time Protocol server provides the current time to the CMs; and (5) the Certification Revocation server provides the current status of certificates used to establish the secure association.

The Network Management System (NMS) is used to manage the DOCSIS network. The NMS consists of different elements: (1) the SNMP Manager allows the operator to configure and monitor the SNMP Agents located both at the CM and at the CMTS; (2) the syslog server is used to collect all the messages related to the operation of the devices; and (3) the IPDR Collector server is used to efficiently collect bulk statistics from the devices.

Proactive network management can be performed based on advanced plant diagnostics and troubleshooting. Even when upstream noise is present in the cable plant, reliable service to end users can be maintained based on advanced spectrum management capabilities.

6.3.2 Exposed protocols

The Cisco uBR7246VXR runs a Cisco IOS Software like any other Cisco router. Specifically, it delivers high-performance routing capability at the edge. Several standardized routing protocols are supported by the platform, such as Open Shortest Path First (OSPF), Internal Border Gateway Protocol (IBGP), Border Gateway Protocol (BGPv4), IP Multicast, and many other routing and switching protocols.

Different QoS features are supported and available for configuration, such as Resource Reservation Protocol (RSVP), Weighted Random Early Detection (WRED), or Weighted Fair Queuing (WFQ). NetFlow switching is also supported.

Regarding the service provisioning, the Cisco uBR7246VXR provides static DOCSIS 1.1 QoS service flow creation and dynamic QoS (D-QoS) service flow creation based on PCMM. This enables the real traffic shaping and management.

6.3.3 Configuration requirements

As with any other Cisco router there are different options to configure the equipment. As long as there is a Cisco IOS Software running on top of it, the procedure is well-known. It can be configured through the console port, via RS-232 serial port (9600, 8, N, 1, XON) from a PC. It can also be configured through the auxiliary port, specifically dedicated for management purpose. Another option is to remotely access the router by using one of the IP addresses assigned to one of the 1GE ports from the Network Processing Engine (uBR7200-NPE-GI). In all this cases, the Cisco IOS CLI is available to configure the equipment.

7 ATCA

7.1 Overview

The Advanced Telecommunications Computing Architecture is a form factor of chassis, blades, and connectors that is standardized by the PCI Industrial Computers Manufacturing Group (PICMG).

Core idea is the use of Commercial Off-The-Shelf (COTS) components from a wide range of suppliers by network equipment providers.

Currently all leading telecom providers use ATCA, mostly in the area of 3G and LTE equipment, according to recent market reviews. The next generation, a 6-slot ATCA box of EMERSON, is introduced briefly as well.

7.1.1 Physical box overview

ATCA chassis are available in different sizes from a number of manufacturers. An ATCA chassis is 600mm deep and carries 2,5,6,14 or 16 blades (see . Configurations up to 6 blades mount the blades horizontally, the larger 14 and 16-blade chassis are made for 19” and 23” racks, respectively, and have the blades vertically mounted. Each blade is 322mm (8U) wide and connects to the backplane of the chassis with pins ordered in 3 zones:

- Zone 1: power supply (redundant -48V) and shelf management
- Zone 2: base and fabric connections. The data transmission format is typically 10/100/1000 Mbit/s Ethernet, 10G specifications (10G-KR, KX) added in August 2012. 40G fabric interconnects available from several manufacturers, 100G announced for 2014.
- Zone 3: freely configurable I/O to rear transmission modules, mid-planes or others. Data transmission format left to the manufacturer.

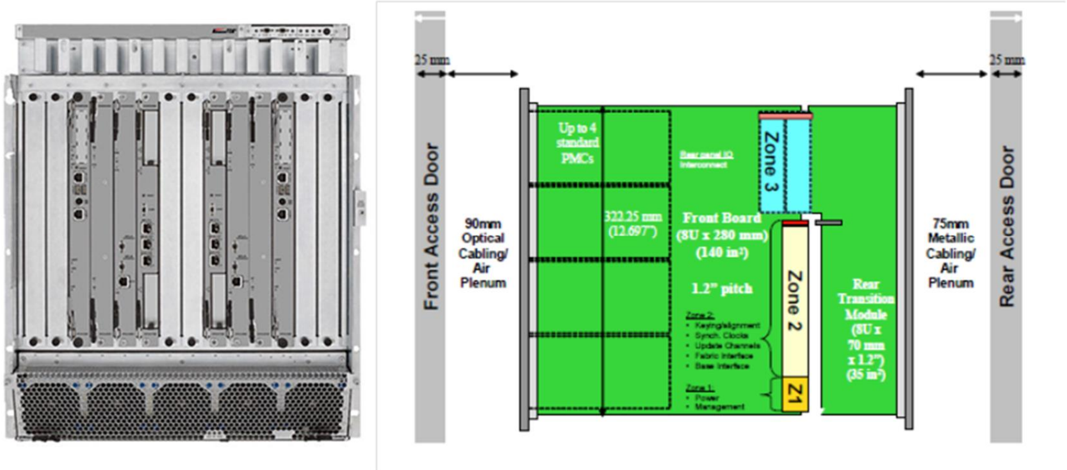


Figure 7.1 Left: typical (here: 14-slot) ATCA shelf; Right: mechanical specification of an ATCA blade.

Figure 7.2 shows the available configuration at EICT. It consists of two carrier boards that each can host so-called Advanced Mezzanine Cards (AMC), a pluggable extension. EICT has two AMC cards with Intel Core2 Duo (AM4020) and one Network Processor card (AM4220) with an OCTEON Plus 5650 on it.

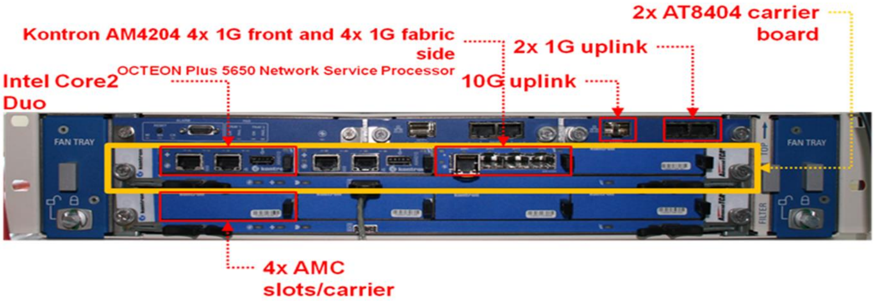


Figure 7.2 KONTRON OM5080 shelf equipped with two AT 8404 carrier boards each capable of carrying 4 AMC cards.

7.2 Data plane

7.2.1 Transmission technology

At the heart of each of the carrier boards there is a Broadcom BCM 56502 Ethernet switching chip with 24 GbE and 2 10GbE ports. One of the 10G ports points to the front plate of the board, while the other one connects back-to-back to the opposite carrier. Each AMC carrier slot is connected with 5* 1GbE, and it depends on the configuration of the AMC card how many of these ports are presented further to the front plate. All other processing capabilities aside, the configuration of the available box can be seen as two stacked Ethernet switches.

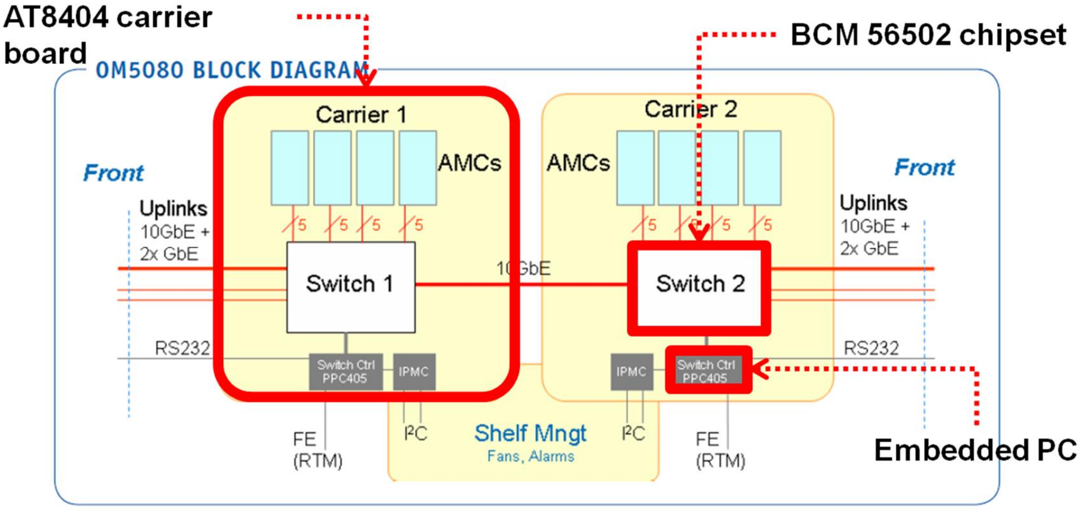


Figure 7.3 Datapath layout of Kontron ATCA OM 5080 configuration, AMC cards can be equipped with Intel PCs or OCTEON Cavium Network Processors.

7.2.2 Switching management rules

The BCM chips on the carrier boards are controlled by a PowerPC 405, configurable from the outside via RS232. In previous projects, the group of EICT has managed to load an OpenFlow datapath implementation into this PPC405, turning the switch into and OpenFlow 1.1 switch [HPSR12JungelRostami]. An essential element that is required to further enhance the implementation is the availability of Board Support Packages for the hardware that any datapath implementation is being developed upon. In the past, these BSPs have not been available from Kontron, forcing us into after-market development and slowing down the entire process of datapath development.

7.3 Control/management plane

7.3.1 Plane purpose

The fact that the two switches on the backplane could be turned into OpenFlow switches makes it easy to think of a complex OpenFlow 1.2 speaking router, comprising different elements that fulfil roles beyond simple Ethernet forwarding. An example is BRAS functionality, representing the first router in the hierarchy of a telecommunication operator. Residential clients use the PPPoE/PPP protocol family to connect to the BRAS, retrieve an IP address used for the home router, and monitor the status of the link between the BRAS and the client. These functions go beyond current OpenFlow specifications, and have to be deployed on a combination of merchant silicon (for pure forwarding), network processors (for encapsulation/decapsulation of PPPoE and PPP frames), and software (for an orchestrating controller).

7.3.2 Exposed protocols

Currently, a datapath implementation of OpenFlow 1.2 is available on x86. The port of this implementation to the OCTEON network processor and the Broadcom chip will be finalized Q1 2013, requiring a rudimentary Hardware Abstraction Layer. This will mean that all individual elements (carrier boards, AMC cards) will expose themselves as individual OpenFlow switches. The design and implementation of the orchestrating controller will take to the end of 2013. During this time, the jump to OpenFlow 1.3.1 will be done as well, so eventually this as well as a simple OfConfig 1.0 protocol will be available to the external researchers.

Different degrees of Openness will apply to the code used on the box. While the OpenFlow generic datapath implementation will be available under a commercial-friendly Open Source license like MPL, lower parts of the code that use directly the Software Development Kits of Broadcom, for instance, will have to respect the license conditions that the vendors impose.

7.3.3 Configuration requirements

In a first step, individual datapath implementations will be configurable via a standard OpenFlow controller like Floodlight or NOX. Further extensions to OpenFlow for better support of virtualization will not interfere with basic interoperation of standard controllers.

8 Layer 0 switch

8.1 Overview

ADVA FSP 3000R7 in University of Bristol is a high-performance WDM networking system for bidirectional transmission of optical signals. These signals are transmitted in a defined number of channels over a single fibre pair or one duplex fibre. The system uses a modular structure which enables a flexible upgrade of capacity and functionality according to network requirements.

The FSP 3000R7 is comprised of various types of shelves and modules. Each type performs different functions within the system. Each shelf can host different modules and components with different functionalities.

8.1.1 Physical box overview

SH1HU-R

The SH1HU-R is a rack-mountable with dimension of 453 x 363 x 44.45 mm, 1 HU rear power access shelf with redundant AC power supply. The shelf is the housing which holds the entire FSP 3000R7 system. The shelf features four horizontal slots on the front side for accommodation of the standard FSP 3000R7 modules. The left-hand slot contains the fan control unit (FCU).

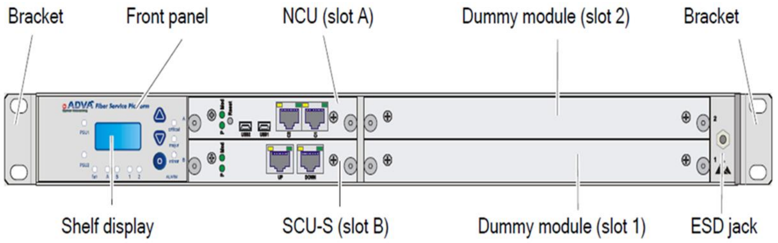


Figure 8.1 SH1HU-R shelf

The SH1HU-R has been designed for low channel count installations, single services and feeder applications. It is mainly used as an access carrier in the area of backhaul transport. This shelf may also be used for single channel add/drop in a CWDM ring (one channel module, one east/west optical filter module).The basic configuration is: 1 x network element control unit (NCU) and 1 x shelf control unit (SCU). Optical modules and management modules may be pre-installed according to the system configuration ordered.

SH7HU

The SH7HU (7HU Shelf) is a rack-mountable with dimension of 452 x 270 x 311 mm , 7 HU-high housing which includes the entire FSP 3000R7 system and the accessories (dummy modules, adaptor brackets, front cover).

22 vertical slots each 4 HP wide are arranged in the middle of the shelf, into which the standard FSP 3000R7 modules are plugged in. The slots occupy 5 HU space. The remaining 2 HU of the shelf are occupied by the fan unit above the slots and the air filter unit below the slots. The front view of an SH7HU is illustrated in Fig. 8-2

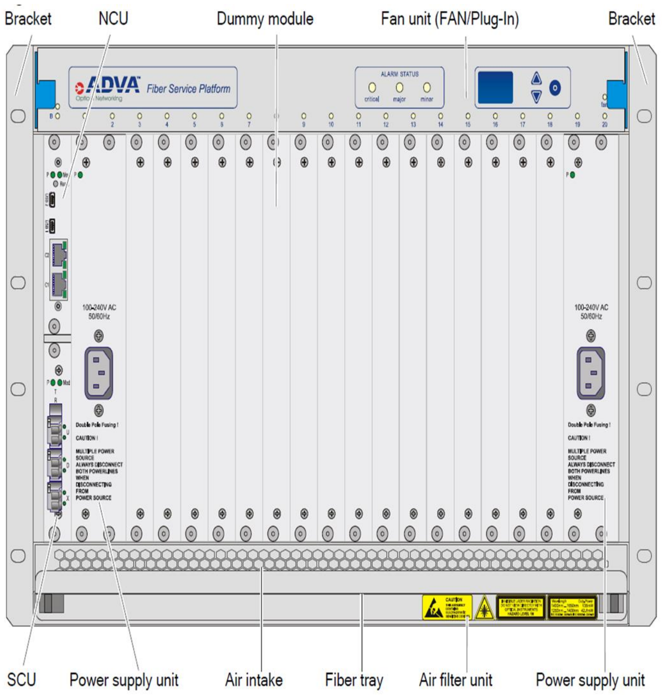


Figure 8.2 SH7HU shelf

8.2 Data plane

8.2.1 Transmission technology

As previously mentioned the ADVA optical switching system has a modular architecture and each module performs a different function. The transmission between modules is optical and passive. This means the signalling and controlling of the modules are completely separated from data plane. In the following, the modules in the data plane will be described. The modules could be enabled and disabled based on the applications and use cases.

Modules hosted in the ADVA FSP 3000R7 are:

CCM-C40/8

The CCM-C40/8 is an eight port MUX/DEMUX channel filter for up to 8 channels. It allows any channel from the 40 supported wavelengths on a network port to be connected to any client port. The CCM-C40/8 uses 2 primary components, a wavelength selective switch (WSS) and an 8 port optical combiner. The WSS is used in the de-multiplex path and the 8 port optical combiner is used in the multiplex path.

In the de-multiplex path, the CCM-C40/8 routes any channel from the network port to each client port. Any of the 40 standard DWDM channels can be routed. In the multiplex path, the module combines the signals from all of the client ports and forwards them to the network port. The 5PSM can be used to create the optical multiplex of up to 40 channels by combining the network ports of five CCM-C40/8 modules. The CCM-C40/8 requires amplification in the add/drop paths in many applications to meet the target power levels for channel transmission and/or reception.

The module’s features are:

- Supports channel selection from the standard 40 DWDM wavelengths up to 40 wavelengths for add and drop channels
- Selectively routes channels from the network port to the client port outputs
- Combines all channels from client port inputs to the network port output
- Provides port level Performance Monitoring for network input and output
- Provides port level Performance Monitoring for client port outputs

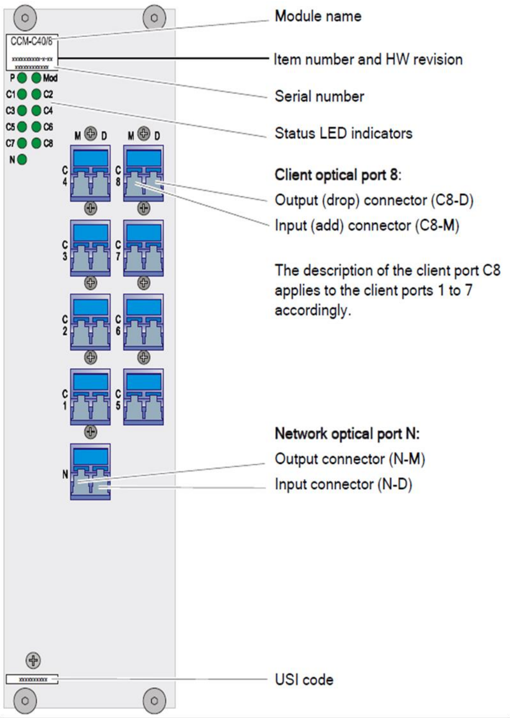


Figure 8.3 CCM-C40/80 MUX/DEMUX

The signal path through a node is typically bi-directional. The network ingress allows a channel or channels from the network input optical multiplex to be routed to one of the client ports. The output power to the client output ports can be monitored and controlled.

The CCM-C40/8 could give two different functional applications:

- Reconfigurable Optical Add Drop Multiplexer (ROADM) Fixed Add/Drop: This provides a "colourless" add/drop of client signals on each network interface. Colourless is a term found in the optical industry that means any wavelength in the ITU grid can be added or dropped at the Network Element (NE). This application is colourless and illustrates how any channel in the ITU grid can be added or dropped from each network interface. This application is limited to 8 colourless add/drop channels per network interface.

- Colourless Directionless Add/Drop: Directionless is also an optical industry term that means any wavelength in the ITU grid can be routed to any NE. This application is also limited to 8 colourless directionless add/drop channels.

EDFA-SGCB

The EDFA-SGCB is a fixed gain-controlled, single-stage Erbium-doped fibre booster amplifier to amplify up to 80 channels in the C band group suitable for booster, inline or preamplifier applications. The power level of the outgoing amplified signal may be monitored.

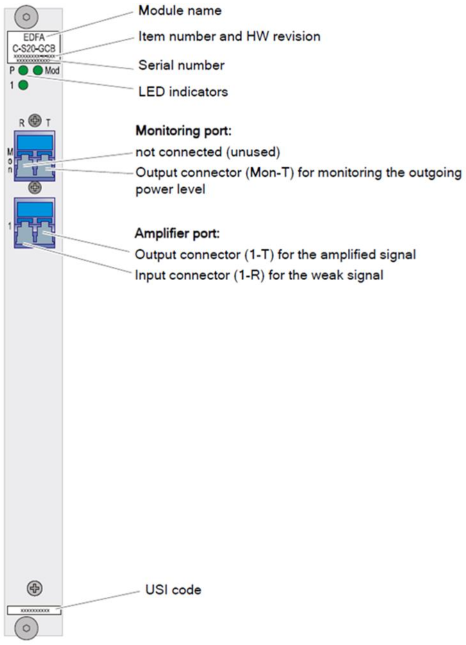

EDFA
C-S20-GCB
P
Mod
1
R
T
Mon
1
Module name
Item number and HW revision
Serial number
LED indicators
Monitoring port:
not connected (unused)
Output connector (Mon-T) for monitoring the outgoing power level
Amplifier port:
Output connector (1-T) for the amplified signal
Input connector (1-R) for the weak signal
USI code

Figure 8.4 EDFA-SGCB amplifier

EDFA-DGCV

The EDFA-DGCV is a gain-controlled, double-stage Erbium-doped fibre amplifier with a variable gain supporting amplification of up to 80 channels in the C band. The first gain block works as preamplifier, the second as booster amplifier. The mid-stage access allows inserting a dispersion compensation module without affecting the line budget. A variable optical attenuator (VOA) at the mid-stage access controls and suitably adjusts attenuation of up to 16 dB depending on whether a dispersion compensating module or a jumper connects the two gain blocks. The EDFA-DGCV can be used as pre-amplifier, booster or inline amplifier.

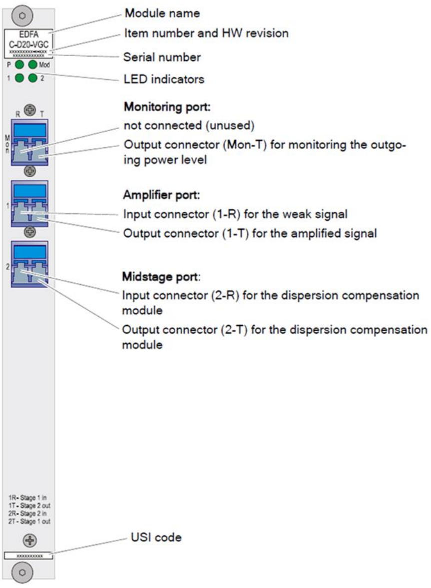

EDFA
C-D20-VGC
P
Mod
1
2
R
T
Mon
1
2
1R- Stage 1 in
1T - Stage 2 out
2R- Stage 2 in
2T - Stage 1 out
Module name
Item number and HW revision
Serial number
LED indicators
Monitoring port:
not connected (unused)
Output connector (Mon-T) for monitoring the outgoing power level
Amplifier port:
Input connector (1-R) for the weak signal
Output connector (1-T) for the amplified signal
Midstage port:
Input connector (2-R) for the dispersion compensation module
Output connector (2-T) for the dispersion compensation module
USI code

Figure 8.5 EDFA-DGCV amplifier

8ROADM-C40

The 8ROADM-C40 is an FSP 3000R7 channel module that functions as a network wavelength switching point. The 8ROADM-C40 module can be deployed alongside existing FSP 3000R7 modules in an FSP 3000R7 NE.

An FSP 3000R7 NE containing 8ROADM-C40 modules supports multiple interconnected rings and mesh network configurations. The 8ROADM-C40 module enables provisioning and adjustment of Add/Drop and pass-through DWDM C-band channels and supports these optical connection configurations:

Add/Drop
Pass-through
Drop/Continue
Multicast
Add
Drop
Drop/Multicast

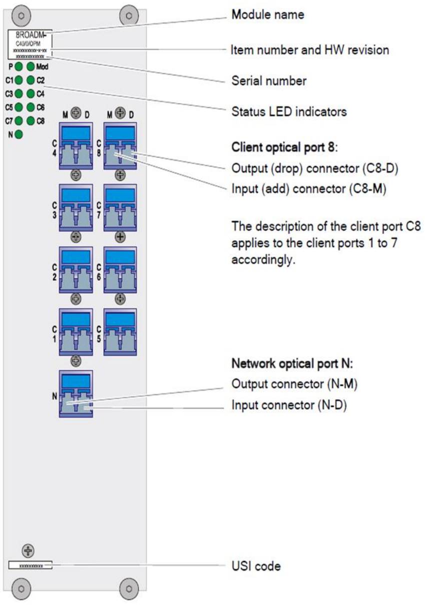


Figure 8.6 8ROADM-C40

The 8ROADM-C40 module can handle 40 C-band channels with 100 GHz spacing. It contains one duplex network interface and 8 duplex client interfaces. The D path (N-D to Cx-D) signal is split into 8 identical copies, which are available on the module front panel at the Cx-D outputs.

The 8ROADM-C40 has the following features:

- Dynamic configuration of add or pass through paths
- Supports forty 100 GHz spaced channels for even frequency channels 192.00 THz through 195.90 THz
- Support of pass through, add/drop, or drop/continue configurations
- Fully transparent
- Transport protocol and bit-rate independent
- Full management support
- Status LED indicators for power, module and optical port operability
- LC type receptacle connectors, non-angled for fibre termination
- All optical connectors front accessible
- Fiber type: SM G.652

ROADM-2DC

The ROADM-2DC is a reconfigurable optical add/drop multiplexer (ROADM) designed as a rack-mountable 3HU high standalone 19-inch shelf. It contains all components necessary for its function. This includes power supplies, the ROADM system, control electronics, management interfaces, and adaptor brackets.

The ROADM system is based on a planar lightwave circuit (PLC) architecture. It uses a switch matrix, a VOA array, internal AWG Multiplexer/Demultiplexers, and an array of photo-detectors for monitoring channel powers.

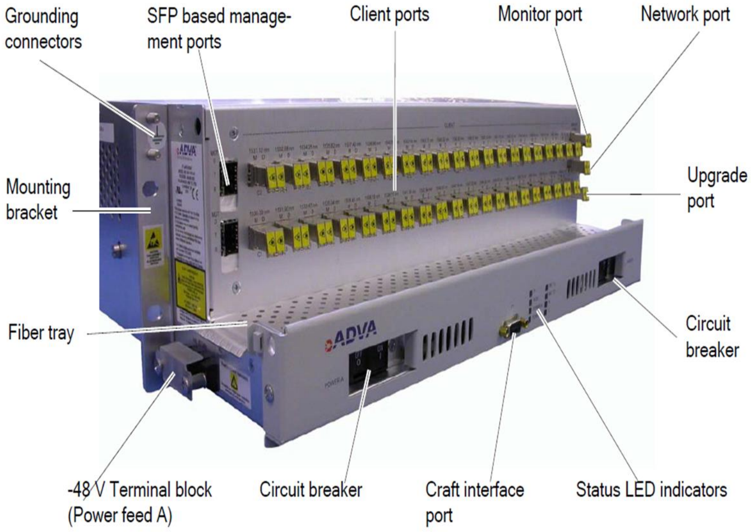


Figure 8.7 ROADM-2DC

The ROADM-2DC provides remote re-configurability of up to forty C-band channels in the range from #D02 to #D32 inc. #DC1 to #DC9, eliminating the need for the user to physically be at the site. Using the NE management tools any channel, or combination of channels can be reconfigured on demand, to either pass-through, drop and continue or add/drop at the FSP 3000R7 node without interrupting existing services.

The key features of the ROADM - 2DC are:

- PLC based switch matrix for dynamic configuration of add or pass through path
- Support of 40 channels (wavelengths) in the C band according to ITU –TG.694.1 (100 GHz channel spacing)
- Support of pass through, add/drop, or drop/continue configurations
- Improved transmission performance due to adjustment of wave-length levels Monitoring of the optical power at the network port, upgrade port, and client ports
- Transport protocol and bit-rate independent
- Full management support
- Managed as main shelf using optical SCU to SCU daisy chain connection

8.2.2 Switching management rules

On ADAVA FS300R7, ROADM and CCM-C40/8 are the components that can be configured dynamically. They allow to access to any wavelength at any node at any time. That means they will give the ability to drop or send any wavelengths (colourless) in any direction (directionless or bidirectional) on any available port on the NE in a non-blocking fashion which means it is contention-less. The components allow provisioning and adjustment of Add/Drop and Pass Through DWDM C-band for up to 40 channels with 100 GHz channel spacing or 80 channels with 50 GHz channel spacing (depending on module’s specifications).

On 8ROADM module, the received signal on network port could be split into 8 identical copies and then they are available on client port outputs. Also, it is possible to receive multiple wavelengths on the network input port and select a wavelength and send it out on the network output port (cross-connecting). On CCM-C40/8 module, the multiplex system takes the input from all 8 client ports and optically combines them into an optical multiplex of up to 40 channels and sends them to the network port. In the de-multiplex path it gets the input from the network port and sends a selected channel to one of the 8 client ports.

All these rules which here could be setting a channel to be added or dropped or passed through the NE can be created by using the CRAFT console, TL1 commands or WEB GUI and then they could be applied on the modules.

8.3 Control/management plane

8.3.1 Adaptation of ADVA's optical ROADM to OpenFlow protocol

The base adaptation architecture of the ADVA's optical ROADM Network Element (NE) assumes cooperation with the GMPLS Control Plane. The architecture assumes that each NE is represented as an individual OpenFlow enabled optical switch. Our approach utilizes OpenFlow Circuit Switching Extensions in order to describe circuit switching features of the optical switch. Therefore the OpenFlow controller is able to see the network topology as well as the optical layer specific information, e.g. wavelengths available on OpenFlow ports. The switch and network discovery process is therefore accomplished using the extended OpenFlow protocol. On the other hand resource allocation is done by the GMPLS Control Plane. GMPLS Control Plane can allocate paths in two modes:

- Loose paths
- Explicit paths

In the former mode the users only provide ingress and egress switches and ports and the Control Plane computes the rest, while in the latter one the users can provide full path description (i.e. all switches and ports along the path) and the Control Plane will verify if it is correct and then try to establish it. Therefore that approach is flexible and can utilize many features available in the ADVA's Control Plane.

The adaptation has the following features:

- Circuit extensions based on V0.3 addendum to OF Version1.0 for OpenFlow (OF) controller. The extensions extend Stanford proposed OpenFlow circuit V0.3 specification to add switching constraints, power equalization, impairments functionalities
- The controller also supports an integrated GMPLS approach to utilize best of both control planes (OF & GMPLS)
- Developed a modular OF Agent architecture which can be easily employed on any native management interface (SNMP, TL1)
- Hybrid (packet-optical) approach to retain packet functionalities
- FlowVisor extended to support OF circuit extensions
- SDN Applications (algorithm suite) for path computation across optical domain.

8.3.1.1 *OF Extensions:*

The implementation of the optical OpenFlow agent is based on the circuit switching extensions ver. 0.3. These extensions were merged with OpenFlow ver. 1.0.0 protocol specification. OpenFlow protocol ver. 1.0.0 is used in the OFELIA Control Framework

Changes introduced to OF protocol concerns specific application of the optical networks, namely an application known as the "alien wavelength". It means that the ADVA ROADM NEs offer purely optical, DWDM interfaces to which packet switches can be directly connected. Therefore no additional adaptation takes place in ADVA's NEs and NEs are not equipped with transponders. Additions to the OpenFlow protocol were also driven by different concepts regarding co-operation with the GMPLS Control Plane implemented on ADVA NEs.

Optical OpenFlow Extensions (OOE) extensions were added as vendor extension messages. Vendor extension feature of the OpenFlow protocol allows for extending the protocol without breaking the compatibility with the base protocol specification ([2]). New vendor (experimenter) specific messages can by piggybacked using standard OFPT_VENDOR type messages

- **Management extensions** were introduced to support Control Plane (CP)-assisted optical OpenFlow which assumes cooperation with ADVA's GMPLS Control Plane. In CP-assisted OpenFlow, an OpenFlow controller uses the GMPLS Control Library module which sets up or tears down light paths using ADVA's management interface, namely the SNMP protocol. OpenFlow controller is required to know SNMP agent transport address and the SNMP community string in order to talk to an NE over SNMP. This information (address and community) can be provided to controller by means of a configuration file or can be dynamically retrieved from NE through OpenFlow protocol using management extensions.

- **Switching constraints** describe how physical ports are connected with each other. This relationship between ports results from internal NE configuration. NE is composed from a number of physical cards connected with each other by fibre jumpers. Switching constraints map tells if the optical signal can flow between particular ports.

- ADVA's ROADM cards require a **power equalization** procedure to be triggered after a cross-connection in the WSS is created. Without power equalization the ROADM card is blocking the signal flow. OpenFlow controller can send a power equalization request to the OpenFlow switch and therefore instruct the switch to equalize optical signal on modules that require such procedure. The equalization is triggered by specifying ports and a wavelength. Equalization is triggered on modules that are located along the internal signal path between these ports and the request is unidirectional.

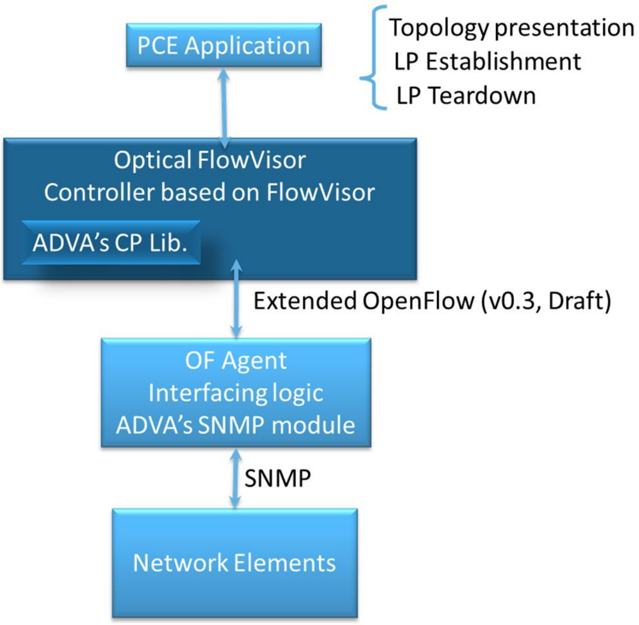


Figure 8.8 Current implementation

8.3.1.2 *OF Agent:*

In an OpenFlow controlled network forwarding functions are performed in the OF switch according to a flow table, and the controller is responsible for the control plane decision like routing, path computation etc. Since many vendors are yet to embrace OF, an OF agent sitting on the network element can be used to provide OF abstractions. One such modular approach is shown in figure below.

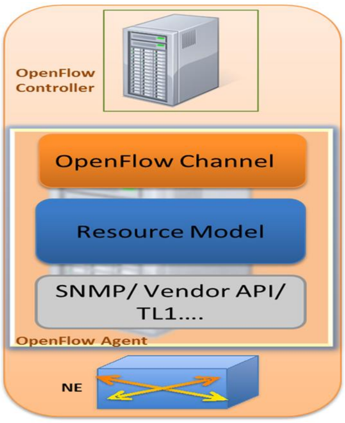


Figure 8.9 OpenFlow Agent

This agent utilizes the NE's management interface (like SNMP, TL1, Vendor API…) to communicate with the data plane. A generic Resource Model can be used to maintain NE's configurations (wavelengths, port capabilities & switching constraints). Resource Model deals with the complexity of the NEs capabilities and represents them to the controller in a generic format. OF agent also includes the OF channel, which is responsible for communication with extended OF controller.

8.3.2 Plane purpose

The FSP 3000R7 supports a fully GMPLS-compliant distributed network control plane. The control plane (CP) allows on-demand real-time provisioning, automatic inventory management and embedded network control. The ADVA Optical Networking control plane functionality gives each network element the capability to discover, control and manage end-to-end transport connections within a network of transport elements.

The primary purposes of the control plane are:

- Network Topology and resource discovery. The CP orchestrates the components inside the system to extend IP-based protocols for auto-discovery which lets network elements automatically discover the topology of the network to which they are connected.
- Path computation. The CP uses methods to enable network components to compute paths through the networks taking into account the constraints on the paths.
- End-to-End Connection Signalling. The CP incorporates methods in order to enable network elements to establish end-to-end provisioned path.
- Service Provisioning. The CP by using methods and components gives the ability to the user to specify the desired characteristics of an end-to-end transport connection path through a network to establish that transport connection in the network.
- Resource Management. The CP incorporates components and methods to enable network elements to discover the transport-layer resources local to the element and to advertise those capabilities to other network elements and to manage reservation and provisioning requests for those resources.

To access to the network elements, Craft Console and Web Browser Console are provided.

The Craft console is a text-based menu system that is resident on the NE and it is accessible by connecting to the serial or USB port on NCU (19200, 8, N, 1).

Craft console provides:

- View all system parameters
- carrying out full system commissioning and configuration
- update software and hardware
- manage user accounts
- access the Linux command prompt

The Web console is a menu system that is resident on the NE. By using an Ethernet connection to the NE the Web console could be operated.

Web console will provide:

- Viewing all system parameters
- carrying out full system commissioning and configuration
- updating software and hardware

- managing user accounts
- using the graphical view to access alarm status

8.3.3 Exposed protocols

ADVA FSP2000R7 uses Data Network Communication (DCN) to carry distributed management, signalling and other communication between NEs in an optical network. DCN is used for:

Management communication

- SNMP communication (traps and Get/Set messages) between NEs and one or more Network Management System (NMS) applications, thus providing remote NE management
- TL1 communication via telnet on the TL1 port (2024) between NEs and one or more NMS applications, thus providing remote NE management
- telnet/ssh communication, providing remote NE management via the local Craft interface (and also providing remote NE raw access via the Linux user interface)
- HTTP communication, providing remote NE management via the web interface

Signalling communication

- OSPF communication among OSPF enabled NEs and other OSPF routers, providing dynamic routing functionality
- OSPF-TE and RSVP-TE communication among OSPF/RSVP enabled NEs, providing control plane functionality

Other operations communication

- file transfer (FTP/SCP) communication between NEs and a server, thus providing remote SW update functionality
- NTP communication between NEs and an NTP server, thus providing time synchronization functionality

8.3.4 Configuration requirements

As previously stated, the ADVA FSP3000R7 can be configured via the Craft console which can be reached via serial or USB port on the NCU module or Web console which can be reached via Ethernet port on the NCU module. To access to the Web GUI an IP needs to be set on the NCU. The IP could be set on the NCU by using Craft console.

Also, the standard TL1 interface is available to manage network elements either by a human or from an operation system. Using TL1, one can manage different parts and aspects of NE such as security, system, configuration etc.

9 Dell/Force10 switch

9.1 Overview

The Dell Force10 Split Date Plane switch is a PowerConnect 7024 with the an hardware module based on the Network Pocessor Unit from Cavium. CN52XX. The hardware module needs to be inserted in the extension slot at the back of the switch. The Cavium processor is running a Debian and its connected to the forwarding switch processor through a 10Gbps link.

9.1.1 Physical box overview

The PowerConnect 7000 Ethernet Switch family is designed for enterprise applications where high performance, high availability, and energy efficiency are key requirements. Operating at wire speed, the 7000 switches deliver up to 160Mpps throughput and a data rate of up to 224 Gbps (full duplex) for both Layer 2 and Layer 3 environments.

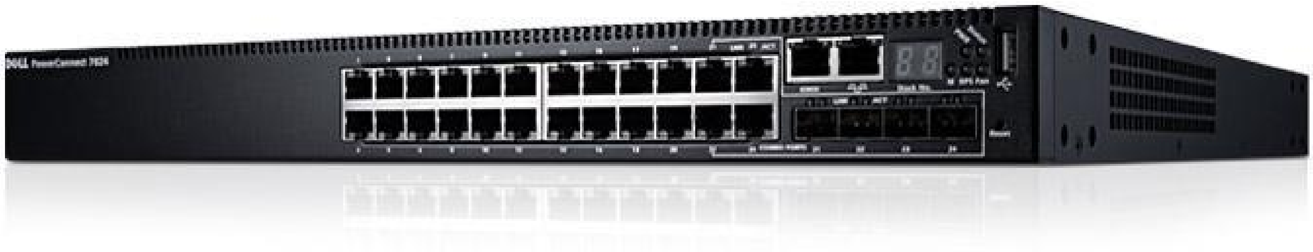


Figure 9.1 Switch external view (ports)

Figure 9.2 Switch external view

The PowerConnect 7024 is the host switch for the Split Data Plane module based on the Cavium CN52XX.

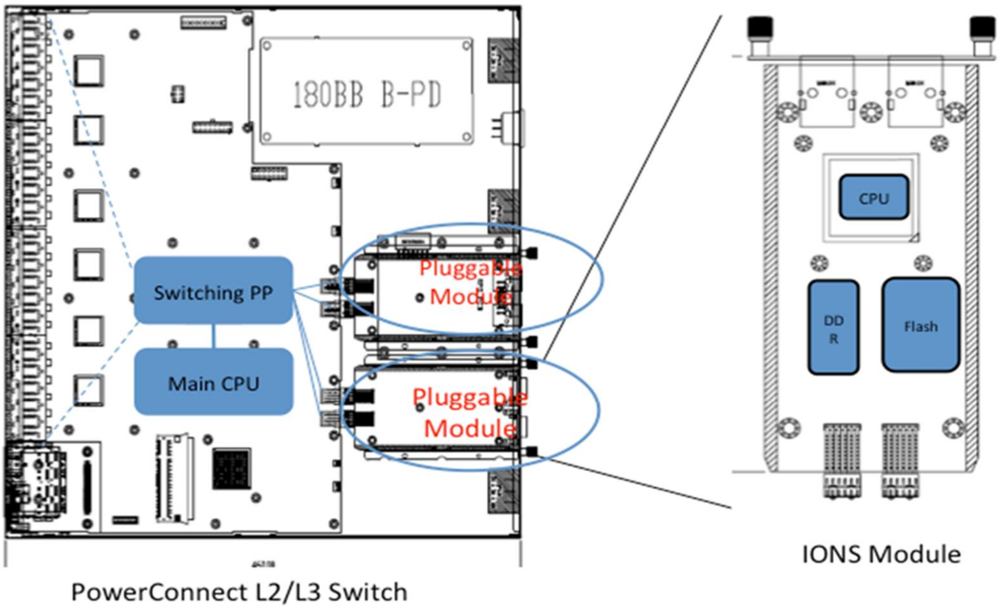


Figure 9.3 Switch schematics

9.2 Data plane

The OCTEON Plus CN52XX family of Multi-core MIPS64 processors targets intelligent networking, storage, and control plane applications in next-generation equipment. The family includes four software and pin-compatible processors, with two to four cnMIPS64 cores at up to 750 MHz speed on a single chip that integrate next-generation SERDES-based networking I/Os including XAUI, along with the most advanced application hardware acceleration. CN52XX processors deliver up to full-duplex 4Gbps application performance.

The CN52XX provides an excellent processing providing up to 7200 MIPS of compute power, high-bandwidth I/O, and essential acceleration for KASUMI encryption, QoS, robust header compression and packet acceleration with very low power consumption.

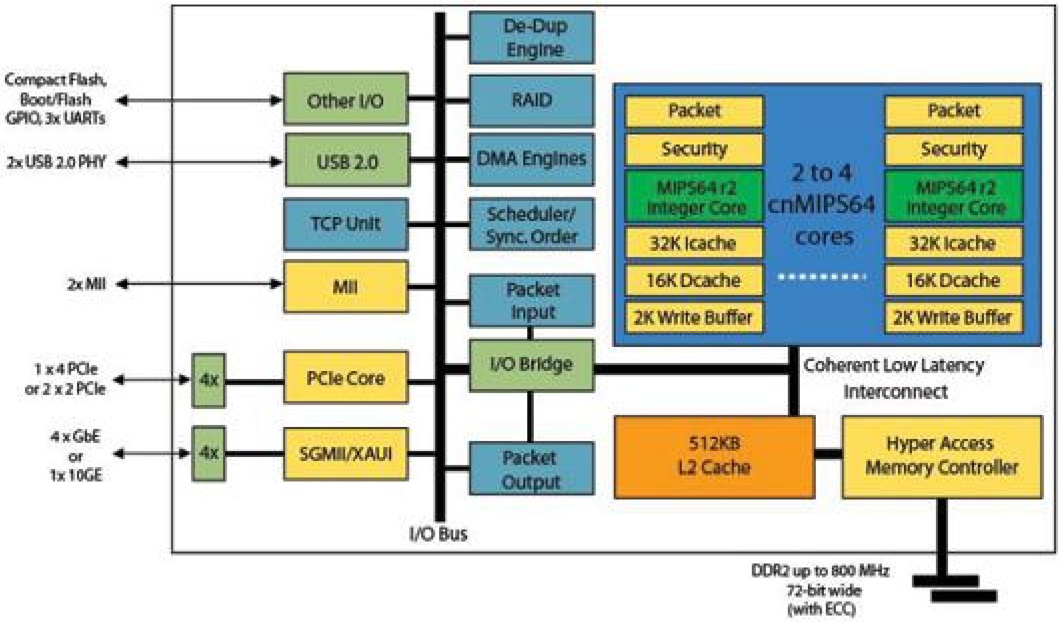


Figure 9.4 Breakdown of switch components

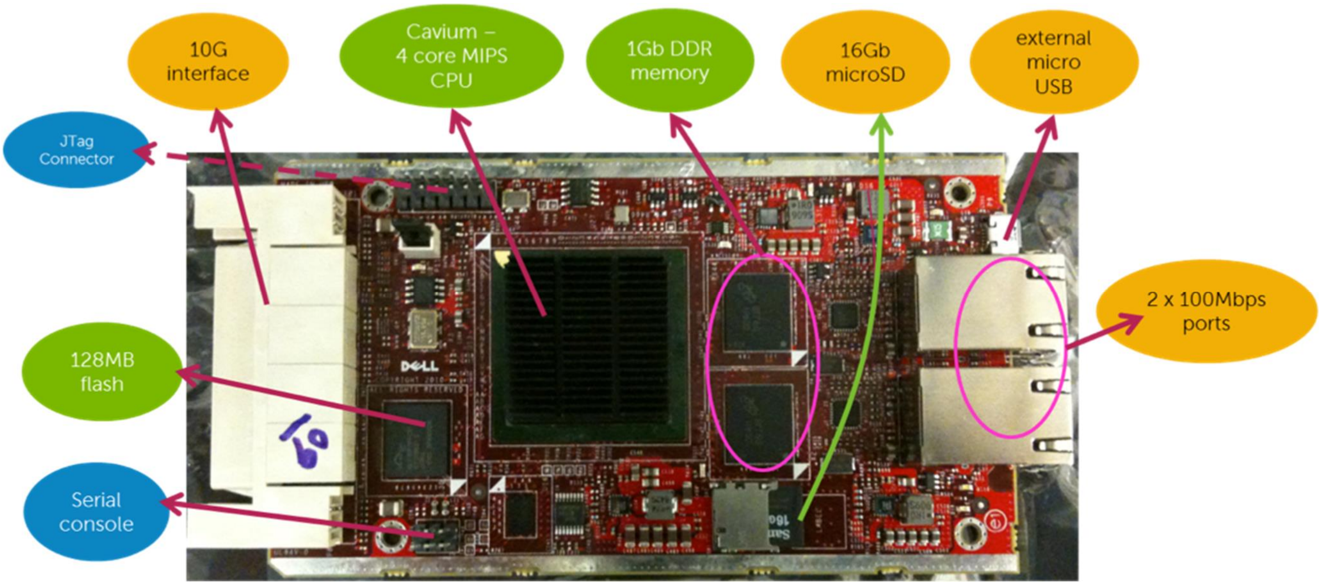


Figure 9.5 Circuit board within switch

9.2.1 Transmission technology

The connection between the switch data plane and the Cavium is done with a XAUI link.

XAUI is a standard for extending the XGMII (10 Gigabit Media Independent Interface) between the MAC and PHY layer of 10 Gigabit Ethernet (10GbE), a concatenation of the Roman numeral X, meaning ten, and the initials of "Attachment Unit Interface".

9.2.2 Switching management rules

The host Switch see the Link to the Cavium as a 10Gbps internal interface. It possible to configure policy based routing to re-route traffic to the Split Data Plane Cavium based module or to use OpenFlow to configure on the Power Connect 7024 host switch and configure flows to be sent to the Split Data Plane module.

The PowerConnect Switch 7024 could be used in two modes:

- As normal operation switch configured through a web interface or CLI and SNMP.
- As an OpenFlow Switch.

9.3 Control/management plane

9.3.1 Plane purpose

The control/management plane is not present by default on the SDPM/Cavium unless it is installed through the Ethernet management port or pushed offline on the internal memory card (microSD). The SDPM is designed to allow and program any control plane or management plane software who could be install or developed and compile with gcc_mips64_octeon for the Linux Debian. The Linux Debian have been the primary host OS choose by Dell Force10, but a Linux BusyBox could be also used. The OpenVSwitch has been already used to run specific usecase scenarios.

Many Debian network applications can be implemented and developed for tunnelling diversity, encapsulation/decapsulation, encrypt/decrypt, Accelarate TCP traffic, compress/decompress, De-Dupe, watermark, Buffer, Guarantee and Encode/Decode.

9.3.2 Exposed protocols

The Split Data Plane module is running on Debian MIPS64 and allows running any specific network application on top of it. Today most of the application works using OpenVSwitch running OpenFlow to the latest specification version.

9.3.3 Configuration requirements

As the split data plane module is based on Cavium if you need to develop specific application, the Cavium SDK is required. You can develop all the application on a Fedora or CentOS using the Cavium SDK simulation NPU environment.

All the booting file system is on micro SD card inserted on the Split Data Plane internal slot, and after booting the Cavium on Debian, you can access it through the module Ethernet management port or the console port.

10 Common hardware themes

The above hardware descriptions give the basic properties of the equipment which is to be configured to use OpenFlow during the Alien project. Many of the pieces of equipment have properties in common. A main goal of ALIEN is the design of a Hardware Abstraction Layer which can be thought of as an intermediate stage between the OpenFlow protocol and the equipment itself. This section attempts to create thematic groupings of equipment which can be treated in largely the same way by the hardware abstraction layer.

Table 10-1 shows which ALIEN hardware fits into which category in the following section.

	Packet processing device	Lightpath device	Point to Multipoint Device	Programmable Network Processor	Physically Reconfigurable System
EZappliance	✓			✓	
GEPON	✓		✓		
NetFPGA	✓			✓	
DOCSIS	✓		✓		
ATCA	✓			✓	✓
Layer 0 Switch		✓			✓
Dell/Force 10	✓			✓	✓

Table 10-1 Assignment of equipment to device types

10.1 Packet Processing Devices

The OpenFlow switch specification [openflow] provides a natural framework to discuss the usual packet processing devices currently found in the networks of many enterprises and ISPs. Hence, it is natural to consider the subset of capabilities that non-OpenFlow-enabled packet processing hardware can provide in the context of ALIEN.

For the purposes of this document, we will define a *packet processing device* as a basic piece of equipment capable of receiving and forwarding network layer packets encapsulated on data link layer frames.

Most devices to be used in the context of ALIEN will use Gigabit Ethernet or 10 Gigabit Ethernet as a layer 2 technology, and IP at layer 3. For those devices providing transport capability, all support the /TCP/UDP/ transport protocols over IP.

Given the almost universal support for IEEE 802.3 in ALIEN testbeds, most equipment to be considered also supports Logical Link Control (LLC) functions such as VLANs (Virtual Local Area Networks) and trunking. In addition, by leveraging the multicast capability of IEEE 802.3, multicast/broadcast addresses can be used to provide point-to-multipoint connectivity in addition to explicit frame copying to multiple output ports.

The cornerstone upon which forwarding decisions are taken in OpenFlow is the 5-tuple, defined as a combination of a pair of source/destination IP addresses, a protocol number, and a pair of source/destination TCP/UDP ports. All packet processing devices to be used in ALIEN support this definition, and many of them can be configured to apply differentiated treatment to flows (examples include differentiated forwarding for QoS and filtering/dropping).

In addition, most packet processing devices in ALIEN support management and configuration interfaces. Typical protocols include the use of a proprietary CLI (Command Line Interface) through a serial port, proprietary packet-based protocols, and standard packet-based protocols such as SNMP (Simple Network Management Protocol).

One important way in which these simple packet processing devices differ from Layer 0 (lightpath) devices is that, when building forwarding rules using OpenFlow, the potential number of 5-tuples associated with a given physical interface is so extremely large that it could be considered unlimited. This is not the case with layer 0 devices, which only support a small number of layer 0 channels (e.g. DWDM wavelengths). This may be important when considering which OpenFlow functionality to include in these devices.

10.2 Lightpath devices

This is not really a group of devices within ALIEN but consists only of the Layer-0 switch which has fundamentally different properties from the devices in section 10.1. However, this is representative of a larger number of pieces of optical equipment which are capable of configuring paths but which do not understand Ethernet or IP packet headers.

Among all alien devices, layer-0 or optical devices (ADVA) have totally different features in terms of data plane and control plane. Unlike other alien devices which are categorized as packet switched devices, layer-0 devices are categorised as circuit switched devices.

Generally, the control mechanism in IP networks is tightly linked to the packet forwarding task and it is fully distributed in each router and switch in the network. On the contrary, circuit switched or transport networks have completely separate control and data plane. The control plane has no visibility of data plane and the provisioning and management of the network is done manually through provider’s management system.

Conceptually, by making an abstraction of cross connection tables in circuit switched devices, it is possible to create a flow table based on them. Flows can be defined based on timeslot SONET/SDH or wavelength DWDM or fibre switching. The abstraction will remove the underlying distinction between packet and circuit switched networks and it regards them as flow-switched network. but unlike the packet switched devices, because there is no visibility on data path and accordingly there is no packet forwarding to controller, therefore there is no backend on circuit switched devices and decisions have to be made prior to creating end-to-end path or circuit (i.e it is pro-active not reactive) and controller is responsible for creating and tearing down the connections.

Pursuing this concept, an extension has been added to OpenFlow ver. 1.0 switch specification to support circuit switched devices in OpenFlow [ref.v0.3]. The extension covers components and basic functions on circuit switches based on TDM, WDM or fibre switching. The flow fields contains input/output port, in/out wavelength, virtual port and in/out TDM time slot. Considering that optical devices fall into circuit switched network category creating a lightpath which is an end-to-end tunnel, certain constraints such as guaranteed wavelength continuity, optical power , etc. have to be taken into account.

Following the OpenFlow extension guideline, the extension adds capability to OpenFlow for supporting circuit switched networking. The features are listed below:

- Switch features
- Port structure
- Port status
- Cross-connect structure
- Circuit flow add/modify/delete
- Circuit flow action types
- Error messages

The abstraction layer has to be designed in order to meet OpenFlow extension requirements.

10.3 Point to multi-point devices

Some devices have an inherent asymmetry between ports and have an inherent single port at the “head” and many ports at the “tail” end. The phrase used for “head” and “tail” depends on the device used. In the case of the GEPON the “head” is the OLT and the “tail” devices the ONUs and in the case of DOCSIS the “head” is the CMTS and the “tail” devices the CMs. Throughout this section the phrases “head” and “tail” will be used to avoid ambiguity or repetition. This class of devices includes the GEPON, DOCSIS and (depending on configuration) the Layer 0 switch although the nature of the layer 0 switch depends on its physical configuration (see section 10.5).

The head device is usually the more capable device. In these devices the input at the “head” end is broadcast to the other connected ports and those ports select the portion of the input relevant to them. Any transmission between two “tail” ports must be via the “head”. In the case of DOCSIS and GEPON then tail devices are configured by commands to the head end and the overall device (head end and tail devices) can be considered together as a single distributed switch. In some cases, depending on configuration, a packet entering at one tail device and destined for another tail device may, in fact, be propagated up to a layer 3 device beyond the “head” end before being routed back to its correct destination.

The tail end devices typically have lower functionality and packets presented directly to them may not be so flexibly processed. For Open Flow then this presents two challenges. Firstly flow patterns which may be sensible on a particular set of ports may be problematic on a different set of ports. The device needs to be able to communicate the fact that the “head” end is special to the Open Flow controller. The capabilities for processing a packet will depend crucially on whether the packet enters a port at the “head” or “tail” of the device.

10.4 Programmable network processors

Some of the alien hardware platforms (EZchip's NP-3 network processor, the NetFPGA board, and the Cavium OCTEON-I network processor units available in the Dell/Force10 and ATCA device) introduced in the preceding sections contain programmable hardware units that can be adapted to a wide range of network processing tasks. These hybrid devices combine (at least partially) the flexibility of software implementations on general purpose CPUs and the performance characteristics of dedicated networking chip sets in terms of packet forwarding speeds. Flexibility is a mandatory prerequisite in order to evolve the device's functionality: One of the open issues addressed by project ALIEN is the restriction of OFELIA to OpenFlow v1.0 and how to move beyond this initial, yet outdated version of OpenFlow's specification (At the time of writing the actual OpenFlow version defined by the Open Networking Foundation is v1.3.1). Reprogrammable platforms allow the integration of newer versions of OpenFlow on a device or extending an existing implementation with new capabilities, e.g. additional protocol match and action definitions.

A network element consists typically of a management part and a backend for conducting the work of forwarding and processing packets. OpenFlow supports heterogeneous backends in its data model, as it allows definition of multiple non-uniform tables in a data path element's packet fwd/proc pipeline. A table represents an abstract building block capable of matching on a packet's fields and applying certain actions like setting header fields, push and pop operations of certain header tags, or sending the packet to a port's set of outgoing queues. The authors of OpenFlow assume a mapping of available hardware units to these abstract tables within OpenFlow, thus, in principle, granting control plane designers direct access to specific hardware units and their capabilities. A modular data path architecture developed within project ALIEN should support the data path developer in this task of mapping available hardware blocks to OpenFlow tables. A table definition in OpenFlow comprises various parameters including among others the set of protocol field available for matching, the set of instructions supported, and the size of a table. However, writing high performance SDN control modules requires additional information like internal connectivity among tables within the pipeline. At present, OpenFlow is lacking such advanced information, which must be documented separately by the device manufacturer.

Surprisingly and in contrast to the data model defined by OpenFlow for programming network elements, an OpenFlow-compliant data path element is mainly a static entity, as it contains pre-defined knowledge about protocol types, matching fields, and processing actions. The first hardware enhanced prototypes of OpenFlow were based on immutable ASICs, preventing control plane developers effectively from applying changes to this static environment. However, with the advent of programmable network processor units we can rethink strategies to enhance a data path by new protocol types, matches, and actions dynamically at run-time. Each version of OpenFlow has seen the introduction of new protocol types like MPLS in version 1.1, IPv6 in version 1.2, or GRE in version 1.3. Without doubting the usefulness of these extensions, a more generalized scheme for introducing support for additional protocols seems to be necessary. A data path element may contain only a limited subset of base protocols (or even no protocols at all) once it has concluded its boot sequence. When establishing a control connection to the control plane, a control module may load a description of a protocol type including the protocol's structure (fields, matches, actions) onto the data path element, before installing new flow controlling entries and thus making use of these protocol definitions. A Hardware Abstraction Layer may provide a generalized model of a packet as a sequence of bytes for defining protocol structures and may contain some form of intelligence to map such protocol structures to the existing hardware units in an effective manner.

Furthermore, for such dynamic scenarios we have to take into account constraints induced by the hardware platform itself, e.g. Cavium's OCTEON-I network processor unit, when executing in stand-alone mode, cannot change the running process image without rebooting. One alternative for dealing with such constraints may be the introduction of a common virtual machine for packet operations that is executed by the process image.

10.5 Physically reconfigurable systems

There are alien hardware platforms (i.e: Dell/Force10 switch, ATCA device and the layer 0 switch) that are composed of different boards or cards which are connected together in order to provide full, modular networking system. These systems can be highly reconfigurable but this must be done physically by removing and replacing physically independent components. Within that kind of a platform, each component offers dedicated capabilities and performs specific traffic processing roles. All components work quite independently from each other and could be easily replaced depending of network operator needs and overall platform role in the network. The system components cooperate together by exchanging transport network traffic in the form of Ethernet frames or opaque optical transmission depending on technical nature of the entire system.

Transport plane interconnectivity between system components can be controlled thanks to existence of configurable switching elements. For example, ATCA boards contain switches equipped with connections towards other boards or internal elements (specific cards, embedded PC). Similarly, the Dell/Force10 switch has internal ports within the base PC 7024 switch to send packets to card extensions (i.e.: Octeon-based cards). The layer 0 switch has a ROADM device which rediresct light to optical amplifiers within the DWDM ring, to electrical/optical converters for add/drop connections or to other kind of optical components. All these switching elements may be controlled by internally or externally installed software like OpenFlow agent; PC7024 switch by default supports OpenFlow protocol but also there are OpenFlow implementations which are able to control ATCA boards and ADVA ROADMs.

A common feature of some physically reconfigurable devices is the Cavium Octeon processor (this is present in the Dell switch and the ATCA device). These cards offer very specific functionality and are used to perform non-standard or heavy computation operations on network packets. They contain a single port used to exchange Ethernet frames with other parts of the device. We can distinguish three basic scenarios for programmable packet processors cards: 1) the programmable packet processor generates packets and sends them out by the port. 2) the programmable packet processor receives packets from the port, transforms packets and sends them out by the same port. 3) the programmable packet processor receives packets from the port and consumes them. Because programmable packet processors are designed to perform any work with packets within network layers 2-4 or even with application content of the packets so such programmable cards may be fully functional and independent network nodes but equipped only with one transport plane port. Programmable cards offer also the possibility to deploy non-packet processing software which could be for example an OpenFlow agent or other kind of card management software.

A challenge for Open Flow and the Hardware Abstraction Layer for these devices is that they offer reconfigurability which cannot be controlled automatically with an Open Flow controller – the device must be physically reconfigured to offer that possibility. A second challenge is that if the Open Flow controller is to be aware of the device’s capabilities and properly control flow within the device, it must be aware of the current physical configuration. A physical change to the configuration of the device must be reflected in a change in the interface to Open Flow. One possibility is to have parts of the device separately as Open Flow controlled devices.

11 Conclusions: Future directions for HAL design

The simplest framework that accommodates both frame/packet processing equipment and lightpath equipment is one in which there is a single primitive, forward, that associates a traffic flow label FA at an input port PA with a traffic flow label FB at an output port PB. In the case of frame processing equipment, FA = FB (the 5-tuple is not overwritten) and forwarding is a matter of simply deciding what the appropriate port for the next hop is. For lightpath equipment, FA is the incoming lambda, PA is the incoming port (fiber), FB is the outgoing lambda, and PB is the outgoing port. In this case, forward establishes the PA:FA to PB:FB mapping like required by e.g. GMPLS. Of course, there is no need for the OpenFlow controller to orchestrate paths manually; the natural mechanisms of the optical network can be leveraged for this. In this case, the entire optical path can be modelled by OpenFlow simply as a pair of ports.

OpenFlow is generally triggered by parsing 5-tuples, comparing to the currently active 5-tuples in the forwarding table, and then either forwarding or sending to the OpenFlow controller as appropriate. In this generalised case, it could be argued the OpenFlow could react not specifically to 5-tuples but to new port:flow label combinations. In the case of lightpath equipment, this could mean “previously unused lambdas” being received in a given port. This use case, although relevant for completeness, does not correspond to a situation in which optical networks are commonly used and hence it may not be of interest. If this is the case, it may be that we decide that all OpenFlow triggers are generated by frame/packet processing equipment and it is only the OpenFlow controller that sets up specific lightpaths in response to these events. This idea goes well with an architecture in which an OpenFlow-enabled edge controls an optically transparent, high-capacity core which is OpenFlow agnostic. This could be achieved simply by programming the OpenFlow controller logic to produce specific outputs directed to the Optical NCU (Network Control Unit) when receiving given inputs. OpenFlow’s data model is limited to an Ethernet port abstraction. For optical devices, we see another layer that sits below the logical Ethernet layer. In fact, this is the same discussion as in GMPLS. For OpenFlow, it seems necessary to extend the existing data path model and its port abstractions towards a more general framework, where we can also describe lambdas or fibres. Btw. the same problem can be seen for wireless networks: A wireless port on a base station may contain various logical connections (the same with Ethernet), but for today’s Wireless LANs you see typically also an authentication state associated with each “connection” (=a virtual port). (The same can be seen in Ethernet with IEEE 802.1x, though).

Section 10 introduces some challenges for the design of the hardware abstraction layer. In particular, the following problems occur:

- the layer 0 switch (and similar optical only devices) is not compatible with the Ethernet and IP framework assumed by OpenFlow for flow identification.
- Point to multipoint devices have an inherent asymmetry which means that the capabilities of the device are not the same for a flow from A to B and a flow from B to A.
- Physically reconfigurable devices have capabilities which change depending on the layout of the board. This layout can be modified but only by physically moving boards in the hardware. The open flow controller must be aware of the current layout (if not the possibility of changing this).
- Programmable network processors have very general abilities. A limitation on the flexibility of the HAL may limit what the programmer can do using such a device while remaining OpenFlow compatible.

These challenges will be addressed in the project while designing the HAL and corresponding modules.

12 References

[EZchip] EZchip Technologies Website, http://www.ezchip.com/

[EZapp] http://www.ezchip.com/p_ezappliance.htm

[IEEE802.3] "Part 3: Carrier sense multiple access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications", IEEE Std 802.3™-2008, IEEE Computer Society

[openflow] http://www.openflow.org/documents/openflow-spec-v1.1.0.pdf

[netfpga.org] http://netfpga.org

[netfpgaguide] http://wiki.netfpga.org/foswiki/bin/view/NetFPGA/OneGig/Guide

[netfpgaregisters] http://wiki.netfpga.org/foswiki/bin/view/NetFPGA/OneGig/RegisterMap

[netfpgareferenceNIC] http://wiki.netfpga.org/foswiki/bin/view/NetFPGA/OneGig/ReferenceNICWalkthrough

[netfpga-buffer] http://wiki.netfpga.org/foswiki/bin/view/NetFPGA/OneGig/BufferMonitoringSystem

[netfpga-modelsim] http://www.mentor.com/products/fv/modelsim/

[PL-LAB] PL-LAB portal, http://pl-lab.iip.net.pl

13 Acronyms

ASIC – Application Specific Integrated Circuit
EPON – Ethernet Passive Optical Network
GEPON – Gigabit Ethernet Passive Optical Network
MAC – Media Access Control
OAM – Operations Administration and Maintenance
OLT – Optical Line Terminal
ONU – Optical Network Unit
PON – Passive Optical Network
QoS – Quality of Service
SLA – Service Level Agreement