SPARC

ICT-258457

Deliverable D2.1

Initial definition of use cases and carrier requirements

Editor:	F.-J. Westphal, Deutsche Telekom (DT)
Deliverable nature:	Report (R)
Dissemination level: (Confidentiality)	Public (PU)
Contractual delivery date:	M5
Actual delivery date:	M5
Version:	3.0
Total number of pages:	62
Keywords:	Use Case, Carrier Requirements, Aggregation Network, Backbone Network, Mobile Backhauling, Software-defined Networking, Seamless MPLS, Data Center, Load Balancer, Split Architecture, GMPLS, OpenFlow, Network Virtualization, Resiliency, Carrier Grade, Multi-layer, OAM, Virtual Port, MPLS, Hierarchy, Topology, Scalability, IP/MPLS, OTN

Abstract

This first deliverable of work-package "Use cases and business scenarios" provides an initial set of use cases and derived requirements from these cases. Use cases are selected from three different areas - access/aggregation networks, data center, IP backbone and multilayer - and are covering the whole environment of future telecommunication and ICT carriers/operators. The derived set of carrier requirements is the basis for the other technical work-packages, and especially of importance for the definition of the architecture and the implementation work within the scope of the project.

Disclaimer

Imprint

[Project title]	***Split Architecture***
[short title]	***SPARC***
[Number and title of work package]	***WP2 – Use Cases and Business Scenarios***
[Document title]	***Initial definition of use cases and carrier requirements***
[Editor]	***Fritz-Joachim Westphal***
[Work package leader]	***Fritz-Joachim Westphal***
[Task leader]	***Mario Kind***

Copyright notice

Executive summary

Besides analyzing new business opportunities of a Split Architecture, the objective of work package 2 is the description of use cases and definition of carrier-grade requirements, derived from these use cases. Both will be provided as input to the development of a Split Architecture in work package 3. The first and initial attempt to fulfill the objective is covered in this deliverable.

The deliverable gives an overview of the different use case areas, which are selected to cover all important aspects of a carrier environment as defined for next-generation-networks by ITU-T. These use case areas are:

- Access/aggregation domain of public telecommunication networks including mobile-backhaul, software-defined networking and dynamic control composition
- Application of Split Architecture in the data center domain
- Multi-layer networks with specific characteristics for packet and circuit-based technologies and a detailed analysis of IP/MPLS transport via OTN in the backbone

In the access/aggregation domains of most carriers, an Ethernet-based aggregation is used to provide services for residential and business customers. State of the art access/aggregation solutions and next generation extension of MPLS towards this network domain are described and corresponding requirements are derived. Implementing a Split Architecture by means of OpenFlow in those extended MPLS domains may give the chance to have a centralized control and to make use of commodity hardware, without influencing the interworking between the different MPLS domains of the telecommunication network. Moreover OpenFlow would allow virtualization and thereby support service separation within the responsibility of one operator or multi-operator scenarios. As mobility is more and more essential, the infrastructure to provide wireless services is a must and the aggregation network is also used to guarantee mobile backhauling. 3GPP tries to integrate the involved types of networks to a common network design and mobility approach. At least handover between base stations would extend requirements on the network in this approach and it would be difficult to support distributed approaches, as are used to implement content delivery networks. Forward-looking backhauling solutions have to take those aspects into account. Splitting the data and control planes might allow the modification or extension of functions on devices during operation and thereby would allow software-defined networking, i.e., in the context of IEEE 802.11 compliant devices, to control or enable functional extensions by a centralized network application on a wireless edge device. In addition to requirements on functions and characteristics, a first set of preconditions on dimensioning and scalability oriented towards a typical Western European aggregation area is provided.

The data center domain use cases are focused on increased network and server utilization through efficient load balancing on different layers in the network stack. Moreover it is addressed to reduce the large amount of complex networking hardware needed in existing solutions, reduce complexity in managing multiple data center customers and improve the energy efficiency of data centers. Support of virtualization of switches, routers and servers in a data center seems to be a valuable use case for OpenFlow. It allows for flow-based resource allocation inside the network and can be utilized for virtualization of network resources. Another case with respect to cloud applications is to use OpenFlow on the interworking of network and data center domains. Furthermore, networks in data centers have to be designed for peak traffic. In off-peak situations, part of the infrastructure could be put to sleep mode to save energy. This implies rerouting and a concentration of processing resources. To support this concentration, traffic engineering and associated migration of virtual machines could be enabled by OpenFlow.

OpenFlow basically collapses several network layers (L2-L4) into one. The attempt is to extend the Split architecture to the optical layer, based on OTN technology like it is used in today's regional and backbone networks. Unfortunately the optical layer differs in characteristics by being circuit switched. Solutions for a multi-layer operation of packet and circuit-switching under a common Split Architecture are provided. Here the key is the level of detail of information about lower layers that should be exposed to the control plane and controllers of a Split Architecture. However, the degree of information that a lower layer exposes to a higher should be left to the implementer and could cover the scope from overlay to peer model. Most of this information will be created by controllers based on available information and alarms coming from OpenFlow-enabled hardware. This means that the Split Architecture and OpenFlow interface will have to be extended with respect to the specification of interface types, the handling of heterogeneous forwarding abstraction, the processing interface and the configuration of termination functions, like OAM.

Based on the three use case areas, 67 requirements have been derived. They were prioritized with respect to overall importance, fulfillment in existing architecture concepts and/or existing implementations and their relevance for one or more use cases. In the context of SPARC four groups of requirements could be identified. The first group covers all modifications and extensions for the data path element or the Split Architecture itself. The other three groups deal with needed extensions of carrier-grade operation of ICT networks. The aspects related to the operation of an ICT network are authentication, authorization and auto configuration; OAM in the sense of facilitating network operation and troubleshooting, network management, security and control of the behavior of the network and protocols.

List of authors

Organization/Company	Author
DT	Mario Kind, F.-Joachim Westphal, Andreas Gladisch
EICT	Andreas Köpsel, Ahmad Rostami, Hagen Woesner
EAB	Sonny Thorelli, Zhemin Ding, Catalin Meirosu
ACREO	Pontus Sköldström
ETH	András Kern

Table of Contents

List of figures and/or list of tables

1 Introduction

1.1 Project context

The project SPARC "Split architecture for carrier-grade networks" aims towards implementing a new split in the architecture of Internet components. In order to better support network design and operation in large-scale networks millions of customers, high automation and high reliability, the project will investigate splitting the traditionally monolithic IP router architecture into separable forwarding and control elements. The project will implement a prototype of this architecture based on the OpenFlow concept and demonstrate the functionality at selected international events with high industry awareness, e.g., the MPLS Congress.

The project, if successful, will open the field for new business opportunities by lowering the entry barriers present in current components. It will build on OpenFlow and GMPLS technology as starting points, investigating if and how the combination of the two can be extended and study how to integrate IP capabilities into operator networks emerging from the data center with simpler and standardized technologies.

1.2 Relation with other work packages

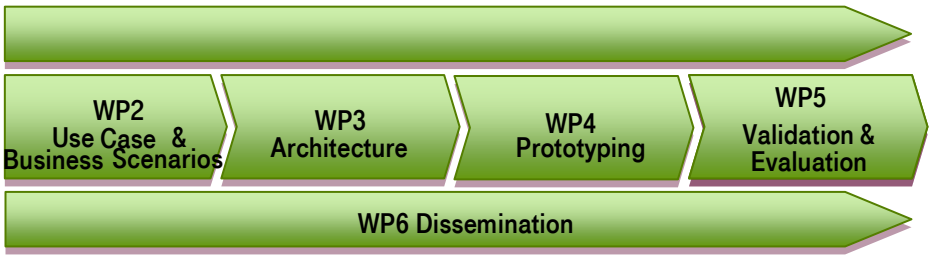

Figure 1: Relation of SPARC work packages

In the "workflow" of the dedicated work packages of SPARC, WP2 sets the stage for the other technical work packages, namely WP3 (Architecture), WP4 (Prototyping) and WP5 (Validation & Performance Evaluation). WP2 will define use cases and requirements. Based on those use cases and additional generic requirements, an initial study for a Split Architecture in relation to an enhanced version of OpenFlow will be done by WP3. In addition, this architecture will be evaluated against certain architectural trade-offs. WP4 will implement a selected sub-set of the resulting architecture, whose feasibility will be validated in WP5. WP6 will disseminate the result at international conferences and publications. The schematic of the workflow is shown in Figure 1.

1.3 Scope of the deliverable

Split architecture defines a paradigm shift in network architectures, offering more freedom to deploy and operate carrier-grade networks. However, carrier-grade networks are characterized by some specific challenges including provisioning of a large coverage area, providing broadband access in urban and rural areas, QoS-enhanced transport of (triple-play) services towards (residential) customers. The objectives of work package 2 are the definition of use cases and, based on these use cases, the derivation of requirements for a carrier-grade Split Architecture. Split architectures offer more freedom to network operators, e.g., to deploy new control plane protocols, to achieve higher cooperation among network management entities, to improve energy efficiency, and network utilization. WP2 focuses on various use case areas, namely access / aggregation (incl. mobile backhaul), data center, virtualization, IP transport in the backbone, multi-layer approaches, and selects the most relevant ones for a more detailed analysis. This includes a detailed look into current state of the art of the different architectures. WP2 started with the studies and identification of the operator driven use cases for large-scale carrier-grade networks. Requirements are derived from these use cases and meant to feed work package 3 for the initial protocol and architectural study. In addition to the results documented in this deliverable, work package 2 will continue to study use cases and may come up with additional requirements.

2 Use cases

2.1 Introduction and overview of use case "areas"

To introduce the different use case areas covered in this deliverable it is a good precondition to remember the essentials of OpenFlow. The OpenFlow concept introduces a fundamental split of control / data plane, and processing functions. That enables operators to evolve data and control plane independently. The potential benefits are based on the following four fundamental characteristics:

- OpenFlow introduces a flexible forwarding concept using a packet's entire (multi-layer) header as tag for identifying packets as part of a flow of related packets,
- OpenFlow enables the definition of sophisticated network applications within the control plane that have the ability to extend basic L2, IP, or MPLS-based forwarding mechanisms by means of content aware forwarding,
- As opposed to tunnel-based architectures like MPLS, OpenFlow allows the individual handling of packets in a flow, even while a stream of packets is flowing through the network.
- OpenFlow allows a smooth integration of enhanced processing capabilities into the network to introduce advanced networking functionality like network coding.

An important aspect of this deliverable is to select use cases in a way that all important areas of a carrier environment are covered. The definitions given by ÍTU-T are good references that can be used to describe a future-proof carrier environment. The corresponding recommendation is ITU-T Y.2001 [35]. Figure 2 is taken from an ITU-T presentation based on recent results of ITU-T study group 13. Despite the more traditional network domains covered by the "Transport Stratum", like user network, access network ("access dependent"), aggregation network ("distribution function") and core transport network, the "Service Stratum" is the key to providing NGN-based services. The latter brings the data center into the game.

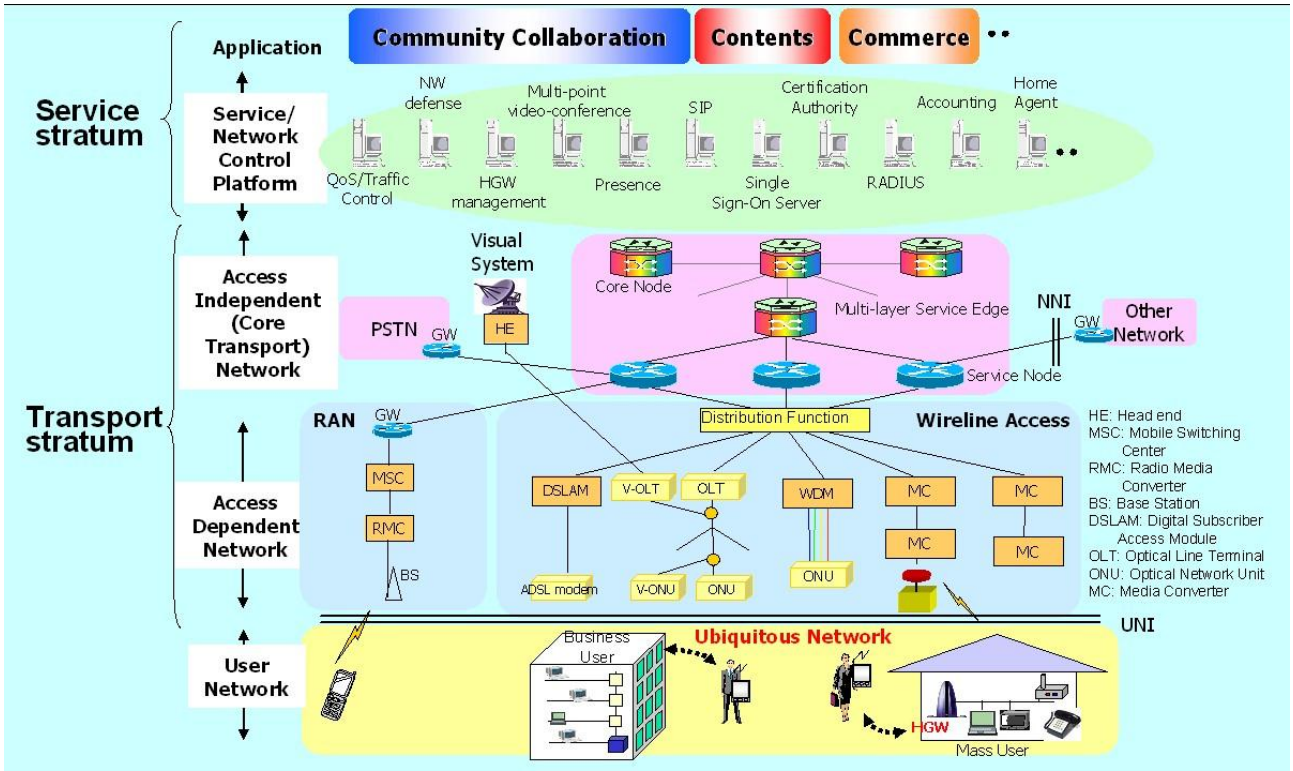

Figure 2: Example of NGN realization based on definitions of ITU-T Y.2001

A simplified illustration covering the most important use case areas in a carrier environment, in line with the definitions of deliverable D3.1, is given in Figure 3:

- Aggregation/access domain including mobile-backhaul, and virtualization in order to isolate different service groups or establish a shared use of infrastructure in between different operators

- Application of Split Architecture in the data center domain could be manifold. Two of the most interesting aspects are load balancing and the support of session / server migration for coordinating switch configuration and assigning server resources in a more dynamic way
- Telecommunication network design is based on multiple layers. While OSI introduces a layering of functions required for networking, each layer in a telecommunication network corresponds to a different technology as in IP/MPLS/Ethernet/SDH/WDM networks. However, often this design principle is not followed consequently. Since OpenFlow inherently flattens the network layering structure, it provides options to simplify optimizations and signaling for multi-layer traffic engineering purposes and multi-layer resilience.

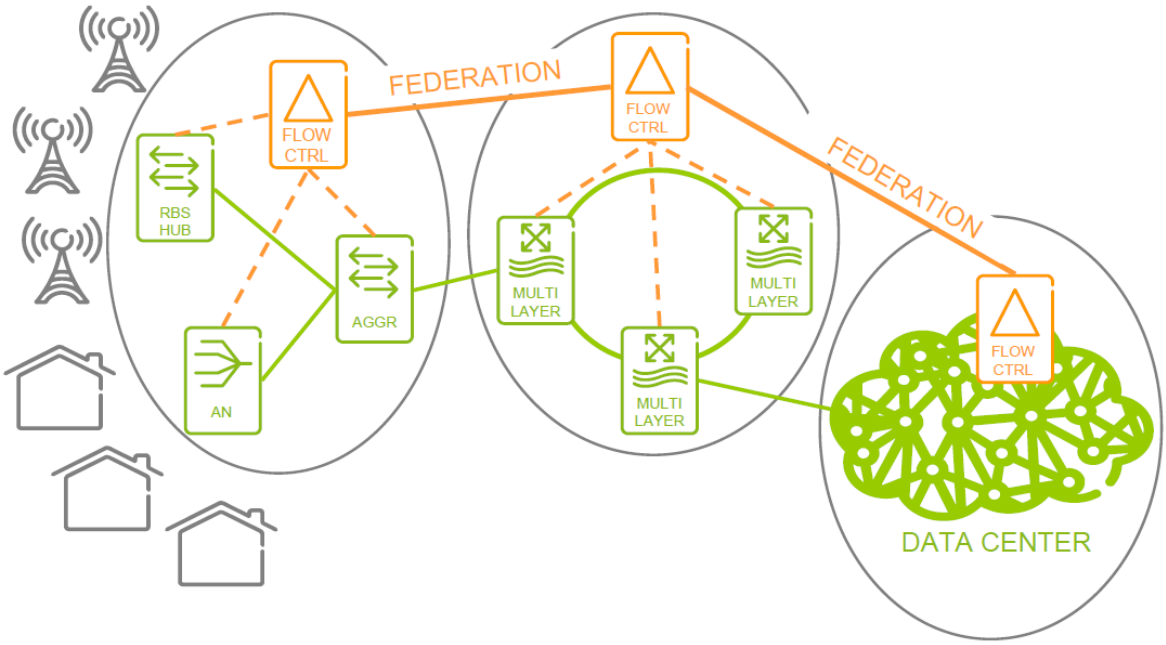

Figure 3: Illustration of the three use case areas: 1) access / aggregation incl. mobile backhaul, 2) multi-layer incl. optical layer and IP backbone transport, and 3) data center.

2.2 Use case 1: Access / aggregation domain of carrier network

2.2.1 General description

In state-of-the-art aggregation networks in use today, Ethernet-based aggregation is used for providing triple-play residential services (see BBF TR-101 [1] for a general overview) as well as dedicated business services, as defined by the Metro Ethernet Forum. The MPLS-based next generation of aggregation networks will extend this set of services with support for mobile backhaul, fixed mobile convergence (FMC), enhanced scalability and security mechanisms (see BBF TR-144 [9] and BBF WT-145 (a.k.a. BBF TR-101bis) [10] for additional details). A unified MPLS-based solution is expected to simplify network management and operation, in order to provide various connectivity abstractions (L1/TDM, L2/ATM/Ethernet/HDLC, L3/IP) suitable for a wide range of services, and to offer sophisticated OAM functionality. In addition to the introduction of MPLS in the aggregation domain, ideas for a joint, MPLS-enabled aggregation and core domain have emerged recently (e.g., IETF draft about Seamless MPLS [2]). Introducing an MPLS-based transport framework promises a separate evolution of transport and service architecture, an expectation that resembles OpenFlow's approach of an independent evolution of control and data plane remarkably.

Although a number of recommendations have been defined by industry fora for unified access / aggregation network architectures, existing deployments are characterized by a wide variety of heterogeneous access / aggregation network architectures. Thus, identifying a common view on the state of the art is challenging. In this document, we assume an advanced MPLS-based aggregation domain as state-of-the-art. Offered services and capabilities of this architecture define the minimal threshold, which a potential competing architecture must offer. A comprehensive comparison of pros and cons of OpenFlow and MPLS principles is outside the scope of this deliverable. However, we will focus on some of the key benefits of OpenFlow and how a future access / aggregation domain could benefit from these. Instead

of discussing a single access / aggregation use case, we will highlight the following four aspects for introducing OpenFlow in a future access / aggregation domain:

Multi-service / multi-provider operation

One major trend driving the evolution of access / aggregation domains in the mid and long term is the imperative to increase available bandwidth at the customer site. The introduction of fiber-based access systems will burden operators with significant capital expenditures. Another trend can be observed in the wholesale market. Driven by commercial opportunities for selling bit pipes and new regulatory constraints emerging in the political arena defined for ensuring fair (and open) access for competitors, the shared use of physical access / aggregation infrastructures is gaining momentum. A constraint for shared use is an efficient isolation of slices so that control planes and architectures deployed by different operators are shielded effectively from each other and any intervention between operator control planes is avoided. However, the notation of virtualization of physical access / aggregation infrastructure could be also used to separate the operation of different entities within the organization of one operator.

Mobile backhaul

Mobility is an essential part of today's work and life and is a basic part of ITU's definition for NGN Y.2001 [35]. Usage scenarios are shifting more and more from wired to wireless mobile access. Today, four different types of mobile networks are available: (a) 3GPP for local and wide area mobility, (b) WLAN for local nomadism, (c) meshed networks for local and small area mobility and (d) other IEEE MAN technologies covering nomadic mobility scenarios. While 3GPP defines a full architecture, the latter three are very similar and do not provide sophisticated mobility or QoS support, IEEE-based standards define wireless access technologies limited to layers 1 and 2 of the ISO/OSI stack and thus, lack a fully specified mobility architecture. 3GPP tries to integrate all types of networks into one common design and mobility concept. In addition, the Broadband Forum defines the access / aggregation network requirements and identifies solutions. Currently there are different problems within the envisioned unified solution. For example, the support of handover between mobile base stations that will result in extended network requirements for delay and bandwidth. In addition, more and more applications demanding QoS, like IPTV or streaming video, are available in the mobile domain. Future mobile backhaul solutions have to take all these aspects into account.

Software-defined networking (SDN) application in context of IEEE 802.11 compliant devices

One of the key benefits of OpenFlow's split of data and control planes is the ability to add, update or extend specific functions on devices during operation. In OpenFlow, three main components have been identified: control, forwarding, and processing. Software-defined networking may be adopted for several control plane functions: beyond flow routing, it is also useful for implementing enhanced adaptation and termination functions, monitoring and fault management, or security functions (e.g., authentication and authorization). This would allow the distribution of functions of specific network devices to different locations in the network and enable a separate evolution of these parts.

Though moving advanced control plane architectures to a separated controller seems attractive, some mandatory functionality must remain on the data path elements either due to tight timing-constraints or performance thresholds, e.g., monitoring or enhanced adaptation and termination functions for interworking on edge devices. A data path element in an OpenFlow-enabled domain should be able to provide appropriate functionality (e.g., to act as a wireless edge device, like an IEEE 802.11 compliant access point), when appropriate hardware functionality has been built in. However, the OpenFlow API should be extended so that such extensions can be easily controlled by a network application suited to controlling wireless operation.

Dynamic Control Composition

Today's networking landscape is characterized by a considerable diversity and heterogeneity of management frameworks. A Split Architecture enables not only a smooth evolution and deployment path; it may also pave the way for a smooth interaction of different management and control frameworks of heterogeneous devices. In an access / aggregation environment equipment on customer premises and devices deployed by network operators jointly carry user-destined or user-generated traffic. A Split Architecture enables the overall control framework to adapt, extend or even replace a device control entity dynamically. Considering for example the transmission of IP-TV traffic, it can be assumed that this traffic is prioritized in the network operator's access / aggregation domain. All CPEs are configured statically to provide the required level of quality of service to this IP-TV stream. An OpenFlow-enabled control plane allows control entities to cooperate and to share (at least partially) control over a specific data path element with adjacent or hierarchically higher control entities. Thus, in a Split Architecture a protocol like OpenFlow enables different data path elements to gain control over their environment to enhance the user's quality of experience.

2.2.1.1 Overview of access aggregation network

One of the most important parts of an operator's infrastructure is the access / aggregation network. Replacing outdated technology and updating this part of the infrastructure to meet future demands and expectations is a very expensive undertaking. Typically the access / aggregation part is organized in a hierarchical manner and consists out of the

following basic building blocks, assuming that services for residential and business customers, as well as mobile backhauling are organized via a common infrastructure:

- Customer devices connected to customer edge (RGW, router from business customers and base stations)
- Customer edges connected to access node (outdoor-DSLAM / GPON OLT)
- Access nodes (outdoor-DSLAM) connected to a first hierarchy of aggregation switches (AGS1)
- Access nodes (GPON OLT) and AGS1 connected a second hierarchy of aggregation switches (AGS2)
- AGS2 connected to a first access router (BRAS), LER and edge nodes of service platforms
- BRAS connected to edge nodes (LER)
- The physical transport is guaranteed by means of optical technologies; also utilizing additional multiplexing based on digital circuit switching and on wavelength, where appropriate.

A graphical overview of this infrastructure is depicted in Figure 4 below.

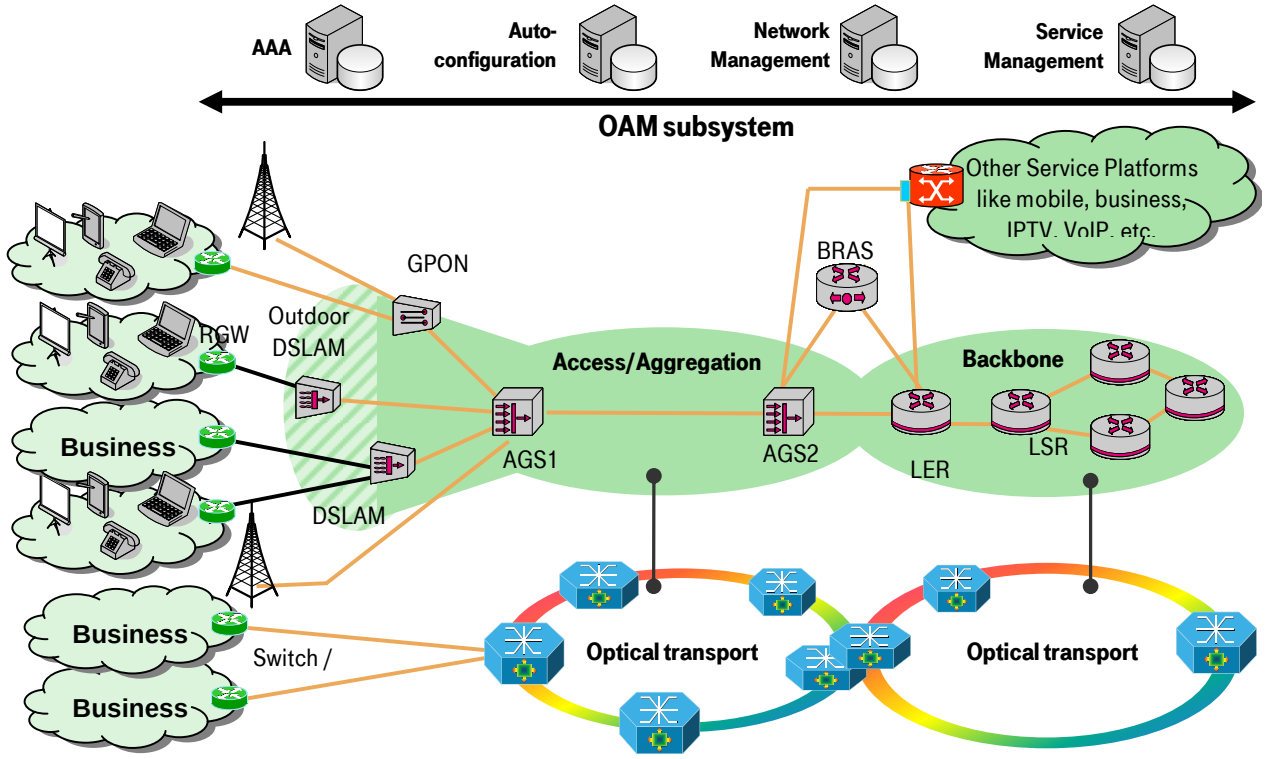

Figure 4: The access / aggregation network

The following assumptions are made to take the state of the art into account:

- Access / aggregation domains provide coverage of large areas, thus network planning usually aims towards an optimal network configuration with minimal capital expenditure, while at the same time avoiding a low utilization and fulfilling all SLA expectations from the operator's customers.
- Typically a tree or partly doubled tree-like physical topology is deployed with no or only a very little degree of meshed interconnection. This reduces the aggregation domain's ability to conduct complex rerouting operations.
- To enable resilience, we assume a redundant physical connectivity with a tree-like topology between the access nodes (DSLAM, OLT) and the root of the aggregation domain's tree (BRAS, LER and other service platform edge devices, see TR-101 [1] and TR-092 [11] for details). However, to ensure those resilience aspects, the backup resources must be appropriately reserved for outage scenarios.
- Such an Ethernet-based aggregation domain is mainly, but of course not only, focused on residential triple-play services and provides different service slices in this cluster; each slice is assigned to a dedicated QoS level so that IP-TV services can be prioritized over best-effort Internet traffic. The Broadband Forum has released several recommendations for defining QoS for IP services and the number of traffic classes to be supported, see TR-059 [12] or TR-181 issue 2 [8] for details. The assignment between service slice and QoS class is typically accomplished as a static association.

- An aggregation domain comprises a number of typical devices including access nodes, business / residential gateways, base station gateways (in the case of mobile backhauling), and service platform edge devices, where BRAS and access node provide sophisticated interworking functions for mapping different transport technologies to each other and terminating the aggregation domain. Today, a BRAS device provides a significant number of functions, most prominently those for authentication and authorization.
- An aggregation domain may comprise several stacked virtual or physical transport layers resulting in various available connectivity abstractions, e.g., IP, MPLS-LSPs, (Carrier) Ethernet, or lambda paths. A significant heterogeneity of technologies has to be assumed for the deployment of access / aggregation domains and may benefit from multi-layer operation (see Section 2.4 for details).

2.2.1.2 Seamless MPLS

In recent years, there has been a trend towards adoption of MPLS for IP core networks. Meanwhile, MPLS is a reliable technology for the transport layer as well as for the service layer of the networks. The success of MPLS in core networks is mainly based on two of its characteristics: the good synergy with IP and its versatility, like potential traffic engineering, QoS, resilience features on the transport layer side and enabling VPNs on the service layer side. In principle, MPLS can also be used in aggregation networks. In case of applying MPLS in both parts of the network, the basic protocols and the encapsulation could be identical, but the implementation with respect to the control plane, signaling, etc., might be different. The same is true for aspects like scalability and dynamic behavior.

The approach to extend MPLS from the core to the aggregation domain and also potentially covering part of the access is also known as "seamless MPLS". Details of the architecture of combined core, aggregation and access MPLS domains can be found in [2].

The most important motivation for operators to go for this Seamless MPLS approach is based on the simplification of operation by using the same technology on the separated service and transport layers. So the goal is to provide an architecture that supports different services and service groups on a single MPLS platform, integrating the different network domains, like access, aggregation and core. Service groups produced via such an infrastructure could be standard residential triple-play services, mobile backhauling or business services.

It is essential that all services use the same transport layer implementation. This will not change, even if new services will be established on a platform. Therefore, the critical aspects like ensuring resilience and maintaining sufficient redundancy, how to implement load sharing or how to implement domain interworking have to be solved only once.

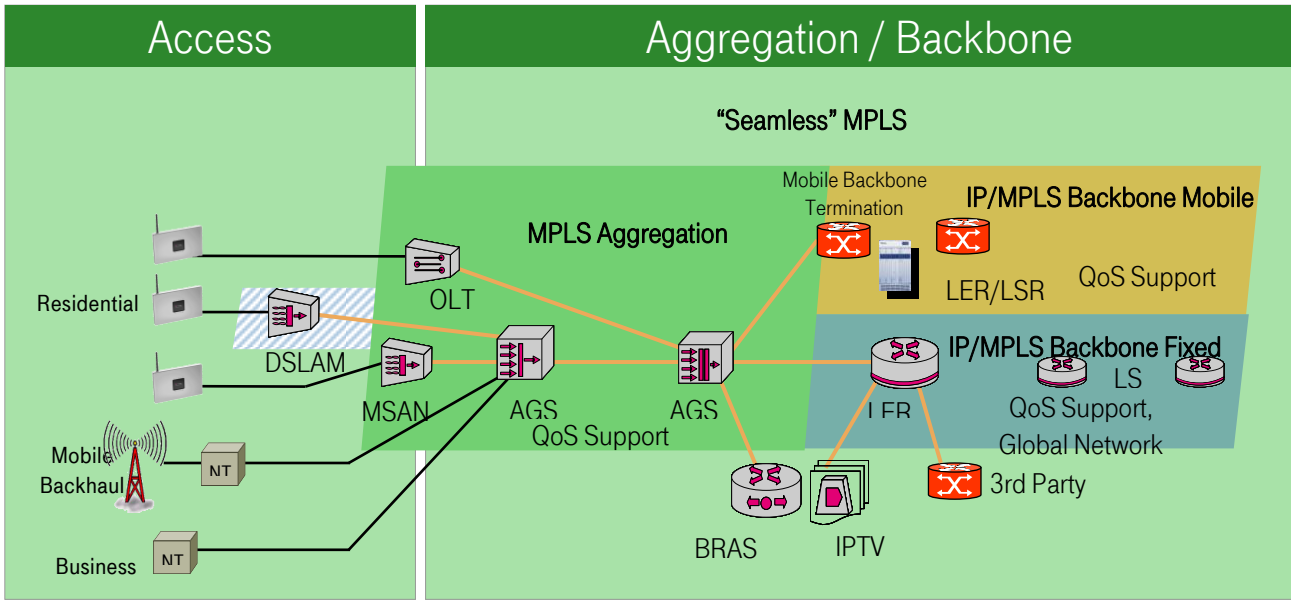

Figure 5: Seamless MPLS: Interworking of MPLS domains, while MPLS is extended towards the access

2.2.1.3 Multi-service / multi-provider operation

The work on multi-operator scenarios is motivated by the need to share physical or virtual infrastructure in aggregation domains due to financial or regulatory constraints or new business opportunities.

Until beginning of the twenty-first century, dedicated network infrastructures have only been used for a single or a very dedicated, restricted set of services. The increasing trend towards manifold and diverse services (e.g., voice services, web services, mobile backhaul, etc.) has led to a shift towards unified, multi-purpose network infrastructure. Especially, incumbent operators with very diverse networks and network technologies already in place and operated in parallel, aim towards regaining "economies of scale" by migrating the different service platforms into one layered network infrastructure. The standardization efforts in the area of packet-based access / aggregation networks (partly based on results of the IST FP6 projects NOBEL and MUSE) have evolved as mentioned in section 2.2.1.

It has to be mentioned that the concept of multi-operator scenarios via a common, shared infrastructure will grow in access / aggregation networks due to increasing demand of backhaul capacity for wireless / mobile and fixed coax, copper twisted-pair and fiber access networks. This convergence is already taking place and leads to the paradox that all network types are competing with each other, while depending on the same infrastructure. The need for cooperation in convergence can be already seen in different aspects today and will become even more important in the future. LTE is the next technology step in the upgrade of mobile networks. Forced by declining revenues and increasing competition, operators are already collaborating on the deployment and operation of infrastructure like physical sites, base stations or even entire networks [18]. Cable networks are evolving with DOCSIS towards so-called hybrid fiber coax networks demanding fiber access at locations similar to cabinets or primary distribution frames in POTS or xDSL networks. The uncertainties in deployment strategies for fiber to the home networks lead to very diverse approaches in Europe, like duplicated fibers in the last loop, access at primary distribution frames or the so called virtual unbundled access on Ethernet link layer. The latter is already approved by regulation authorities in the UK and Austria and might be a potential solution to provide appropriate access conditions for any provider and service. Different flavors of the corresponding business model are presented in Figure 6 [17]. In general, two splits are discussed: the three different roles of physical infrastructure (PIP), network (NP) and connectivity provisioning (CP) and service operation (SP). As many potential technical solutions compete and different levels of openness are under discussion, a future converged solution cannot be anticipated today. For example, model (a) is already approved in the UK and Austria by the regulation authorities and might be the potential solution to provide appropriate access conditions for any provider and service. The models (e), (f), and (g) are typical business models for incumbents (openness is enforced by regulation). In Sweden, models (b), (c) and (d) can be found as well.

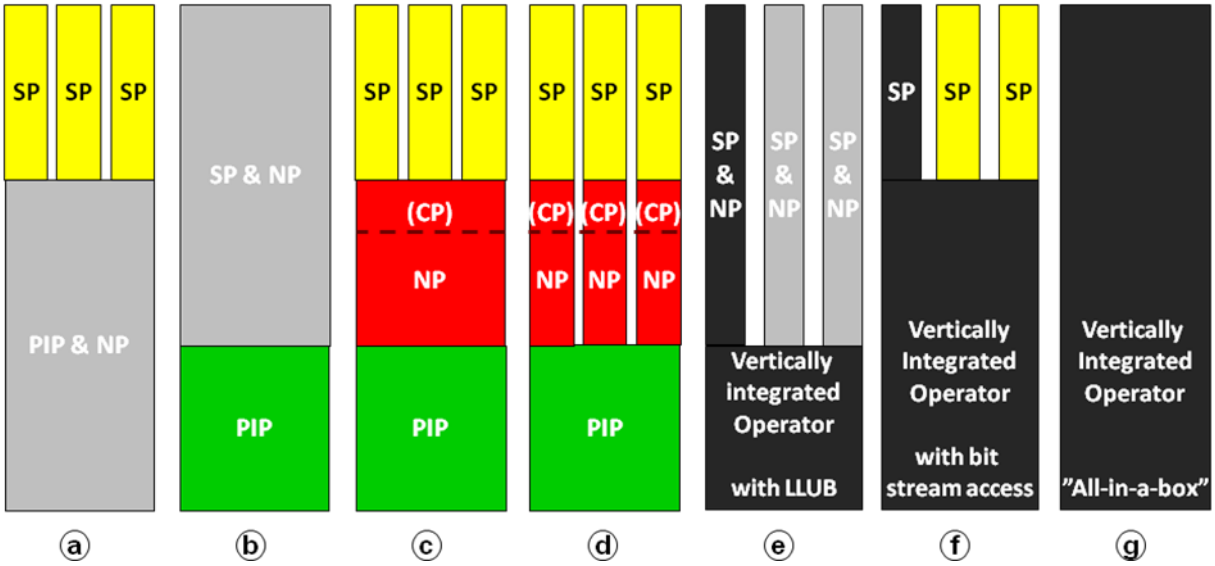

Figure 6: Business models in access / aggregation networks

While having multi-service and multi-provider operation, the complexity in network management must remain on the lowest possible level. This includes basic network requirements of forwarding, control and management with sets of functions like, e.g., QoS, AAAC, auto configuration and security. In addition, each provider wants to have some freedom of choice in level of operation of the network in order to differentiate products. Three basic cases can be identified:

1. Fully standardized and coordinated solution with detailed specified network functions and identifiers similar to former PSTN networks. This solution might violate the freedom of choice for providers and results in very similar end products with respect to the different operators.

2. The same forwarding principles with more advanced IT-supported control and management, but major functions are still network device-based. This would be a kind of virtualization, but not to the extent described in the deliverable D3.1. This solution provides each operator some choice about usage scenarios for a given set of network functions like resource allocation and provides a limited potential for product differentiation. Some coordination is required in the usage of identifiers and how management of

resources is done. For example, the identifier address space is split between all providers (similar to IP addresses) or an additional "meta" identifier (a tag) is introduced for differentiation (e.g., MPLS or PBB-TE). Resource allocation needs to be coordinated for appropriate classification and identification of classes. This could be supported by IT systems providing mechanisms for resource management like analysis of available resources during provisioning.

3. The fully IT-based network design would make use of the current developments in the IT-sphere especially in virtualization as discussed in the deliverable D3.1. Network devices would be virtualized so that providers would have complete freedom of choice in the deployment of solutions. This could end up in concurrent distinct forwarding, control and management planes and principles. It is questionable how much of this idea could be enforced by today's hardware.

Today, most networks are (at least) based on Ethernet IEEE 802.1q and 802.3ad reflecting case 2. It can be anticipated to some extend that aggregation networks will face an evolutionary step from Ethernet-based architecture to an MPLS-based system in the midterm, supporting different kinds of services in the aggregation domain. In addition, this allows operators to use a seamless MPLS architecture in aggregation and core domains and simplifies network management.

As OpenFlow supports at least IEEE 802.1q (VLAN) in its current specification (v1.0), tag-based differentiation of providers may be used in OpenFlow-enabled aggregation domains for multi-provider scenarios. Furthermore, it may smooth migration from Ethernet towards MPLS by enabling mixed operation for a specific transition period. Mixed operation resembles a multi-operator scenario and faces similar challenges (case 3): the operation of different competing control planes requires a certain level of coordination (though being under control of a single operator) and their admission control strategies must be mutually aligned, either in a cooperative or enforcing manner.

An important topic for offering triple-play services is resource allocation. Ethernet-based aggregation domains introduce service classes for managing voice, IP-TV, and best-effort Internet traffic: Different service classes are typically prioritized to fulfill the customer's quality of experience expectations. However, with upcoming new cloud computing services more dynamics can be expected: cloud computing services replacing or extending traditional desktop applications may require low-latency and low-loss transport connectivity. In the light of the ongoing net neutrality discussion, a provider-only defined prioritization of individual applications or service providers seems not to be desirable.

The term "slicing" is frequently used to depict sharing of resources and infrastructure, but this term requires a precise definition in the context of access / aggregation and what level of isolation is required and finally, what can be done at all in today's deployments. The state of the art can be characterized by the following statements:

- Taking into account net neutrality, a network provider-only defined prioritization scheme no longer seems sufficient. Furthermore, customers or providers should be granted the ability to assign service classes to specific applications to tune the customer's perceived QoE while meeting the provider's utilization goals. As a consequence, a more dynamic mechanism for defining connectivity requirements towards the network must be expected, potentially increasing complexity on customer, network and service provider sites. This might include advanced, complex and costly network functions like deep packet inspection. Here OpenFlow could make a difference as some simple identification of traffic patterns is possible without the need to establish a dedicated signaling path between customer and operator via the UNI. Another aspect is the introduction of sophisticated content delivery networks: smart caching strategies or rerouting of service requests for reducing overall load may be useful in future core networks.
- It must be possible for an operator to further operate prioritized service slices for dedicated services like IP-TV or Voice over IP for triple-play services.
- The complexity in aggregation domain nodes should be reduced, especially for interworking functions, authentication and authorization, service slice assignment, and QoS mapping. OpenFlow's basic principle of splitting the data and control planes offers the ability to conduct certain functions (e.g., AAA functions) externally on a controller device, which may reduce complexity and thus costs for data plane nodes.
- Prioritizing specific application flows due to customer request may require sophisticated traffic shaping functions in order to control data rates of low-priority flows. Here, OpenFlow may be an interesting candidate for controlling data rates e.g., by using ECN mechanisms to throttle TCP data rates. Such functions deployed on the edge nodes of an aggregation domain or even internally may balance overall load on aggregating nodes.
- The deployment and use of multiple control planes (either for a single or multiple operators) must be possible. Admission control conducted by different control planes must not impair other control planes running in parallel. A cooperative or dedicated model for coordinating resource allocation and admission control should be possible.

2.2.1.4 Mobile backhaul

The ITU-T has defined generalized mobility as part of the NGN definition of ITU-T in [35]. The term generalized mobility describes "the ability for the user or other mobile entities to communicate and access services irrespective of changes of the location or technical environment". In addition, it could be with or without service continuity.

As already mentioned, a number of different technologies and specifications for mobility support exist. In this chapter, the analysis and description will be limited to the 3GPP networks as they could provide general coverage and sophisticated mobility support capable service continuity. It should be noted that different BBF activities (see [13], [14], [15] and [16]) currently define requirements and potential solutions for mobile backhaul and should be taken into account while designing the architecture.

Focusing on 4G, defined by 3GPP SAE/LTE, mobile backhaul is the transport service which connects base stations (evolved NodeB or eNodeB) with the mobility controller (MME), the Service Gateway (SGW), the Packet Data Network Gateway (PGW) and the interconnection between neighbouring base stations. A backhaul network typically consists out of three aggregation levels.

- An eNodeB has three connectivity interfaces
  - between neighbouring eNodeBs, the X2 interface
  - to Mobility Management Entity (MME), the S1-C interface
  - to Service Gateway (SGW), the S1-U interface
- Non-3GPP access points (e.g., WiFi) connects to the PGW, via the S2 interface

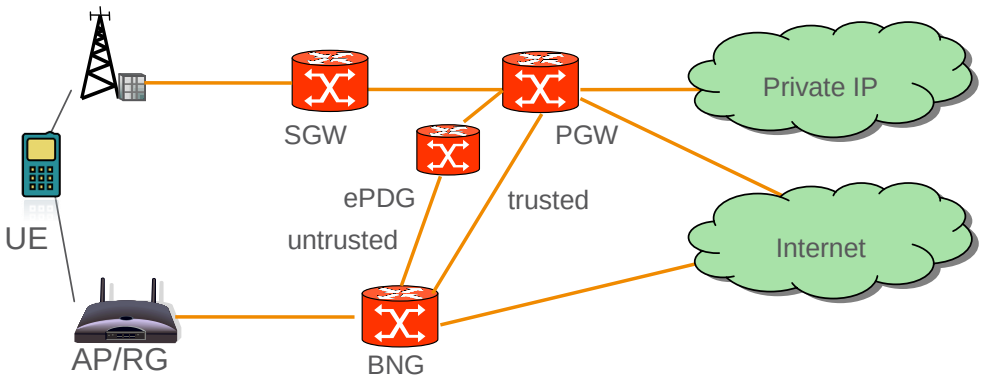

Figure 5: User plane of fixed-mobile converged architecture in line with BBF WT-203 and 3GPP TR 23.839. FMC also includes options for integrated authentication and policy control not shown here.

In the context of mobile backhauling, OpenFlow defines a number of potential use cases:

1. High capacity packet transport service between mobile base station and SGW (S1 interface).
2. Shared network where typically more than one provider utilizes the same mobile base station and same backhaul, but still uses separate mobile core networks (MME/SGW).
3. Distributed mobile service to enable local cashing and selective traffic off-load, e.g., supported by the 3GPP SIPTO (Selective IP Traffic Offload) approach [20].
4. Inter-base station connectivity (X2), supporting simple configuration of connectivity between neighbouring base stations.
5. Fixed-Mobile Convergence (FMC) to support, the increasing capacity demand by utilizing fixed-line access / aggregation between other access points (e.g., WiFi) to PGW (S2 interface).
6. 3GPP ANDSF (Access Network Discovery Selection Function) [21], where the operator can steer mobility between access types, e.g., based on SSID.
7. FMC supporting common network functions (like QoS, policy control, AAA, etc.).

The scenarios in which Open Flow potentially improves current deployment strategies (1, 2, 3, 5 and 6) are covered by multi-provider / multi-service use cases. New technologies like LTE Advanced with increasing demand on bandwidth and reduced delay create new requirements on mobile backhaul and functional support.

According to the NGMN alliance performance requirement R-48 [19], the NGMN backhaul solution must guarantee an end-to-end two-way delay of 10ms. Therefore, any improvement of the transport architecture is valuable. The X2 interconnection, in particular could be improved with the support of this interface by OpenFlow in the transport path (scenario 4). An OpenFlow aware switch could identify traffic on the X2 interfaces and discover the best connection between eNodeBs and configure corresponding flows in the aggregation network accordingly. It could be assumed that this interconnection is quasi-static, defined by the installation of aggregation network and eNodeBs or after a restart of either of the involved network elements.

Another issue in mobile networks is the increase in traffic. Offloading of traffic to local networks, e.g., handover to WLAN (so called I-WLAN) will be relevant. Again, this use case requires OpenFlow aware switches in the access / aggregation domain. WLAN is covered in Non-3GPP access networks and could be a potential solution to relax this issue. There are two different principles to realize this offloading: trusted and untrusted. Both principles need connectivity towards the PGW, in the case of untrusted relationship, connectivity via the evolved Packet Data Gateway (ePDG) would be necessary. OpenFlow aware switches in the access/aggregation domain may detect corresponding flows and discover connectivity between a WLAN and the PGW or ePDG.

Furthermore potential Home-eNodeBs would be connected in a similar way as "public" eNodeBs, but using the broadband access of the corresponding customer as backhaul to a SGW that connects further to a PGW. By complementing the Home-eNB with local gateway (L-GW) functions, mobile traffic could be offloaded locally to the Internet or a local network [20]. Again OpenFlow could be beneficial to establishing the path for offloading.

2.2.1.5 Software- defined networking (SDN) application in context of 802.11 compliant devices

We discuss the benefits of Software-defined Networking (SDN) in the context of 802.11 compliant devices. The IEEE 802.11 standard has seen various extensions in recent years: for increasing rates on the wireless channel, adding advanced management functionality, and the like. While devices with time-critical control plane functions in IEEE 802.11 [3] are usually implemented directly in a PHY/MAC chipset, most control plane functions with less time-critical functionality may be separated from the 802.11 data path element and moved to a separate controller device. Compared to the monolithic approach as used today, Split Architecture allows deployment of new management functionality at run-time. However, beyond this static patching of AP functionality, a Split Architecture also enables deployment of advanced functionality and network services like enhanced mobility schemes.

Static deployment of 802.11 extensions

As an example of the benefits of static functionality extensions in a Split Architecture, consider the enhanced authentication and authorization framework defined in IEEE 802.11i (check current aggregated IEEE 802.11 specification for details [3]) that defines several complex function blocks for key management, key generation and the encryption process for protecting data frames. Today, these functions are typically collocated within an 802.11-enabled device. With OpenFlow's split of control, forwarding and processing, 802.11i's key management related functionality (4-way-handshake, derivation of key hierarchy, etc.) may be moved to the controller, while time-critical encryption should be done on the data path element. Multiple encryption schemes have been defined during IEEE 802.11i's lifetime, e.g., TKIP for re-using existing WEP-based chipsets and an AES-based one, acting as default encryption scheme on legacy-free hardware.

A pluggable 802.11 implementation must be enabled to use various encryption schemes that should be loadable at run-time. According to the 802.11 standard, different STAs should be able to use different encryption schemes.

The decryption / encryption process may be physically separated from the radio channel point of termination and traffic may be conveyed in the OpenFlow cloud in an encrypted state. Decryption may occur on an edge device once the flow leaves the OpenFlow cloud.

Dynamic adaptation of 802.11 functions to flow requirements at run-time

Wireless LAN environments have replaced wired installations considerably in the past and inevitably, beyond conveying best-effort traffic, 802.11 are in use today for the transport of time-critical applications like Voice over IP or video telephony as well. Providing QoS support in IEEE 802.11 is challenging: 1) due to the use of unlicensed spectrum and 802.11's inability to monitor radio channel conditions during operation, and 2) due to limitations of the 802.11 framework architecture itself, e.g., handovers cause considerable delay due to the security subsystem's complexity. The IEEE 802.11 working group has addressed these deficiencies recently by publishing new amendments.

In the IEEE 802.11k amendment [4], a framework for conducting radio measurements has been defined whose reports may be used by various system management entities in an 802.11 station (STA) or access point (AP). It enables stations and access points to monitor and assess their radio environment for initiating handover processes and conducting radio resource management for utilizing the available radio resource more efficiently.

With the IEEE 802.11i amendment [3], the limitations of WEP have been solved by introducing new sophisticated authentication and encryption schemes. However, one of 802.11i's side effects is its impact on handover delay and

latency times, which makes an 802.11i-capable ESS ineligible for time-critical and delay-sensitive applications when frequent handovers occur. The IEEE 802.11 working group has addressed these problems in an amendment to the security subsystem incorporating support for Fast Basic Service Set Transitions (see IEEE 802.11r [5] for details).

A Split Architecture should allow the control plane to activate specific features like IEEE 802.11k/r based on the flows' characteristics traversing the WLAN. When a new flow requires enhanced quality of service, the control plane may activate 802.11r support on all APs adjacent to the customer.

In case of high load conditions in the wireless domain, the control plane may deploy and activate 802.11k at run-time and feeding a dedicated radio resource management entity with generated radio channel measurements. When activated on several APs in a specific neighborhood, the radio resource management may increase radio efficiency.

The control plane may advertise Fast Transition information elements in the WLAN APs' beacons and deploy Fast BSS Transition support at run-time when STAs enter a BSS that support a Fast Transition protocol as defined in 802.11r.

Enhanced mobility services

Though wireless LANs have mitigated the physical boundaries of their wired counterparts, WLAN coverage areas are typically small compared to public, legacy mobile networks. However, virtualization of WLANs is a well-known technique for emulating different networks on the same physical infrastructure. With a Split Architecture, an existing WLAN deployment on the edge of an OpenFlow cloud may be used to provide enhanced mobility services: 3GPP has defined new mobility protocols in releases 8, 9, 10 [6] including support for PMIPv6. Proxy MIP emulates a device's home environment independently of its current geographical location and enables mobility for devices that lack support for enhanced IP mobility services.

With a Split Architecture, such virtualization of wireless resources may be created dynamically per user. While the user moves, the virtualized WLAN will be created at run-time on all devices, starting adjacent to the user's current physical location. The control plane must be enabled to generate virtualized SSIDs and information elements for the transmitted beacons on an 802.11 AP.

2.2.1.6 Dynamic control composition

Service-delivery frameworks (as supported by TMForum TR139) along with good practices, such as ITIL v3, define conceptual structures that enable organizations to deliver next-generation services independently on the network technologies that carry the actual implementation. A Split Architecture solution for the network implementation would therefore need to provide the mechanisms for proper integration into a service-driven world. One important aspect of service-centric management is the need to automate everything that can be automated. With so many running instances of different types of services, each of them steered by different service level objectives, providers need to reduce the manual steps involved in designing, deploying and operating network services as much as possible. Service delivery frameworks provide guidelines on what steps are required, but the actual implementation and automation of these steps is specific to each particular service category.

Connectivity services are defined for Ethernet technology by the Metro Ethernet Forum in the MEF 6.1 and 10.2 specifications. These documents describe how service-level attributes are supported by network-level capabilities. The MEF 17 specification depicts a multi-operator architecture supporting operation, administration and management of connectivity services in a service-centric environment. MEF 17 defines specific methods that operators could employ to fulfill the requirements imposed by Service Operations in the ITIL and TMForum frameworks. The service-centric management capabilities that should be supported by nodes in an OF-aware environment are:

- Discovery: this function allows service-centric nodes to automatically learn about the other service-centric nodes supporting the same service instance
- Monitoring of connectivity status for a connectivity service within the domain of responsibility of the service provider
- Measuring of frame loss ratio performance, e.g., counting the number of frames lost within the service provider's domain of responsibility
- Measuring of frame delay performance as an expression of the delay experienced by different frames forwarded on the same service instance
- Measuring frame delay variation performance to determine the dissimilarity in the delays experienced by different frames forwarded on the same service instance
- Cross-layer capabilities for receiving fault notifications from the underlying transport layer

In practice, these requirements are implemented through Operations, Administration and Management (OAM) features specified in the IEEE 802.1ag and the ITU-T Y.1731 recommendation. As a transport technology, MPLS-TP comes with its own set of OAM features, based on the Bidirectional Forwarding Detection (BFD), LSP-ping and ITU-T

Y.1731. All OAM tools require active participation from the data plane for generating, forwarding and consuming OAM frames with minimal interactions with the control plane. Supporting such tools in an OpenFlow-aware network would mean that the data plane would need to be enhanced with capabilities that allow it to handle OAM frames.

Capabilities fulfilling the service design and deployment requirements of ITIL v3 and TMF TR139 were also described for Ethernet connectivity services. In particular, the use of a GMPLS control plane for automating the configuration of the network nodes along with Path Computation Elements that simplify the service design by automatically calculating the best path through the network while taking into account constraints (see IETF RFC 5828). Other advanced service-centric requirements, such as service composition and the ability to govern services based on high-level goals specified at the business level, although mentioned in TMF TR139, are still active research areas when it comes to actual capabilities to be implemented in the network nodes.

In an OpenFlow-aware environment, service composition requires the ability to allow cooperation of split arch control entities of different administrative domains. In the access / aggregation domain this includes the cooperation of control entities deployed within the customer's local environment on one hand and within the network operator's domain on the other. Let us assume that today's monolithic architectures will be replaced with a split arch approach in the future, not only within carrier-grade equipment, but also tackling typical customer premises equipment.

In order to mimic legacy behavior in legacy environments, a CPE may incorporate a control entity for ensuring the device's basic operations. However, in an OpenFlow-aware networking environment, such a device may change its behavior and become part of a distributed control plane, sharing control over its data path element with other control plane entities.

To enable control entities to cooperate in an OpenFlow-aware environment, registration, announcement, and discovery of these entities must be available. Furthermore, control entities should be enabled to share properties of the data path elements and to depict the level of control they are willing to share with others. Inevitably, means for authentication and authorization are required.

As a practical use case, consider the delivery of IP-TV streams to a customer. Today, the network operator deploys CPEs that are statically pre-configured to handle IP-TV with the necessary priorities (see BBFs TR-069 and TR-181 issue 2 for details), limiting the customer's abilities to self-deploy arbitrary networking equipment without breaking the end-2-end delivery path.

In an OpenFlow-aware environment, CPE control entities should interact with control entities in the operator network. Here, an operator control entity may provide a module for configuring a customer's access point and request control over the customer's access point data element while the IP-TV traffic is streamed towards the user environment.

Please note that such a cooperation of control entities of different administrative domains or within a single administrative domain is also applicable to provider-to-provider scenarios.

2.2.2 Functional description

The different use cases show a widespread demand of telecommunication services for all kinds of customer groups:

- Residential customers with single, double or triple-play service combinations based on
  - Internet access
  - VoIP
  - Video (as real-time and on demand)
- Business customers with leased line and LAN services (VPN)
- Mobile network providers with demand for backhaul services, to interconnect base stations with switching centers
- Wholesale customers, demanding a mix of the aforementioned transport services for their customer groups

So, from a high-level point of view, the following requirement could be defined:

R-1	**A wide variety of services / service bundles should be supported.**

In most use cases administrative domains have to be separated. This could be distinguished in two different directions. First, a horizontal split in between different domains of a network. For example, one provider could operate the access / aggregation network and another one the core network. Both must be interconnected in a way that the providers require only a very limited amount of configuration adaptations for changes in the interconnection interface. Second vertical split through one infrastructure. Today, operators, sharing infrastructure and network technology, have to adapt to the underlying network principles and configuration options provided by the operator of the active network infrastructure (e.g., OTN, Ethernet, IP, etc.). A certain degree of freedom should be provided in order to provide more flexible use of infrastructures. Based on this principle, high-level requirements could be defined:

R-2 The Split Architecture should support multiple providers.

R-3 The Split Architecture should allow sharing of a common infrastructure, to enable multi-service or multi-provider operation.

R-4 The Split Architecture should avoid interdependencies of administrative domains in a multi-provider scenario.

From these general requirements, technical requirements may be defined. Following the discussion in Section 2.1, this could end up in different solutions. Today's switches support a wide variety of technologies like different flavors of Ethernet, MPLS or IP. Especially the transition of IPv4 towards IPv6 is still a challenge. Most probably, parallel support of both is required in an access / aggregation network. Other examples are parallel support of IEEE 802.1d, 802.1q, 802.1ad and potentially 802.1ah or MPLS.

R-5 The Split Architecture should support operation of different protocols / protocol variations at the same OSI layer in the case of shared networks.

For operational and administration purposes, it is important to rely on policies. Policies are transferred into network devices and reduce manual configuration efforts. If possible, this should be done automatically. However, policy distribution is out of the scope of SPARC. Examples for this principle are policies regarding bandwidth control. Today, two different bandwidth groups primarily define services; the assured and technical guaranteed bandwidth and the rest of the available bandwidth up to the maximal capacity of the connection. The maximal available bandwidth is important in the design of the first mile in order to guarantee best possible customer experience. Moreover, the operator's network should not be overloaded and the demand coming from different sources must be controlled, in line with the guaranteed bandwidth. Based on product definitions, policies could be defined and transferred into the network devices. Here the policies must be enforced and in case of violation, the policy describes how to handle the traffic. Possible options are dropping of corresponding packets or the exchange of the bandwidth class identifier with a lower class. Additional functions benefitting from policies are auto-configuration, authorization or accounting.

R-6 The Split Architecture should support policy-based network control of network characteristics.

R-7 The Split Architecture shall enforce requested policies.

R-8 The Split Architecture should support automatic transfer methods for distribution of customer profiles and policies in network devices.

Synchronization is required for different services like TDM emulation or mobile backhaul. Different mechanisms are available, the basic classes are:

- Physical-based methods (e.g., Synchronous Ethernet)
- Long-term stable oscillator (stable for months)
- Protocol-based methods (e.g., NTP, IEEE1588v2) with/without intermediate nodes support (e.g., transparent clock implementation in intermediate backhaul nodes for IEEE 1588v2)
- GNSS (e.g., GPS, Galileo, GLONASS, BeidouSession limitation for PPPoE (Internet access), VoIP or Video (session equal to channel))

R-9 The Split Architecture should support TDM emulation and/or mobile backhaul.

Each network element should be able to correlate packets / flows with a customer. Therefore, different information must be included in the header. For example, a Layer-2 switch is limited to use Ethernet headers (e.g., MAC address, VLAN IDs), while a Layer-3 switch could make use of IP addresses. To add flexibility and overcome scalability limitations, different functions in a device like stripping, adding, dropping/popping of headers and header fields might be required. The available namespace in IEEE 802.1q is limited to 4096 addresses and thereby defining an additional constraint.

R-10 The Split Architecture should provide sufficient customer identification.

The manifold services have different requirements regarding performance parameters of the network in order to provide the guarantied customer experience. Here, the major aspects with respect to guaranteed customer experience are just touched upon briefly. For more details, see the related Metro Ethernet Forum documentation. It is important to mention that any control messages like multicast group events or OAM flows have to be sent in the highest priority class in order to retain a certain level of control, even in case of congestion. In addition, any traffic which does not carry a QoS class identifier (e.g., VLAN header provides eight possible markings, IP ToS / DSCP 64) should be handled in a default class, which is typically best-effort. Traffic which carries an invalid QoS class identifier (not in line with local definition) should be handled as best effort unless mapped to the local QoS class identifier space. This should prevent unpredictable network behavior. As mentioned in R-6 and R-7, policing is a prerequisite and must be enforced.

R-11 The Split Architecture should support best practices for QoS with four differentiated classes according to the definition documented by the Metro Ethernet Forum.

R-12 The Split Architecture should handle data not carrying QoS class identifier as default class.

R-13 The Split Architecture should map data carrying invalid QoS class identifier to a valid QoS class.

R-14 The Split Architecture must handle control data as highest priority class.

Real-time television services are based on the broadcast principle and require much more bandwidth compared to web browsing. Broadcast in a packet network has certain limitations and it is not possible to transfer the principle of the traditional television broadcast networks directly to the packet network. Multicast could help to minimize network resources occupied by IPTV-services. Multicast was originally designed to enable N:N communication (multicast group) between a number of peers (N). Each of the peers should have the possibility to send data (e.g., voice or video) to all other peers in the multicast group, but this behavior is not desirable for television services. Therefore Source-Specific Multicast should be used for providing IPTV-services, e.g., PIM-SSM or IGMPv3 / MLDv2.

R-15 The Split Architecture should support Source-Specific Multicast.

The access to the network and to specific services by customers must be controlled. This is done with authorization mechanisms. Typical mechanisms for authentication of network access work by including some Line ID information like DHCP option 82 or PPPoE intermediate agent. In addition, the architecture should be open to support other mechanisms like PANA (Protocol for carrying Authentication for Network Access). Access control to specific services like IPTV or VoIP might have to be supported by some network functions as well. Typically, this is implemented with the same mechanisms used to control network access in single provider scenarios, but in a multiple provider scenario with some additional service-integrated mechanisms.

R-16 The Split Architecture must control the access to the network and specific services on an individual service provider basis.

Different security mechanisms are required in the network. Flavors of switched Ethernet are state-of-the-art Layer 2 technologies in carrier networks today. One of the security problems is that Ethernet switches forward specific packets like authentication or ARP requests to all ports. This means that confidential information from one customer is potentially sent to another one. To prevent this unwanted behavior, mechanisms enforcing traffic directions should be used. In addition, the characteristic of a switch must be changed in a way that spoofing is restricted. E.g., identifiers, which have been distributed under control of operators, have to be checked to prevent uncontrolled usage (like using an IP address other the one in the DHCP message). A related issue is ARP/RARP message inspection, so that information could not be requested arbitrarily from customers. A firewall type of mechanism should be implemented for specific Ethertypes in order to prevent the unwanted access to certain functions in switches.

An important scalability requirement for devices is the availability of high-speed memory for different lookup tables. In large-scale network environments, operators want to control the number of entries in these lookup tables, especially the number of MAC addresses. A simple mechanism is to limit the number of identifiers at the network border. Another potential mechanism is the MAC address translation at the access nodes. Besides the limited amount of memory, the processing capability of a switch is a scarce resource. Control messages are an important source of CPU time consumption, therefore their send rate (requests per second) should be limited.

R-17 The Split Architecture should provide mechanisms to control broadcast domains.

R-18 The Split Architecture should support enforcement of traffic directions.

R-19 The Split Architecture should support control mechanisms for identifiers. This should include any kind of identifiers like addresses or protocols as well as limitation for send rate.

R-20 The Split Architecture should prevent any kind of spoofing.

Network management requires information from the network in order to define appropriate actions. In general, the network management should be based on common, well known mechanisms. Manifold parameters should monitored, e.g., bandwidth usage. In addition to the monitoring, traps should be generated in case a certain rule is violated, e.g., too many requests for a control protocol were sent, a limiter is at 100% level, etc. In addition, accounting information like start/end of a session should be monitored.

R-21 The Split Architecture should monitor information required for management purposes.

R-22 The Split Architecture should generate traps when rules are violated.

R-23 The Split Architecture should extract accounting information.

R-24 The Split Architecture should collect traffic statistics.

For operation and maintenance, a number of functions are already widely defined and implemented. This includes identification of link failures, connectivity checks and loopbacks in the data plane. In addition it should be possible to send test signals and measure the overall performance. Standards to be supported are ITU-T Y.1731, IEEE 802.3ag/ah and respective IETF standards for IP and MPLS. Interfaces of switches, routers, etc., provide additional information for monitoring and operation like link down, etc. This includes monitoring of links between interfaces as well.

R-25 The Split Architecture should support OAM mechanisms according to the applied data plane technologies.

R-26 The Split Architecture should make use of OAM functions provided by the interface.

R-27 The Split Architecture shall support the monitoring of links between interfaces.

For Software-defined Networking, a number of general principles should be applied to a Split Architecture. By introducing a fundamental split of control, processing and forwarding, the data path element is now split logically into two functional element groups that may occur in a data path: 1) the forwarding element that provides the ability to switch packets between different in/out-ports based on its Forwarding Information Base, and 2) the data processing element that provides all functionality for changing a packet's content including the ability to rewrite the packet's header**.** In this document it is assumed that termination and adaptation functions are part of data processing.

These two functional elements may be collocated on a single physical device, but the logical separation between these two should be reflected by the Split Architecture implementation. Therefore, a Split Architecture should aim towards a general, protocol agnostic control of forwarding and processing over logical disjunctive interfaces, although a multiplexed transport over a single connection between controller and data path element may be used.

R-28 The data path element should provide logically separated access to its internal forwarding and processing logic in order to control both independently.

A data path element may consist of forwarding functionality only (i.e., a trivial forwarding element may exist without any processing functionality) or may provide some additional processing capabilities. This allows data path elements with various degrees of performance and capacity. The forwarding function must be enabled to filter a flow of packets through a chain of processing functions, e.g., several interacting processing functions may be mapped on different layers of the networking stack. There are two different possibilities (and a mix of both) for implementation of the decision logic. First it could be provided by the forwarding function or, second, a processing function decides based on the content of the processed packet. Currently, it seems to be favorable to use the forwarding function in conjunction with the Forwarding Information Base as decision logic.

R-29 It should be possible to define chains of processing functions to implement complex processing.

A Split Architecture for Software-defined Networking should allow the introduction of arbitrary processing functions defined by the operator. These processing functions should be loadable on the data path element as these should be tightly coupled to the data path and be crucial for an optimal flow treatment performance. A processing function should be loadable at run-time without resetting the data path element. A processing function must provide the ability to reconfigure its internal state at run-time and to exchange flow-related information with the corresponding controller entity in the control plane.

R-30 The Split Architecture shall support deployment of legacy and future protocol/service-aware processing functions.

R-31 The introduction of a new protocol/service aware processing function should not necessitate the update of other functions.

R-32 The architecture of a data path element shall support loading of processing functions at run-time without service interruption.

R-33 A processing function instance should be controllable at run-time by the associated control entity in the control plane.

Ports on a data path element vary in their characteristics, e.g., optical and wireless interfaces, thus having interface specific configuration parameters on a data path element. Moreover, these ports may implement protocol processing functions (e.g., GFP), too. The control plane may be enabled to manage these port related attributes. This logical protocol may also be able to manage the forwarding function and the processing function, such as disabling/enabling a processing function or managing configuration attributes assigned to those functions. This logical protocol can operate via existing device management interfaces (SNMP, proprietary vendor protocols, netconf) or via a third general control interface complementing the forwarding and processing control interfaces as defined.

R-34	**A processing function should expose module-specific configuration parameters to an associated entity in the control plane.**
R-35	**The Split Architecture should allow the exchange of opaque control information between a processing function on the data path element and the associated control entity with a well-defined protocol.**
R-36	**The level of detail exposed by a processing module is module and vendor specific. However, each processing module should support a common API for control purposes.**
R-37	**Urgent notifications sent by a data path element should be prioritized and not be delayed by data traffic to the controller.**
R-38	**A data path element classifier should be constructed in a protocol agnostic manner or should be at least flexible enough to load new classifier functionality as a firmware upgrade with identical performance.**
R-39	**The Split Architecture should introduce a clean split between processing and forwarding functionality.**
R-40	**The Split Architecture should provide means to control processing functions from a controlling entity on heterogeneous hardware platforms.**

2.2.3 Requirements on dimensioning and scalability

Section 2.2.1.1 has already detailed the structure of the access/aggregation network, a graphical overview could be found in Figure 4. In the following, an overview of dimensions of the customer edge side and some high-level estimation for the access/aggregation network itself is provided. It should serve as the first input for an extended scalability analysis.

Currently, network design is still based on deployment of copper-based access networks. With the further extension of fiber in the local loop, FTTH networks will be deployed. It can be assumed that most of the large-scale deployment will be based on GPON, at least in the next couple of years. FTTCab deployments are another step between a copper-based access and a solution based on optical transport. Typical assumptions to dimensioning these different flavors of access network are:

- FTTLEx with DSLAM as access node and up to 1,400 customers per DSLAM
- FTTCab with DSLAM as access node and up to 200 customers per DSLAM
- FTTH with GPON OLT as access node, today
  - GPON OLT has 15 slots
  - An OLT line card has eight ports
  - Some slots are required for interconnection towards the network;
    1/3 for uplink (Ethernet), 2/3 for downlink (GPON) (theoretical overbooking of 1:2)
  - Per OLT port up to 128 (typically 32) customers could be connected

In order to reduce the complexity of the calculation, the number of devices of the two hierarchy levels of aggregation switches (AGS1 and AGS2) and the IP service edges (e.g., BRAS, LER) is estimated based on the knowledge of the involved technologies, system designs, traffic behavior assumptions and the utilization of statistical multiplexing. In principle, the numbers should be:

- Number of customers devices >> customer edge >> access nodes >> edge nodes
- Sum(Access nodes) : Sum(aggregation + edge node) = 10 : 1

Another trend taking place in fixed networks is the consolidation and potential reduction of the number of sites. Typically, a PoP (point of presence) location reflects the connection between access / aggregation and backbone network and is some kind of "fixed" reference value. Typical rule of thumb for one of the access / aggregation area is:

- Today: 1 PoP location for 500,000 households (about 1 million inhabitants)
- Future: 1 PoP location for 2,000,000 customers
- Long-term: 1 PoP location for 4,000,000 customers

The values presented in the following are based on an average 500,000-household scenario and could be simply scaled up for the other two forward-looking scenarios. In principle, a take-up rate (with time horizon) and a sharing between different technologies/providers should be taken into account. But the calculated numbers represent the maximum case

and any sharing of infrastructure between different providers is neglected. In reality, customers are shared between different providers, but remain on one infrastructure. Beyond the assumptions included in this model, for residential customers it could be assumed, that four large nationwide and six regional providers are operating in parallel.

The multi-service scenario assumes four different customer groups, where the wholesale group is only a mixture of the other three. For each group, the following parameters are estimated:

- Absolute number of customers
- Number of MAC addresses visible in the network (average and maximum case)
- Number of VLAN IDs visible in the network
- Number of IPv4 addresses
    - Number of PPPoE sessions for typical Internet access
    - Number of DHCP-based address for IP services like IPTV
- Number of IPv6 addresses (normal and advanced scenario) and prefixes

This following list describes the assumptions for a scenario with specific household coverage:

- Assumptions related to residential customers
    - Mixture of single and multi-play; 40% IPTV customers
    - Three (standard household) up to ten devices in customer domain (+ RGW)
        - Normally, each device connects to RGW, which terminates L2 and translates IPv4 addresses
        - In the worst case, each device directly connects to access node (based on state of the art product definition up to ten)
    - If devices are connected to RGW, which is responsible for the IP prefixes, then
        - Today one IPv4 prefix (subnet mask /32) for PPPoE traffic and one IPv4 prefix for IPTV (based on DHCP)
        - Upcoming, one IPv6 prefix (/64) per RGW and one IPv6 prefix (/56) per customer network and additional two for IPTV
- Assumptions related to business customers
    - Number of business customers will be in the order of 10% of residential customers
    - Most important groups of the customer edge
        - 80% are basic services with xDSL/GPON-based connectivity
        - 20% are advanced connectivity services
            - 75% are PtP connectivity locations with connectivity to another one
            - 25% are MPtMP connectivity locations with connectivity to different other ones
    - Number of stations is about
        - 10 for “basic” services per customer edge
        - 100 for advanced services per customer edge
    - Each station has one MAC address, VLAN Id and one IP address
    - Each customer network has one IP prefix; customer edge device one IPv6 (/64) prefix
- Assumption related to mobile backhaul
    - Each mobile station covers 1000 households
    - Up to four mobile network providers
    - Each base station has one MAC and one IP address/prefix

Taking all this different assumption into account the calculation is done per identifier with

- # of business customer = 10% x # of residential customer
- # of business customer “basic” service = 80% x # of business customer

- # of business customer “advanced” service = 20% x # of business customer
- # of base stations = Nr. of residential customer / 1000
- Sum of identifiers with
  - # of residential customer x # of identifiers per residential customer
  - # of business customer “basic” service x # of identifiers per business customer “basic” service
  - # of business customer “advanced” service x # of identifiers per business customer “advanced” service
  - # of base stations x # of identifiers per base station

The numbers for each of the identifiers in the 500,000 household scenarios are:

- MAC addresses
  - normal case: 1.902.000
  - maximum case: 6.402.000
- VLAN ID: 2.102.000
- IPv4
  - addresses for PPPoE sessions: 500.000
  - addresses for IP services: 1.602.000
- IPv6
  - Addresses normal scenario: 3.652.000
  - Addresses advanced scenario: 7.152.000
  - Prefixes: 1.502.000

The number of devices (assuming today’s technology and system designs) in the access/aggregation network is summarized in the following table.

Table 1: Devices in access/aggregation scenarios

Topology	FTTLEx	FTTCab	FTTH
Access nodes: normal	386	2.700	211
Access nodes: max			53
Aggregation + Edge	39	270	21
Total (Normal + Aggregation + Edge)	425	2.970	232

2.3 Use Case 2: Data center

2.3.1 General description

Large-scale data centers have lately become an integral part of the Internet e.g., as the backend for large websites, performing distributed calculations such as indexing the entire web, or as the backbone of cloud services (e.g., Amazon EC2). Large-scale data centers can consist of tens of thousands of servers which in turn may contain dozens of virtual machines; such large server farms require a complicated infrastructure of cooling, power distribution, networking, and management systems. This has led to developments of several proprietary approaches and equipment for data centers that increase both CAPEX and OPEX. In addition, proprietary approaches reduce the flexibility of data center architectures in terms of the ability for adopting new protocols and optimization tools. As a result, efforts have recently been put into simplifying the data center architectures and moving in the direction that most functionalities of a data center can be realized through employing inexpensive commodity hardware (scale-out architectures instead of scale-up). In the area of data center networking and management, the concept of OpenFlow, as elaborated earlier in this document, offers high potential for facilitating this trend. Specifically, OpenFlow decouples the intelligence from the hardware of the networking equipments and moves it to a central controller. As such, OpenFlow provides the possibility to develop solutions based on commodity switches instead of the typically high-end expensive commercial equipment.

The issues we focus on are increased network and server utilization through load balancing on different layers in the network stack as well as on the server and virtual machine load level, the large amount of complex networking hardware needed, complexity in managing multiple data center customers and energy efficiency.

In Figure 7, a typical traditional data center network architecture is illustrated. At the bottom of the figure are server racks, each with roughly 20 servers that are connected to the rest of the network using Gigabit Ethernet to a Top of Rack (ToR) switch. The ToR switch connects to a multi-tiered switched network which in turn connects to a routed aggregation network which finally connects to the external world. The links going up in the hierarchy consist of a mixture of GbE and 10GbE, with an emphasis on higher bandwidths closer to the core. All connections to nodes higher in the hierarchy are doubled to provide 1 + 1 redundancy in case of equipment or link failure.

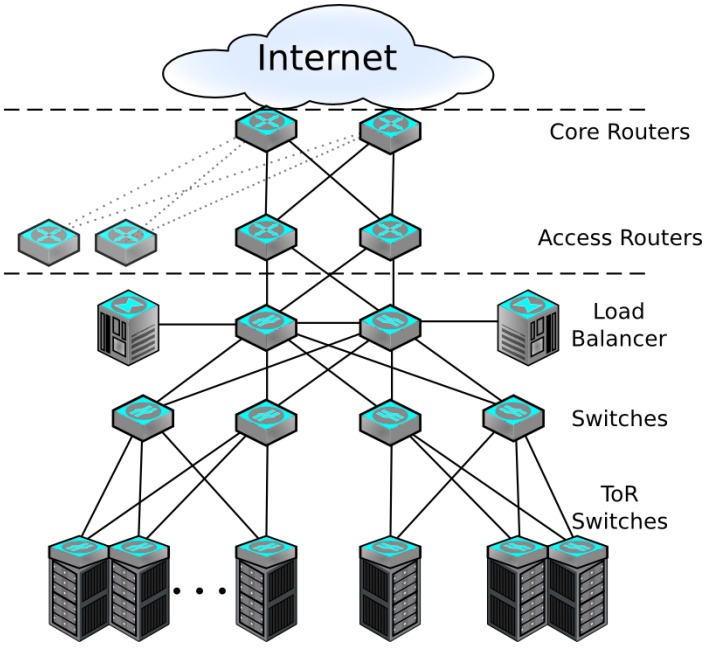

Figure 7: Data center network design ([25])

There are several problems with this architecture, e.g., scalability, inflexibility and asymmetric bandwidth availability. Since traffic is aggregated towards the core, the requirements on equipment further up in the hierarchy are increased which leads to high costs for the central nodes and results in problems concerning scaling of the network once the bandwidth limits of these routers and switches are reached. Hierarchical packet networks typically lead to high levels of oversubscription and more congestion further up in the hierarchy. This reduces the data center's flexibility as applications have to take the topology into account when transmitting data between servers, for each level of hierarchy the available bandwidth between two servers may be an order of magnitude lower.

The topology itself creates additional inflexibility when it comes to VM migration due to the access routers and Virtual LAN configuration. The access routers impose a specific IP subnet on all servers under them which makes it difficult to transparently move or duplicate a server to a machine connected via another access router. Virtual LANs are typically used in the Layer 2 part of the network to isolate traffic from different customers or services and provide rudimentary QoS. These also have to be reconfigured in the event of VM move.

2.3.2 Virtualization

Modern data center servers are moving towards virtualization in order to achieve cost savings through server consolidation (reduction of power & cooling, increased server density, better resource utilization and better server administration), high availability (better application isolation, better virtual machine migration) and new service opportunities (cloud computing, servers on demand).

A sound infrastructure for the data center should be robust enough to support virtualization of every networking component and should be flexible enough to support both applications on dedicated hardware and applications on virtualized hardware. Management software can fully and dynamically configure parameters such as performance, security policies and subscription ratios.

Supporting the virtualization of a physical platform (switches, routers and servers) in a data center is a valuable use case of OpenFlow. Virtualization of a platform allows for several services to share the same physical platform independently without interfering with each other. OpenFlow allows for flow-based resource allocation inside the network, which can be utilized for virtualization of the networking resources.

The following sections provide two detailed examples of data center virtualization: Multi-tenancy and private Cloud.

2.3.2.1 Multi-tenancy

The fast growth of data centers over the past decade has redefined the way of designing and building data centers. The innovation is fast but often proprietary. There are many data center networking architectures which each hold a unique point in the design space. Different data center operators may employ different architectures based on their understanding of cost and performance, which often implies different requirements on networking hardware. Additionally, networking requirements depend on what type of applications will be running in the data center, an application that mainly serves as a database for video streaming will put a different load on the network compared to an application that performs distributed web indexing (the traffic matrix will differ between a majority of outgoing vs. internal traffic). The different data center applications and their networking requirements raise the question: Can we have a common network which can be configured to suit the requirements of different data center applications? In other words, how can we build data centers that implement multiple network architectures on top of the same underlying hardware?

One possible solution based on OpenFlow is called Ripcord [22]. It provides a uniform control interface to a physical network so as to abstract high-level data center networking solutions from the underlying network infrastructure. This control interface is logically centralized so that schemes using the interface do not suffer from the need to implement complex distributed algorithms.

In addition, Ripcord supports multiple tenants. Each tenant on Ripcord is a logical entity (e.g., data center customers, services, jobs, etc.) requiring a separate treatment in routing/management. These tenants are isolated from each other and can customize their routing and management through Ripcord modules. A researcher may use this capability to evaluate two architectures side by side (or even simultaneously); an experimental data center can host multiple researchers (or customers) at the same time; a multi-tenant hosting service may provide different customers with different logical networks; and a multi-service data center may use architectures optimized for different services.

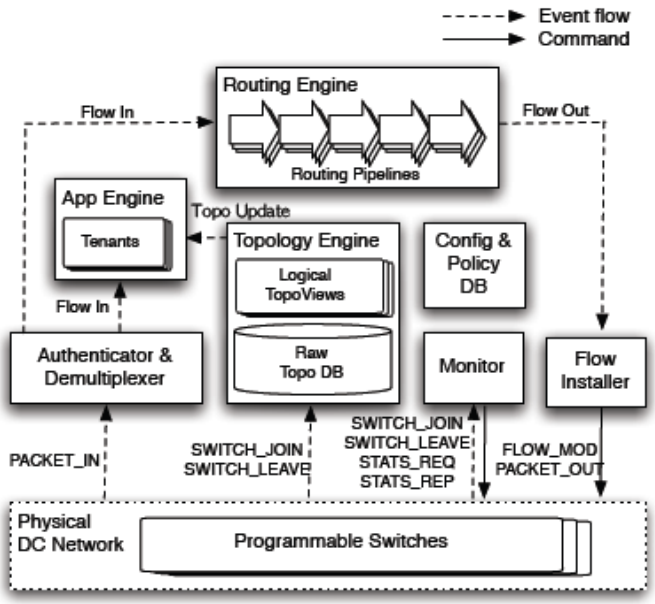

Figure 8: RipCord architecture and event flow

The high-level architecture of Ripcord is shown in Figure 8. It consists of the following seven components:

1. Config & policy DB: is a simple storage for platform level configuration and data center policy information. Administrators configure the database with global network characteristics as well as tenant-specific policies. This centralized configuration provides ease of management. As this module merely stores the configuration, the actual policy enforcement is delegated to other components.
2. Topology engine: maintains a global topology view by tracking SWITCH JOIN as well as SWITCH LEAVE events. This allows for real-time network visualization, expedites fault-detection and simplifies troubleshooting. The component also builds per-tenant logical topology views which are used by app and routing engines when serving a specific tenant.
3. Authenticator-demultiplexer: performs admission control and demultiplexes to the correct application. Upon receipt of a PACKET IN event, it invokes the configuration / policy database and resolves the tenant in charge of the packet. If the packet is not legitimate, the component drops it. Otherwise, it passes on the

routing request to the app and routing engines, as a FLOW IN event tagged with the packet and tenant information.

4. App engine: each tenant can have its own management app. Hence, the management app can be seen as a centralized controller for a particular tenant. This component typically inspects incoming packets in a FLOW IN event and updates its internal state. On receipt of a FLOW IN event, the app engine dispatches the event to a proper app based on tenant information associated with the event.
5. Routing engine: this module calculates routes through a multi-stage process: starting as a loose source route between the source-destination pair, a path is gradually filled through each of the routing pipeline stages. One pipeline stage may consist of zero or more routing modules. Ripcord does not limit the size of routing pipeline. It does, however, enforce the order of stages so as to help verify routing modules' composability.
6. Flow installer: is in charge of translating FLOW OUT event into a hardware-dependent control message to modify the switch flow table. We introduce this indirection layer to make Ripcord independent of a particular switch control technology.
7. Monitor: provides support for passive and active statistics collection from network elements. Passive collection periodically polls switches for aggregate statistics, while active collection is targeted to probe a particular flow in the network. When a switch joins the network, the component records its capabilities (e.g., port speeds supported) and then maintains a limited history of its statistics snapshots. Snapshots contain aggregate flow statistics (e.g., flow, packet and byte counts for a switch), summary table statistics (e.g., number of active flow entries and entry hit-rates), port statistics (e.g., bytes/packets received, transmitted or dropped) and their changes since the last collection.

The requirements of multi-tenancy can be summarized as the following:

R-41	**Providers should have a maximum degree of freedom in the choice of data center technologies.**
R-42	**Data center virtualization must ensure high availability.**

In addition the requirements R-2 - R-8 are of importance for use case 2 as well.

2.3.2.2 Private cloud

Data center, WAN/MAN, and the end user are three of the components that make up the Cloud in the vision of Cloud Computing. However, the existing technologies often treat each component as black boxes, detached from each other. This fact limits the overall cohesiveness of an end-to-end service. For example, the network often views the data center as a black box, meaning the network has no control or visibility (from a standards point-of-view) into the data center. As a network provider, a Cloud-service product may be offered across multiple data centers globally, some of which may be owned by a network provider while others may be owned by a partner/vendor. In addition, multiple Cloud-service products can be offered in the same data centers.

The success of VPN services in the enterprise and the government world is largely due to its ability to virtually segregate the customer traffic at layer 2 and layer 3. The lower the layer that segregation can be maintained, the safer it is for the customers from security and privacy perspectives. Today data centers segregate the customer traffic at layer 7 (application), and there is no standard for extending the VPN into the data center. Network service providers view the VPN extension into the data center, allowing traffic segregation per VPN, an essential necessity to the success of Cloud-services in the enterprises and government markets. Cloud applications (or the virtualization function) SHOULD have the ability to access VPN (including Layer 2/3 VPN) services, to segregate different Cloud services traffic through the network.

One possible solution is to use the OpenFlow on the network and data center domains. The network domain OpenFlow controller will communicate with the data center domain OpenFlow controller so each Cloud-application shall be transmitted over a pre-defined set of VPN connections, and each VPN utilizing the application shall be transmitted over a sub-set of application connections. The OpenFlow controller will make sure that each cloud application has its own independent routing table on both the data center and network domains.

Today, data center virtualization is totally handled by data center servers and hypervisors. The entire process is invisible to the underlying networks and the users. There shall be a way that the network can influence some virtualization functions that are important to the concept and spirit of the VPN.

The following lists the requirements of data centers interworking with the existing L2 and L3 VPN services.

The requirements with respect to data center virtualization are:

R-43 The Private Cloud provisioning and management system shall have the ability to dedicate a specific share of resource per VPN.

R-44 Each VPN may have the exclusive access to the specific share of resource.

R-45 Each VPN shall have the ability to hold the requested resources without sharing with any other parties.

R-46 Each VPN may have the ability to limit the stored data mobility to a certain geographic region confinement (country/state).

The requirement with respect to network restoration and private cloud service restoration is:

R-47 The restoration capability awareness should be scalable.

The requirement with respect to QoS synchronization is:

R-48 The virtualization functions QoS requirement should be synchronized with VPN service.

Today's Cloud traffic balancing and congestion avoidance is purely data center-based. The network condition is not taken into consideration. The requirement with respect to load-balancing and congestion avoidance is:

R-49 The VPN extension should support the network condition to be used for the traffic balancing and congestion avoidance decision-making.

The requirement with respect to cross-layer optimization is:

R-50 The VPN resource requested by the server can be optimized by statistical multiplexing of the resource.

For example, for each VPN resource, it is possible to configure committed bandwidth for each QoS resources and peak bandwidth for best-effort traffic, and the peak bandwidth resources can be shared by different VPN services.

The automatic end-to-end network configuration will reduce the operational cost and also the probability of occurrence of misconfiguration. The requirement with respect to automated end to end configuration is:

R-51 The VPN Extension should support the automatic end-to-end network configuration.

The requirement with respect to end-to-end quality of experience is:

R-52 Quality of Experience management should be supported.

Quality of experience (QoE) management refers to maintaining a set of application / service layer parameters within certain threshold with an objective to retain the user experience for a specific service.

The requirements with respect to OAM considerations are already covered in R-25, R-26 and R-27.

2.3.3 Load balancing

In order to prevent a hotspot inside the network or among the servers, and to properly utilize the available resources, it is crucial to implement efficient load balancing. Most of the commercially available load balancers operate in Network Address Translation mode sitting in the data path between the routed and switched network, meaning that all incoming traffic to a data center must be routed through a load balancer which is responsible for deciding to which server the incoming requests should be forwarded, it also has rewrite packet headers in order to redirect packets. This approach reduces the scalability of a data center, and in addition to that, the load balancing is only performed on the server layer and the distribution of the load inside the network is not taken into account. Furthermore, network configuration, which is necessary to support VM migration, is a complicated task in this approach.

Other proposed networking architectures are more extensive and provide load balancing of the internal network traffic as well as external. One example is VL2 (details see annex B) that uses a Closed network topology, centralized address lookup scheme and IP-in-IP encapsulation to implement valiant load balancing in order to spread traffic uniformly over a network of unmodified switches [25]. Another example is PortLand which modifies the switches to route based on a pseudo-MAC header, and aims to eliminate switch configuration [29]. Other researchers have proposed Monsoon [26] and FatTree [30]; Trill [27] (which could also be used in a data center environment) and DCE [28] have been proposed as standards.

In an OpenFlow-enabled data center, load balancing can be realized in a more efficient and integrated way. In fact, efficient load balancing necessitates flow detection/classification as well as session persistence, which is the ability to direct all requests of any given user to the same back-end server during a session or consecutive sessions. These

requirements are all readily supported in the OpenFlow concept. A sample scenario for load balancing based on OpenFlow is depicted in Figure 9.

In this scenario, a load balancing unit is employed as an application in the OpenFlow controller. The unit is composed of three main sub-units, which are: network and server monitoring, network and VM control, and load balancing algorithm sub-units. This approach has several advantages over the conventional ones; first, in this approach load balancing is carried out in the control layer, which, besides increasing the scalability, allows the load balancing algorithm to be easily modified or updated if needed. More importantly, since OpenFlow can be aware of flows in multiple layers, e.g., TCP flows and IP flows, the OpenFlow-based approach can realize both transport and network layer load balancing, leading to a cross layer optimization.

Using Ripcord, different load balancing schemes can be implemented as load balancing sub-units and deployed on a per service per customer basis. If one integrates server load measurements, VM configuration and mobility into the same system, one may adapt to network and server loads by automatically deploying/moving VMs and reconfiguring the network.

Nonetheless, there are some issues that require further investigation in this approach. One of these issues is the functions which are not directly supported through OpenFlow but are available in commercial load balancing products, e.g., Layer 4 proxies, Layer 7 load balancing and SSL off-loading. Other issues may be reliability and security issues that could be associated with a centralized OpenFlow controller and the number of flow entries that fits in a single switch TCAM.

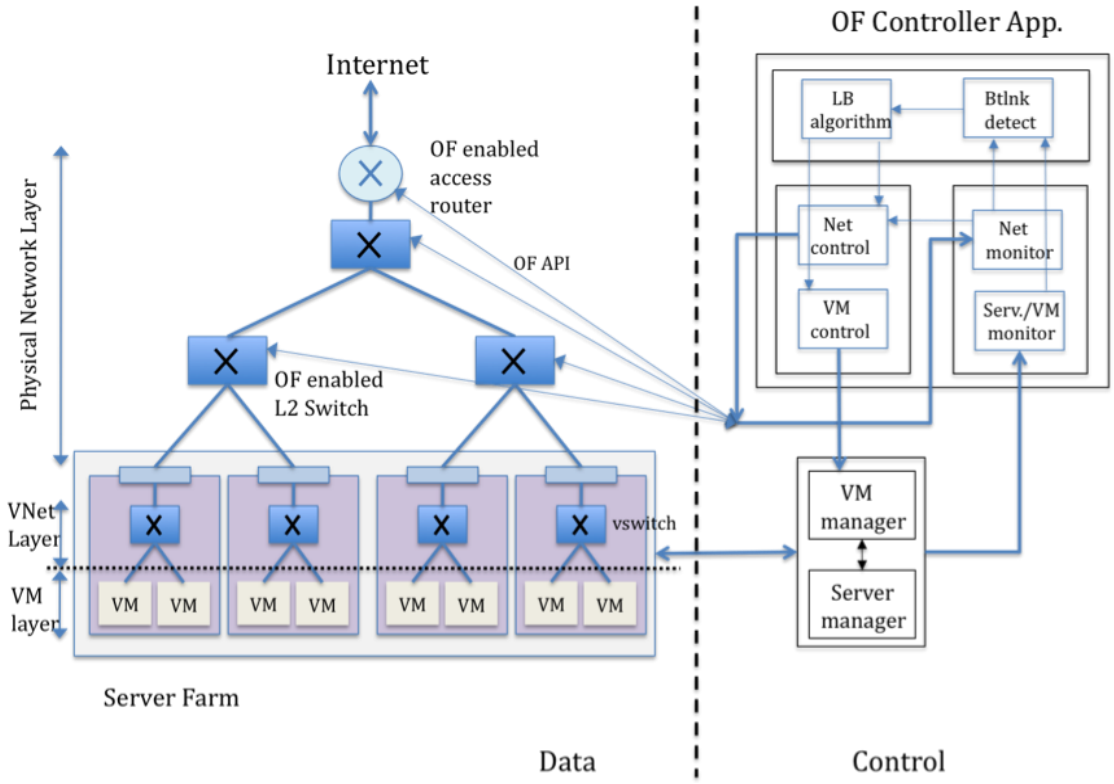

Figure 9: A sample scenario for load balancing in a data center using OpenFlow

The requirements of the load balancing are summarized in the following.

The requirements with respect on data path elements (switches/routers and links) are:

R-53	**The data path element should expose to the load balancer/network controller the information regarding their availability, connections to other elements and load situations.**
R-54	**The data path element should provide an API exposing mechanism that can be used to configure for switching/routing flows of packets.**

The requirements with respect on server/VM manager are:

R-55	**The Server/VM manager should expose to the load balancer/network controller the information regarding the operation of VMs including availability, load situation and the association to the servers.**

R-56	**The Server/VM manager should provide an API exposing mechanisms that can be used to control the instantiation and migration of VMs across the server farm.**

The requirements with respect on load balancer are:

R-57	**The load-balancing solution should support L2-L7 flow detection/classification.**
R-58	**The load-balancing solution should provide session persistence.**

2.3.4 Energy efficiency

Optimizing the operation of a network with respect to energy consumption is a hot research topic in the area of networking. Data center networks are usually designed for peak traffic conditions; therefore, during the times when the intensity of requests to the servers is small, some switches could be put into the sleep mode to save energy. This might require that traffic from lightly loaded links are rerouted and concentrated on other active routes. This implies that traffic engineering as an important part of this process could be realized via OpenFlow very well. In fact, some modifications to the sample scenario, presented in Figure 9, can turn it into a mechanism for energy efficiency in data centers. For this purpose, the load balancing and control sub-units have to be appropriately adjusted.

The same set of requirements on the data center equipment as explained in Section 2.3.3 is applicable here.

R-59	**The data path element should provide an API exposing mechanisms for switching the data path element between sleep/normal operation modes.**
R-60	**The data path element should expose metrics that can be used by energy optimization algorithms.**

2.3.5 Network management and configuration

The successful operation of a data center depends, among other things, on optimal configuration of switches and routers in the network as well as regular monitoring of their operations. Taking into account the large number of servers and networking equipment in a data center, these operations become quite challenging tasks.

The following section provides two examples of Layer 2 data center management and configuration: IEEE 802.1Qbg Edge Virtual Bridging [23] and IEEE 802.1Qbh Bridge Port Extension [24].

As virtualization and high density servers are deployed, we increase the number of complex bridges. Even without virtualization, the same challenge exists: the sheer number of blade racks and servers with their associated bridges is growing dramatically, which in turn leads to a significant increase in the complexity of Ethernet networking in the data center. The complexity may be reduced by aggregating the more complex bridging functions onto fewer bridges and by collapsing bridge layers from a management perspective.

IEEE 802.1Qbg Edge Virtual Bridging allows multiple virtual stations to share a common bridge port to obtain the services of bridge relay and also enables coordinated configuration and management of bridge services for virtual stations.

Typically there are many Virtual Machines (VM) instantiated in a single physical end station as indicated in Figure 10. Each VM contains at least one virtual NIC (vNIC) that is associated through the hypervisor with a physical NIC. To create this association, hypervisors have incorporated Virtual Ethernet Bridges (VEB) into the physical end station effectively adding one or more Ethernet switches per end node. A VEB is a frame relay service that supports local bridging between multiple virtual end stations (an internal private virtual network) and optionally the external bridging environment. A VEB may be implemented in software as a virtual switch (vSwitch) or as embedded hardware within a Network Interface Controller (NIC). Each vNIC is associated with a Virtual Station Interface (VSI).

VEB packet forwarding supports both traditional end station-to-adjacent bridge as well as local VSI-to-VSI packet forwarding. A VEB forwards packets as follows:

- VEB forwards packets based on the MAC address and optionally via a port group or VLAN identifier.
- VEB forwards packets from a VSI to the uplink from an adjacent bridge or between co-located VSI.
- VEB supports only a single active logical uplink.
- VEB does not participate in or affect spanning tree operation.

VEB solutions have been shipping for a number of years and are available from multiple suppliers. By definition traffic between VMs connected to a VEB stays within the server. However, some clients prefer the traffic to be sent through an external switch, so the external network's access and security policies can be applied to the traffic. To address this type of requirement, a Virtual Ethernet Port Aggregator (VEPA) is proposed.

A Virtual Ethernet Port Aggregator (VEPA) is a capability within a physical end-station that collaborates with an adjacent bridge to provide frame relay services between multiple co-located virtual machines (VMs) and the external network.

A VEPA collaborates by:

- Forwarding all station-originated frames to the adjacent bridge for frame processing and frame relay.
- Steering all frames and replicating multicast and broadcast frames received from the adjacent bridge to the appropriate VM destinations.
- A VEPA takes advantage of a special reflective relay forwarding mode (i.e., allow forwarding back out the port a frame was received) on the adjacent bridge to support inter-VM communication within the same physical host.
- Similar to a VEB, a VEPA may be implemented in software or in conjunction with embedded hardware within a NIC.

A VEPA provides a number of benefits but it too has limitations:

- Promiscuous support – to support a promiscuous VSI, a VEPA address table must be configured with all VM source MAC addresses. This requires either adding MAC address learning support or provisioning large address tables. Either option increases implementation costs and complexity.
- Support for simultaneous VEB, VEPA, and directly accessible ports on the same physical link – the adjacent bridge can only process a frame based on its contents and therefore lacks sufficient information to delineate between these three operating modes.
- Hierarchy of unrestricted physical ports – normal bridge learning and flooding is not possible due to the lack of information within a frame.

To address these limitations, IEEE 802.1Qbc-2011 is applied. This standard enables multiple virtual channels to be multiplexed on a single physical LAN – referred to as S-channel functionality. Individual S-channels are delineated by a tag which is added to the frame and processed by S-VLAN components (a bridge component) which are logically inserted into the adjacent bridge and the physical end station below the virtual bridge layer as illustrated in the following figure. The S-VLAN component recognizes, inserts and removes service VLAN tags (S-Tag) to enable multiple S-channels in the bridged network. Adding an S-VLAN component to an end-station allows VEPA, VEB, and individual VSI to operate independently and simultaneously. Each VEPA, VEB, or individual VSI operates over its own virtual uplink instantiated by a pair of S-VLAN components - one in the adjacent bridge and one on the end-station.

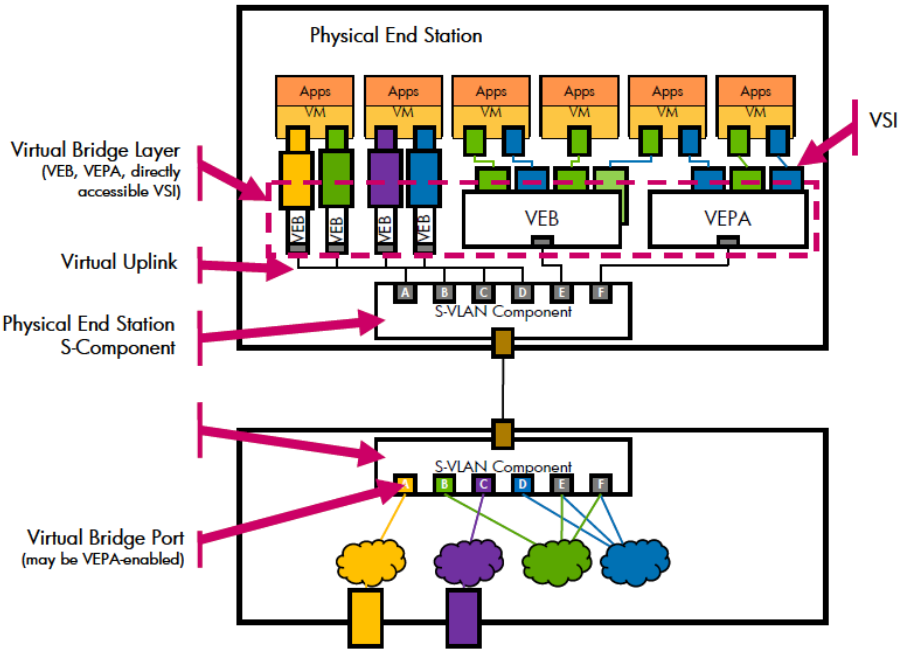

Figure 10: Edge virtual bridging architecture

IEEE 802.1Qbh Bridge Port Extension will utilize the Edge Virtual Bridging capabilities to reduce management complexity by aggregating the more complex bridging functions onto fewer bridges. The purposes of this project include:

- To reduce the management cost of networks comprising large number of bridges (such as those commonly found in data center environments) through significant reduction in both the number of devices to be managed and the management traffic required.
- To decrease the total cost of ownership by reducing initial capital expenditure along with management and operational costs.

The basic idea is to design a port extender that extends the bridge reach. A port extender attaches to a MAC port of an 802.1Q bridge and provides additional MAC ports that are logically ports of the 802.1Q bridge to which it is attached (i.e., the "Controlling Bridge").

As in the following figure, we push complexity up into the components of which we have fewer (bridges) and attempt to simplify the components that appear in higher quantities (NICs).

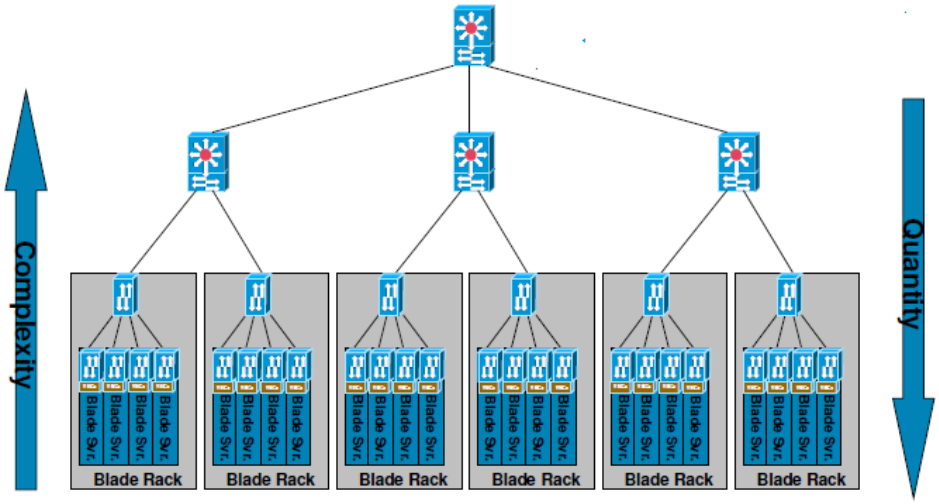

Figure 11: Bridge port extension example

The requirements of the network management and configuration with respect to management capabilities and cost effectiveness are summarized as follows:

R-61	**The network management and configuration must provide predictable and consistent capabilities.**
R-62	**The network management and configuration should provide a cost vs. benefit ratio better than today's approaches.**

2.4 Use Case 3: IP backbone including multilayer aspects

2.4.1 Introduction

The previous sections have covered use cases for carrier-grade operation of packet networks. Most of these use cases require an adequate level of quality of service for maintaining the SLA negotiated between operator and customer, e.g., for triple-play services or mobile backhauling. While packet switching increases utilization of capacity in communication networks, maintaining quality of service and SLAs in pure packet controlled environments is a challenging task with significant complexity for scheduling and traffic shaping. A controlled load approach is a feasible approach for ensuring a minimum QoS level, but decreases resource utilization considerably.

In addition, this use case differs in two aspects from the previous one. First the IP backbone is the overall network part which interconnects with other networks and therefore requires different protocol solutions. Second, the IP backbone could be considered as a separate network with different design paradigms, for example highly aggregated traffic and traffic matrixes.

Regarding the first aspect, the interconnection with other networks is done with an exterior gateway protocol (EGP) such as BGP. More details for a sample network design can be found in Annex A: Example of carrier IP backbone - topology and functionality.

R-63	**The OF domain should be able to interact with other domains through EGP.**

2.4.2 Extension of the scope towards circuit-based networking

The physical medium of most of the access aggregation and all of the core networks is nowadays fiber. The gap between the total bandwidth available in a single fiber and the electronic signaling speed of even the fastest serial transmissions is estimated to be at least three orders of magnitude (comparing the total 25 THz within the three transmission windows to the 25-33 GHz of a polarization multiplexed QPSK 100 Gbit/s Ethernet). For this reason, WDM transmission has proven to be valuable, as it multiplexes different and potentially transparent data streams onto several carrier wavelengths. Typical WDM systems of today operate up to 80 channels per fiber. WDM channels in one fiber are typically of the same granularity. In core networks, this will be 40 (100) Gbit/s channels provided in a 100 GHz ITU grid, while in metro or access aggregation networks the individual wavelength will often carry a bitrate of 10 Gbit/s @ 50 Ghz grid. The transmission format on these WDM channels has been standardized as an extension of the traditional SONET/SDH PHY. ITU-T G.709 defines a so-called digital wrapper around a SDH signal that adds FEC in order to extend the reach of a transmission without frequent regenerators [45].

Besides the fixed circuits (TDM or WDM), that are made up of the OTN, Ethernet and MPLS can be found that create virtual circuits with properties that are a function of a local output port scheduler. Also wireless technologies like 3G/LTE (CDMA/OFDM) or even Wireless LAN in its current version (enhanced DCF and HCF mode of operation) support QoS enhanced transport channels.

Inevitably, the question arises how to detect and make use of such dedicated QoS enabled transport channels in a carrier-grade network, finally harmonizing operation of packet and circuit switching under common Split Architecture. In addition, the level of detail of lower layers, which should be exposed to the control plane and the controllers inside a Split Architecture, has to be defined.

This problem space has already been discussed in the context of GMPLS and control of optical networks (see [31]). Two options can be distinguished for the control plane of such networks:

- Client/server control planes (overlay model)
- Peer model (use a single GMPLS controller)

In the peer service model, higher layers are granted access to lower layer details in order to configure paths on several layers of a communication network. In contrast, the hierarchical (overlay) model encapsulates all lower layer details, thus effectively preventing higher layers from adapting to lower layer information.

The virtual circuits mentioned above differ in one major point from the circuit technologies discussed: VLANs and MPLS use Ethernet as the encapsulation format. This means that no specific interfaces are needed at the OpenFlow switches and all actions (forwarding and modification of headers, i.e., processing) still require specific commands, in other words, the OpenFlow hardware has to be VLAN and/or MPLS capable.

R-64	**Information of a lower layer has to be exposed to a higher layer appropriately.**
R-65	**A data path network element should enable control plane applications to poll detailed configuration attributes of circuit-switching capable interfaces.**
R-66	**A data path element should enable control plane application to set configuration attributes of circuit-switching capable interfaces.**
R-67	**The Split architecture may allow implementing additional OAM solution when the interface does not provide any.**

A number of requirements from previous sections also apply to this use case. These are especially the requirements regarding network management, policies and OAM with R-1 to R-8, R-11 to R-18 and R-21 to R-27.

2.4.3 Examples of OpenFlow integration in carrier IP backbones

A backbone network implementation can be considered as at least two layers: IP/MPLS and an optical layer that consists of WDM and/or OTN links. The optical layer may show different degrees of reconfigurability. The WDM/OTN layer may be split into two sub-layers by introducing OTN cross-connects on top of a wavelength-switched network. As a result, several packet optical node architectures can be enumerated as examples on various implementation of a multilayer network.

The simplest case, corresponding to the IP backbone example in the Annex A (section 5), is a static, pre-configured optical network with interfaces that create point-to-point links between router ports.

Regarding a Split Architecture, and specifically OpenFlow, the degree of information that a lower layer exposes to a higher should be left to the implementer. Most of this information will be created by controllers based on available information and alarms coming from OpenFlow-enabled hardware.

2.4.3.1 IP/MPLS over static optical links

For what regards the MPLS backbone of carriers, the introduction of OpenFlow will not change the existing network architecture, as OpenFlow-enabled Ethernet/MPLS switches will work in the same place and switching format of MPLS routers today. In fact for the outside world it will not be visible that the MPLS equipment that is being deployed is OpenFlow-controlled.

Figure 12 depicts node architectures where the WDM/OTN layer is static in the sense that the connections are statically provisioned providing a fix "virtual" topology denser than the physical one. As the optical layer is static and no dedicated OTN digital cross connects are deployed, OTN features are implemented in so called transponders included in switching elements to the client or more common, switching elements outside the client, but part of the terminal sub-system of a separate WDM-system. Therefore, the key use of OTN is to provide a reliable framing and transmission mechanism over longer distances. The actual switching happens at the client packet layer.

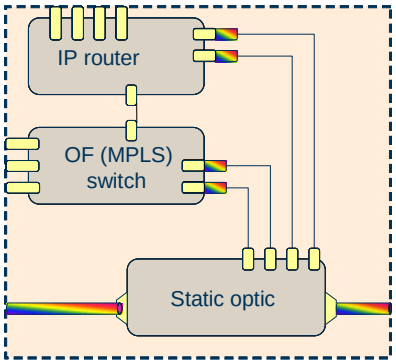

Figure 12: Combined client layers over static optical one

2.4.3.2 IP/MPLS over dynamic optical networks

The architecture depicted in Figure 12 can be radically extended by substituting the fix optical multiplexer (e.g., an AWG) for a reconfigurable element (e.g., a ROADM). This makes the optical layer dynamic by allowing the reconfiguration of the optical routes as well (see Figure 13). Reconfigurability of the optical network can be motivated by the need for multi-service delivery platforms that integrate the IP traffic between routers with other, potentially non-IP traffic, on the same fiber network. This other traffic may be the support of leased lines for Ethernet or streaming of high data rate video. In this case, the IP traffic is multiplexed with other traffic onto higher-order OTUs or on different wavelengths.

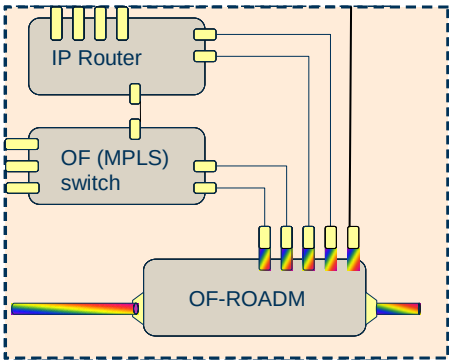

Figure 13: Node architecture with reconfigurable optical layer

The last node architecture (see Figure 14) provides the most flexible system: it comprises of dynamic WDM layer (through a ROADM) as well as it implementing an OTN switch in synergy with the ROADM. This construct provides an integrated WDM/OTN switching technology supporting both optical and TDM switching. Multiple packet switches could be connected as client switches sharing the same OTN/WDM transport infrastructure. Finally, such architecture will result in a full three-layered technology stack.

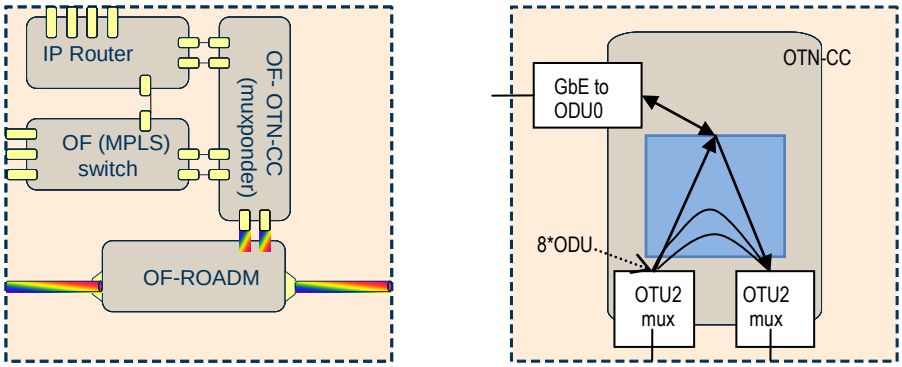

Figure 14: Complex node architecture with dynamic OTN and WDM layers, the figure on the right provides more detail of the OTN crossconnect.

A multilayer network could contain any combination of the abovementioned node architecture classes.

2.4.3.3 Specification of interface types

Communication between hardware and controller is typically initialized by the discovery of available interfaces (ITU-T G.805 [46] or G.809 [47] adaptation functions) in the data path. These interfaces encapsulate a certain information format (e.g., Ethernet) in a lower layer by adding some form of header and encoding. It is necessary to describe the nature of this encoding to be able to match to another interface of a compatible type in another switch. For example, the description of the interfaces is rather simple in Ethernet switching, as despite a variety of Ethernet standards, many of the interfaces are interoperable on the basis of RJ45 and auto-negotiation. There are no other values to be set for Ethernet interfaces other than up/down in practical operation. For other interfaces, the number of parameters to be configured is substantially higher. In the following, three classes of interfaces are considered that may be found in a carrier network: GFP, OTN, and tunable optical interfaces.

An interface switching capability descriptor as defined in RFC 4202 may be the syntax of the required information. It should be noted that the specification of the interface types states which parameters can be set, not how to set these or their values.

In the following, some examples for interface descriptors are presented. The internal interfaces in Figure 12 to Figure 14 also adapt one data format to another. These adaptation functions add framing information, compute error detection information (CRC), encode, and multiplex different streams into a new flow. For MPLS, the adaptation function takes care of removing the Ethernet header; label push/pop, associated TTL copy operations, and re-framing into a Layer-2 Ethernet header. Mapping of Ethernet frames into OTN can be performed by several options: GFP-F, GFP-T, ODUFlex, OTU-4 all take Ethernet frames as input and map into Optical Data Units (ODU).

Generic Framing Procedure (GFP-F, GFP-T)

GFP (ITU-T G.7041) has been introduced as a generic way of encapsulating non-voice-centric data into SONET/SDH. It overcomes some of the limitations of so-called PoS interfaces by distinguishing multiple user streams and providing better delineation functions.

GFP fulfils the classical two ISO/OSI layer-2 functions LLC and MAC: it organizes streams into logical links (using Channel IDs) and maps them into a byte-oriented PHY layer (which is the abstraction that SONET/SDH offers).

GFP distinguishes between frame-mapped (GFP-F) and transparent-mapped (GFP-T) operation, the latter offering lower delays because of a smaller granularity of the mapping (GFP-T maps 8B10B words into [520,536] superblocks and transmits these immediately instead of the store-and-forward operation of GFP-F).

GFP adds simple header information that consists of multiple hierarchies. The outer header (core header) of a GFP frame only contains 2 octets of packet length information followed by 2 octets of CRC that secure the integrity of the frame delineation. Further headers are distinguishing payload type and extension headers add Channel ID of the individual stream that is multiplexed. GFP therefore is an adaptation function in the sense of ITU-T G.805. The single defined termination function is client signal fail (CSF).

Optical Transport Network (OTN)

For OTN, the physical interfaces are named OTU-1 to OTU-4, respectively, and denominate bit rates between 2.5 Gbit/s and 100 Gbit/s. The OTN defines a hierarchy of layers similar to SONET/SDH. The OCh defines the optical transmission, the OPUk encapsulates the client signal (SONET/SDH or Ethernet) the ODUk performs functions related to multiplexing of electrical signals. OTU adds a forward error correction (FEC) that, based on a Reed-Solomon encoding, is able to enhance the OSNR by around 6 dB, effectively extending the range of un-regenerated optical transmission.

OTN was designed to natively encapsulate SONET/SDH signals, but a number of mappings of other tributary signals, especially Gigabit Ethernet (GbE), have been defined.

The basic data rate (OTU-1) was designed to encapsulate a full STM-16 (OC-48) signal. A mapping of a single GbE into this data rate of ~2.5 Gbit/s would be inefficient, so a sub-container was defined (ODU0) that is meant for the transport of 1 GbE. Two of these ODU0 are then mapped into one OUT-1 signal. This multiplexing of several lower-order ODUs into one higher-order OTU has the advantage of de-coupling the client data (ODU0, 1 Gbit/s) rate from the transmission rate (say, OUT-2, 10 Gbit/s).

Optical interfaces

However, optical interfaces in a wavelength switched network are currently less standardized and require a complex description. Examples of parameters for this description are the grid in which the interface operates (ITU 100 GHz or 50 GHz), the tunability and range, wavelength band, existence of optical or electronic equalization, range of output power, and many more.

In summary, the nature of available external and internal interfaces has to be exposed from the switch to the controller. The less these interfaces are standardized, the more individual parameters have to be added to the OpenFlow interface specification. An external dictionary where all parameters for specific interfaces are listed would help in limiting the amount of configuration traffic.

2.4.3.4 Handling of heterogeneous forwarding abstractions (flows & circuits)

As long as the (outer) frame format is Ethernet, the forwarding abstraction used in OpenFlow is that of a flow. MPLS label switched paths, VLANs and sequences of IP packets only differ in the way they are established, not in the actual representation in the flow table. Inside optical circuits, however, flows are typically not recognizable. Therefore there will not be any "unknown frames" that have to be sent up to a controller. Instead of the defined parameters of flows, specific parameters of the dedicated circuits (e.g., optical circuits) are required as forwarding abstraction.

A combination of input port and some form of channel identifier is what characterizes a circuit. This circuit can be switched to another output port or mapped into a different format and potentially multiplexed. As an example, Figure 14 shows a GbE line that is mapped into ODU0 before being switched towards an OTN muxponder that multiplexes this ODU0 together with seven others into an ODU2 and transponds this into the OTU-2 format.

The changes to the OpenFlow 0.89 specification that were proposed for circuit switching (see Figure 15) foresee a possibility of combining different port types and therefore an implicit declaration of the conversion. This is clearly a workaround for a missing piece in the current OpenFlow architecture. The next section will deal with this in more detail, but the Interface Adjustment Capability Descriptor (IACD) as defined in RFC 6001 [34] addresses the problem of the internal conversion interface: So far all traffic engineering extensions of routing protocols assume that the backplane of a node is an ideal medium (no delay, infinite bandwidth) and that all constraints can be mapped onto the links, and their respective interfaces. IACD acknowledges that node-internal format conversions, i.e., layer-crossings, are typically bandwidth limited. As an example, a hypothetical Gigabit Ethernet switch with one OUT-2 interface and 10 GbE input ports can still be limited in throughput by the number of available ODU0 mappers that are internally available.

A cross-layer forwarding abstraction therefore has three characteristic parameters: The forwarding entry between input port and internal format converter, the parameters of this converter (mapping of one format into another requires some form of adjustment and a new address in a different format), and the forwarding entry between the converter and the output port. Figure 15 shows the two tables (named packet and circuit) and the format-conversion (processing unit) in-between.

2.4.3.5 Processing interface

So far, Split Architecture means the clear separation of data and control path and, together with plain forwarding and processing in the data path, this can be advantageous. The definition of the OpenFlow interface started with the pure forwarding. Gradually more and more features are defined in the interface that allows certain forms of processing in the data path.

For example, the forwarding of an IP packet involves swapping MAC headers (replace source MAC with own, destination MAC with that of the IP next hop), decrementing of TTL, and re-calculation of the Ethernet CRC. It would be expensive to do this in the controller, so the means have been found to

1. put the required functionality into hardware
2. make this functionality configurable from the controller.

The same applied to the introduction of any other new functions (e.g., ARP, VLANs and MPLS label pop/push). While one may in principle continue along this road, new functions and addressing schemes have to be introducible before

being cast into silicon. Therefore it would be desirable to have a separate interface that either configures special purpose hardware or even general-purpose hardware (e.g., FPGAs) that could be loaded with executable code for specific data conversion functions. The complexity of the interface specification for OTN and wavelength switched networks can be seen in Figure 15.

Figure 13 introduces several interfaces for the forwarding abstraction (one for circuit, one for packet) along with a configuration interface (Config IF). This interface is used to configure the parameters of the interfaces (shown here only for the internal processing unit). These parameters are discovered through the specification of the interfaces, but need to be modified by the controller when anything other than a pure overlay model is intended. The configuration of the interfaces in an overlay is opaque and can be done manually or automatically.

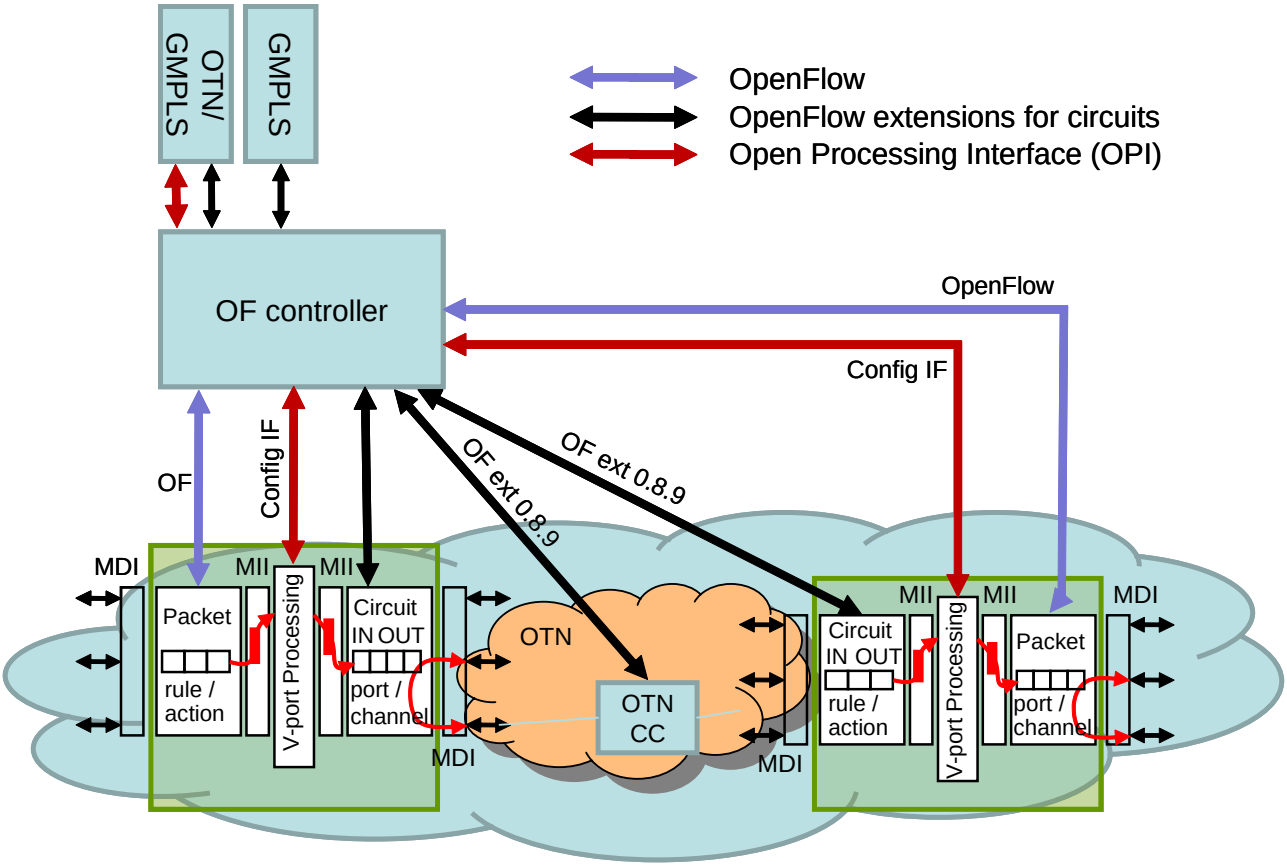

Figure 15: MPLS nodes connected through a switched OTN. The hybrid OF switches contain two forwarding elements and an adaptation/termination function in-between.

2.4.3.6 Configuration of termination functions – OAM in the data path

A general prerequisite for restoration is the existence of triggers that indicate connectivity or loss thereof, i.e., the need to restore LSPs. The individual strategies (path or link restoration) have been briefly mentioned in the SPARC deliverable D3.1 section 10, but much more sophisticated strategies exist.

Triggers are produced by processes that monitor the link state. These are part of ITU definition for termination function [46]. The individual monitoring function depends on the nature of the link; Ethernet has defined its own connectivity fault monitoring functions (IEEE 802.1ag and ITU Y.1731), while OTN and GFP have own in-built monitoring. For example, OTN is capable of supporting so-called tandem connection monitoring (TCM). This means that up to 6 levels of overlays can be monitored within one OTU stream. It must be possible to define the current TCM level to which the OAM message belongs.

The plethora of different connectivity monitoring functions creates an architectural problem for OpenFlow. If the monitoring functions are to run in the controller then an insertion of new monitoring frames from the controller to the switch is easy and does not require standardization, but may lead to serious performance bottlenecks. Required detection speeds prevent a fast reaction to the failure, as the OpenFlow interface would be swamped by monitoring frames when it should transmit the resulting TCAM entry of a Fast Reroute.

Table 2: Some reasoning about the position of OAM functions in the architecture

	OAM in the controller	OAM in the data path
OF link	Reaction speed to failures leads to high number of monitoring frames across OpenFlow interface	Separation of control and data path for OAM (state machine in the data path controlled by corresponding network control app in the controller)
OF standardization	Little to no requirements w/ regard to OpenFlow standardization	Protocol-specific extensions to OpenFlow needed (for each monitoring protocol)

Similar considerations apply for the monitoring of queue status. Here, thresholds in the packet queues may be used to trigger re-configuration of circuits that connect switch ports. This way additional transport capacity can be provisioned for overloaded links provided that spare interfaces exist.

Table 3: Required parameters for including OTN and wavelength switches. For MPLS, these are taken from OpenFlow specification 1.1.0 [32], other values were proposed in [33] for OpenFlow 0.8.9.

	MPLS	OTN	Lambda switch
Interface description	n/a interface switching capability descriptor (ISCD, RFC 4202)	/*Description of a physical circuit port */ Struct ofp_phy_cport { } enum ofp_port_features { OFPPF_OTU1 = 1 << 29, /*OTU-1 2.666 Gbps */ OFPPF_OTU2 = 1 << 30, /*OTU-2 10.709 Gbps */ OFPPF_OTU3 = 1 << 31 /*OTU-3 42.836 Gbps */}	enum ofp_port_features { OFPPF_X = 1 << 20, /* Don't care – applicable to fiber switch ports */} Further parameters: - Grid (100 or 50 GHz) - Tuning range - Electronic or optical equalization - Output power, input sensitivity - Filter bandwidth
Forwarding abstraction	struct ofp_match{ uint32_t mpls_label; /* MPLS label. */ uint8_t mpls_tc; /* MPLS TC. */}		enum ofp_port_swtype { OFPST_WAVE = 1 << 8, /* Wavelength switch */ OFPST_FIBER = 1 << 9 /* Fiber switch */}
Processing	OFPAT_SET_MPLS_LABEL, /* MPLS label */ OFPAT_SET_MPLS_TC, /* MPLS TC */ OFPAT_SET_MPLS_TTL, /* MPLS TTL */ OFPAT_DEC_MPLS_TTL /* Decrement MPLS TTL */ OFPAT_PUSH_MPLS, /* Push a new MPLS tag */ OFPAT_POP_MPLS, /* Pop the outer MPLS tag */		

3 Overview of derived requirements

In the previous section, the different use cases are described and requirements have been derived. In total, 67 requirements have been identified. In principle, several requirements appear more than once in the different use cases and would increase the number to approximately 100. These "redundant" requirements have not been included in multiple use case sections. Most of requirements have been identified in use case 1, with 40, followed by use case 2 with 32 (including 10 already covered in use case 1) and 25 in use case 3 (including 20 already covered in use case 1 and use case 2).

The study and analysis of the requirements comes to the conclusion that, with respect to importance in the context of SPARC, the set of requirements can be subdivided in different clusters. So the total of 67 requirements had to be reduced in order to concentrate only on those requirements that are not already fulfilled with respect to existing architecture concepts and available implementations. The selection process is based on the opinion of the technical experts of the projects. Besides this rating, one important indicator has been the use of the key words "must", "should" and "may" as specified in IETF RFC 2119. But, there are three other important indicators as well. The first one details the relationship to the different use cases. As indicated in the previous paragraph, some of the requirements are relevant for more than one use case. These requirements are not stated explicitly, but references to the relevant paragraphs in the previous part of the document are included. Overall, the remaining requirements were prioritized with respect to overall importance, fulfillment in existing architecture concepts and/or existing implementations and their relevance for one or more use cases. Regarding the latter aspect, the relevance for more than one use case is expected to be more important. A detailed list covering all aspects of this prioritization process is provided in a separated document.

Overall, four groups of general, important requirements could be identified. The first group ("A") covers all required modifications and extensions for the data path element or the Split Architecture itself. The other three groups deal with needed extensions of carrier-grade operation of ICT networks. The aspects related to the operation of an ICT network are authentication, authorization and auto configuration (not to be mixed up with "AAA", as accounting is use-case-specific) (group "B"); OAM in the sense of facilitating network operation and troubleshooting (group "C"), network management, security and control of the behavior of the network and protocols (group "D"). Within network management, the aspects for the use of policies in network environments are included. The detailed list of requirements is as follows:

A. Modifications and extensions of data path elements and split architecture: R-5, R-28…R-40, R-64…R-66

B. Authentication, authorization and auto configuration: R-6, R-7, R-10, R-23

C. OAM: R-25…R-27, R-60

D. Network management, security and control: R-6, R-7, R-12, R-13, R-21, R-60

4 Conclusions

In conclusion, the deliverable fulfils its objective to provide an initial description of use cases of a Split Architecture and definition of requirements for carrier-grade operations, derived from these use cases. Overall, 67 requirements are derived, covering in their characteristic a broad range from very generic to very use case specific. It is expected that the majority of the requirements will have a direct potential impact on the ongoing discussion on the Split Architecture definition.

The different use case areas were selected to cover all important aspects of a carrier environment as defined for next-generation-networks by ITU-T. Namely these areas are:

- Access/aggregation domain of public telecommunication networks including mobile-backhaul, software-defined networking and dynamic control composition
- Application of Split Architecture in the data center domain (i.e., the service stratum of NGN)
- Multi-layer networks (i.e., IP, Ethernet, OTN circuit switching) with specific characteristics for different packet and circuit-based technologies and a more detailed analysis of IP/MPLS transport via OTN in the backbone

For all those areas, Split Architecture, e.g., split of forwarding / control and split of control / processing is a promising approach to achieve better flexibility and programmability of networks and OpenFlow is one candidate for this approach.

In the access/aggregation domains of many carriers, an Ethernet-based aggregation is used to collect and concentrate traffic for residential and business customers. State-of-the-art access/aggregation solutions and next generation extension of MPLS towards this network domain are technically described and corresponding requirements are derived. The requirements cover functional and quantitative requirements for typical sizes of those networks. Implementing a Split Architecture by means of OpenFlow in those MPLS domains may give the chance to have a centralized control and make use of commodity hardware, without influencing the interworking between the different MPLS domains of the telecommunication network. Access/aggregation domains are a very promising area of carrier networks to implement a Split Architecture because of the strong overall requirement on a very high degree of functional flexibility and in most aspects predictable structures of traffic flow. Moreover OpenFlow would allow virtualization and thereby support multi-operator scenarios or service separation within the responsibility of one operator. In addition to requirements on functions and characteristics, requirements on dimensioning and scalability are also provided. The latter are oriented at a typical Western European aggregation area. Until now, there has been no profound analysis on the scalability of OpenFlow and Split Architecture in a carrier environment. The extracted information on the dimensioning collected in this deliverable tries to contribute to this analysis and sets the stage for a detailed study.

As mobility is more and more essential, the infrastructure to provide wireless services is a must and the aggregation network is also used for mobile backhauling. 3GPP tries to integrate the different types of networks involved to a common network design and mobility approach, but within this solution extended network requirements will occur due to handover between base stations and it will be difficult to support distributed approaches, like content delivery networks. Forward-looking backhauling solutions have to take those aspects into account. The consequent split of the hardware-centric data plane and software-centric control plane might allow for modification or the extension of functions on devices during operation without changing the hardware, and thereby would allow software-defined networking, i.e., in the context of IEEE 802.11 compliant devices, to control or enable functional extensions by a centralized network application on a wireless edge device.

The use cases in the data center domain are focused on seamless solutions to manage the network together with the server infrastructure and on increased network and server utilization through efficient load balancing. Current data centers consist of complex network functions as well as the server infrastructure, virtual machines and storage facilities. It is extremely challenging to manage the large amount of complex entities together with the customer access to the multiple data center services. Supporting of seamless virtualization of switches, routers, servers and storage in a data center seems to be a valuable use case for OpenFlow. It allows for flow-based resource allocation inside the network and can be utilized for virtualization of network resources. Another aspect with respect to cloud applications is the overarching use of OpenFlow across network and data center domains. The network controller will communicate with the data center controller so each application shall be transmitted over a pre-defined VPN connection, and each VPN utilizing the application shall be transmitted over a sub-set of application connections. The OpenFlow controller will make sure that each cloud application has its own independent routing table on both data center domain and network domain. Moreover, networks in data centers have to be designed for peak traffic. In off-peak situations, part of the infrastructure could be put to sleep mode to save energy. This implies rerouting and concentrating processing resources. So traffic engineering and associated migration of virtual machines could be enabled by OpenFlow. To some extent, data centers could be seen as islands with a certain degree of isolation from the outside network and carrier environment. That will provide the opportunity to create islands with a new architecture without affecting the

overarching system. Therefore, the ongoing trend towards larger and more powerful data centers will be an interesting challenge in terms of scalability.

Packet/frame-based technologies play an important role in carrier-grade networks, but without flexible circuit-based transport technology no large-scale network could be operated efficiently. Therefore OTN technology is used in today's regional and backbone networks to utilize the enormous bandwidth provided by optical fibers. From the control perspective OpenFlow basically collapses several network layers, L2-L4 (ETH MAC to TCP/UDP port), into one. The attempt is to include the optical layer, but these differ in characteristics by being circuit-switched network layers. The deliverable describes the problem space, how to consolidate operation of packet and circuit switching under a common Split Architecture. Here the key is the level of detail of lower layers that should be exposed to the control plane and controllers of a Split Architecture. Two principle options are identified and discussed for many years in the context of GMPLS: The overlay and the peer model. However, the degree of information that a lower layer exposes to a higher should be left to the implementer. Most of this information will be created by controllers based on available information and alarms coming from OpenFlow-enabled hardware. This means that the control interface of the Split Architecture and OpenFlow interface will have to be extended with respect to the specification of new interface types, the handling of heterogeneous forwarding abstraction, the processing interface and the configuration of termination functions, like OAM (e.g., monitoring of links between interfaces). An important use case is the IP/MPLS backbone transport, so the different scenarios of multi layer implementation are described in the deliverable. Besides providing basic requirements, the deliverable already provides initial ideas regarding architectural options with respect to a multi layer approach and thereby kicks off the discussion that will be continued in work package 3 of the project.

In summary, this deliverable describes a set of important use cases covering the whole carrier environment and derives requirements that have to be considered in the sub-sequence studies within this project and the detailed definition of a Split Architecture. Four groups of general, important requirements have been identified. The first group covers modifications and extensions for the data path element or the Split Architecture itself. The other three groups deal with extensions of the carrier-grade operation of ICT networks. These aspects, related to the operation of an ICT network, are authentication, authorization and auto configuration; OAM in the sense of facilitating network operation and trouble-shooting; network management, security and control of the behavior of the network and protocols.

5 Annex A: Example of carrier IP backbone - topology and functionality

5.1 General topology

A typical backbone design is shown in Figure 16. It consists of a core network and the regions. The core consists of a number of LSR locations which are fully meshed by WDM connections. The regional PoPs are connected to the core PoPs via loops or direct WDM links. A loop connects two or more regional PoP and supports a redundant path from the regional PoP to the core PoP. Each location supports redundant hardware and facilities. The design of a PoP with connections to OTN/WDM systems is shown in Figure 17.

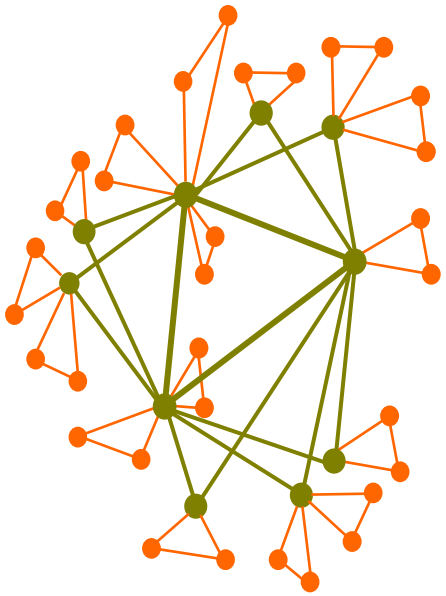

Figure 16: Typical IP backbone topology

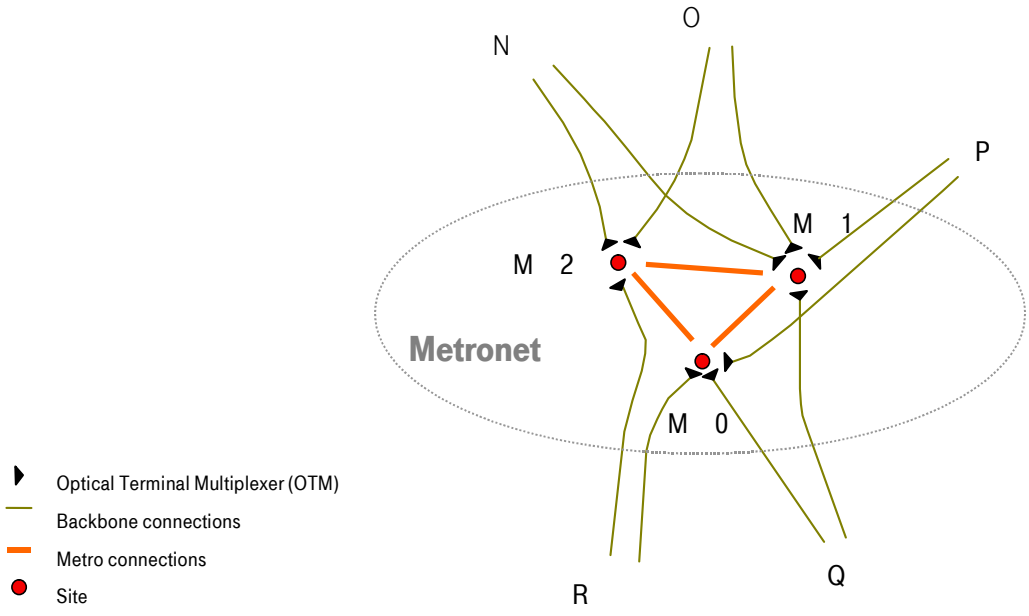

Figure 17: Core PoP details

Each PoP location uses two or three independent head-ends for the OTN/WDM links. The local links between the head-ends are realized by dark fiber or OTN/WDM.

To produce services on the platform, it has to provide a very high reliability. To meet this requirement, the platform has to be robust to down-times of the backbone routers and the backbone links. There are several approaches to meet the

requirement of high availability. A common implementation is the deployment of a backbone network with redundant nodes and links, usually called A/B networks. The A/B network structure satisfies the following requirements:

- Half-load parallel operation
- PoPs are broken down into separate security and fire section areas
- Service quality not affected by installation, expansion and maintenance work

5.2 Core network

In the core network area, an A/B network structure is to be established by connecting a core network location to the core network via two LSRs. The core network is to consist of a redundant fiber optic network featuring high availability. This is detailed in Figure 18. A connection is routed from node network A and node network B to each of the other directly connected locations (example with two adjacent locations net N-1 and N+1). The two LSRs at one location are also connected with each other. The PoP LAN is connected to the two LSRs.

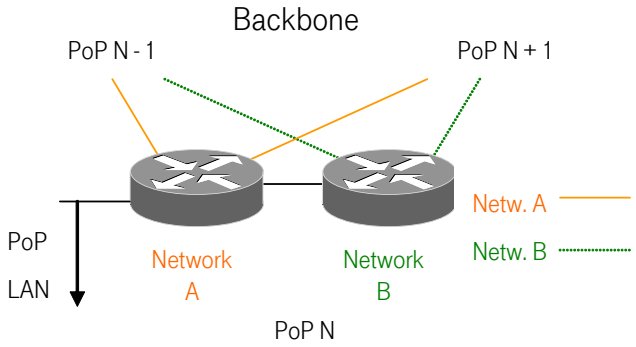

Figure 18: Connecting a core network location to the backbone

5.3 Regional network

A simple A/B network structure is to be established at the regional locations; the LSRs for connecting the regional location to the backbone are to feature redundant design. The connection to the core network features single loops, and a loop is always formed by two regional locations. Multiple Loops can share an edge; the larger bandwidth must be used by both loops. It is necessary to ensure that the loop ends are terminated at the redundant LSRs of the core network location. Figure 19 details a regional location.

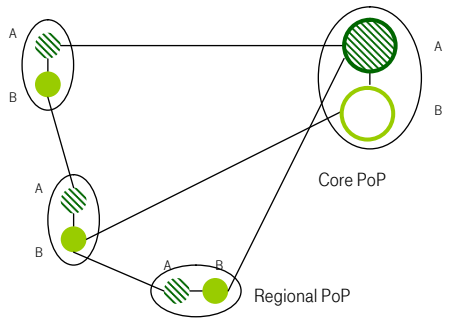

Figure 19: Connection of regional locations to a core network location

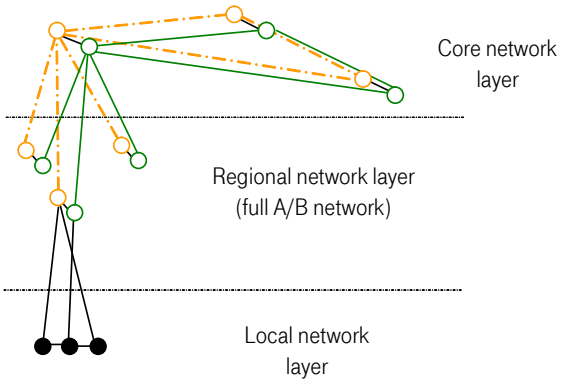

Figure 20: Possible extension of regional connection

Figure 20 illustrates the network topology for possible implementation of the second stage. This structure is prepared for the significant increase in the number of PoPs. It could be possible to move the A/B structure to these regions. In any case, the A/B structure requires the disjunct paths of the A and B network. If this cannot be guaranteed by the infrastructure, additional disjoint links are required.

All regional links are OTN/WDM Groups of trunks should be avoided to get the benefit of the best aggregation of traffic without the need of a load balancing between parallel trunks.

5.4 Label Edge Router (LER) functionalities

The LER supports all customer access for public Internet and IP-VPN. Usually there are specific Broadband Label Edge Routers (BB-LER), which serve all access lines beyond 100Mbit/s. The BB-LER has to support the same features as the LER for higher bandwidth.

5.5 MPLS

The IP backbone uses the MPLS protocol suite to

- minimize the amount of required routing information inside the backbone
- support the integrated IP-VPN
- support traffic engineering to control the paths the traffic takes

To meet the first goal, the IP platform uses only MPLS in the whole backbone. There are no IP packets from outside the platform, because the nodes don't have any knowledge about the outside routing world. This requires a label distribution protocol based on the IGP routing information or an explicit LSP mesh in the backbone. To avoid the $O(N^2)$ problem of the explicit LSP mesh, the backbone uses a label distribution protocol based on the IGP information. This label distribution protocol will be LDP as specified by the IETF [41][43][42].

Each router binds one or more IP prefixes to a LSP and distributes these bindings via LDP to its neighbors. This requires LDP running on all routers in the backbone and the edges participating in the MPLS cloud.

The MPLS path spans a full mesh between the LER. The bandwidth of the related path depends on the bandwidth generated on the related aggregation areas, on the related service creation points and on the related LER. The distribution of the bandwidth of the elements of the end to end matrix (connecting LER-LER) is far from even. Figure 21 shows the frequency count for a load of 1 Tbit/s and the Figure 22 for a load of 80 Tbit/s.

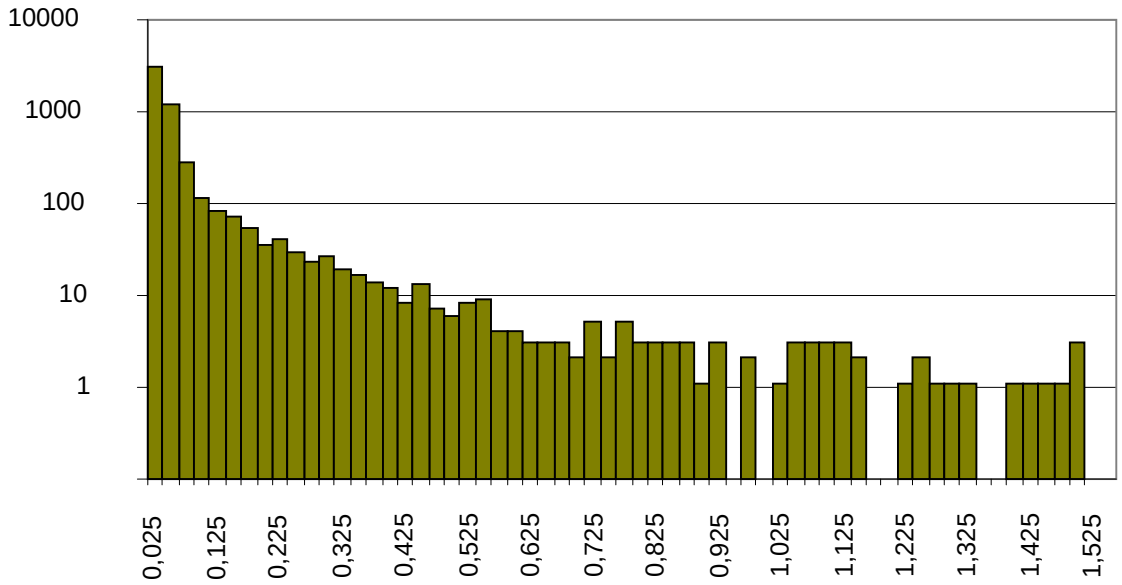

Figure 21: Count of LER-LER path bandwidth cluster (80 Tbit/s load 25 Mbit/s steps)

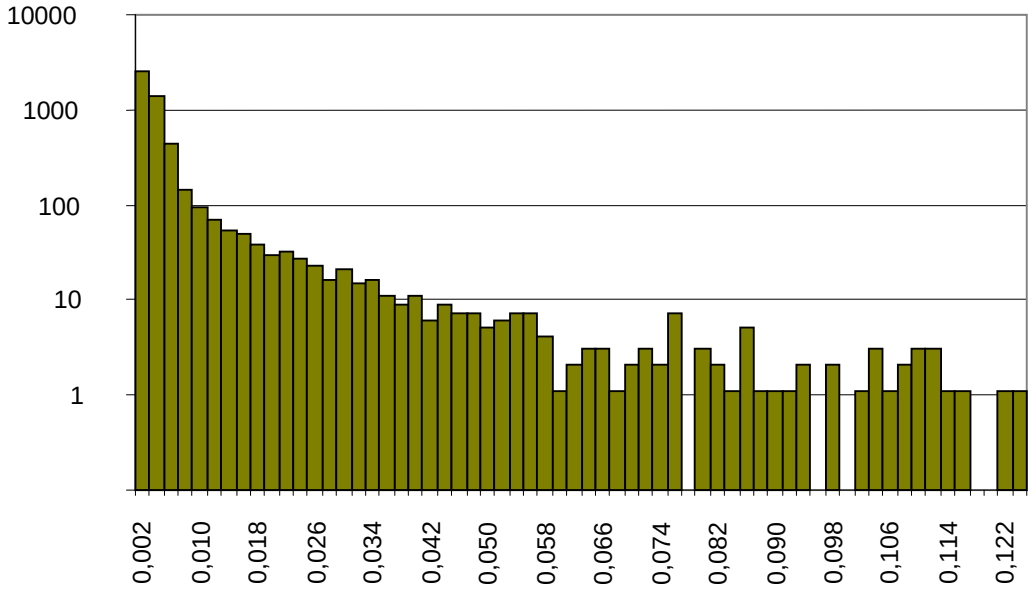

Figure 22: Count of LER-LER path bandwidth cluster (80 Tbit/s load 2 Gbit/s steps)

5.6 Exterior Gateway Protocol (EGP)

BGPv4 is used as the exterior gateway protocol. [36][37][38][39]

5.7 Internal BGP (iBGP)

To prevent the backbone from the impact of BGP updates to the forwarding of MPLS/IP packets, no BGP information should be present on the LSR routers. The full BGP information is available on all LER routers.

The BGP covers a BGP for public Internet and the BGP for IP-VPN. Route reflectors (RR) for the public Internet are available on each core PoP. There is a second RR is for redundancy reasons. All RRs are connected in a full mesh and RRs are dual homed to the LSR. All LER routers (RR client) of a core PoP area (the regional PoPs around the core PoP and the core PoP itself) are connected to the RR of the core PoP. If the number of RR clients increases and one RR pair is not sufficient to serve all RR clients, a new RR pair will be placed in the core PoP and added to the full mesh. The limit of the RR clients to a RR server will be tested in lab test.

5.8 Interior Gateway Protocol (IGP)

IS-IS is a usual IGP that is scalable for large carrier backbones [40]. The implementation of IS-IS must support traffic engineering features, therefore the IS-IS stack is based on TLV (type, length and value object).

The design of IS-IS has to consider two boundary conditions:

- Usually vendor implementations limit the amount of routing instances within a single area
- IS-IS implementation TE tunnels are bound to a single Level 2 or Level 1 routing database (LSP). That means that TE tunnels currently cannot cross the boundary from L1 to L2 routing.

The traffic engineering should be done from PoP to PoP. Each PoP builds an area and all routers within a PoP belong to this area. There are no L2-only routers in the backbone at all. Within a PoP, all LSR routers and all route reflector routers (RR) are L1-2 IS-IS routers; the LER routers are L1 IS-IS routers.

The IP addresses have to be aggregated on the area border. The loopback addresses of the routers must not be aggregated. MPLS derives its LDP ID from the highest loopback addresses, therefore these must always be 32 bit masked and reachable.

Prognosis of routing table size: Scalability requirements

Figure 23 shows the prognosis and the history of routing table size for the different domains and applications, BGP, customer and IPv6.

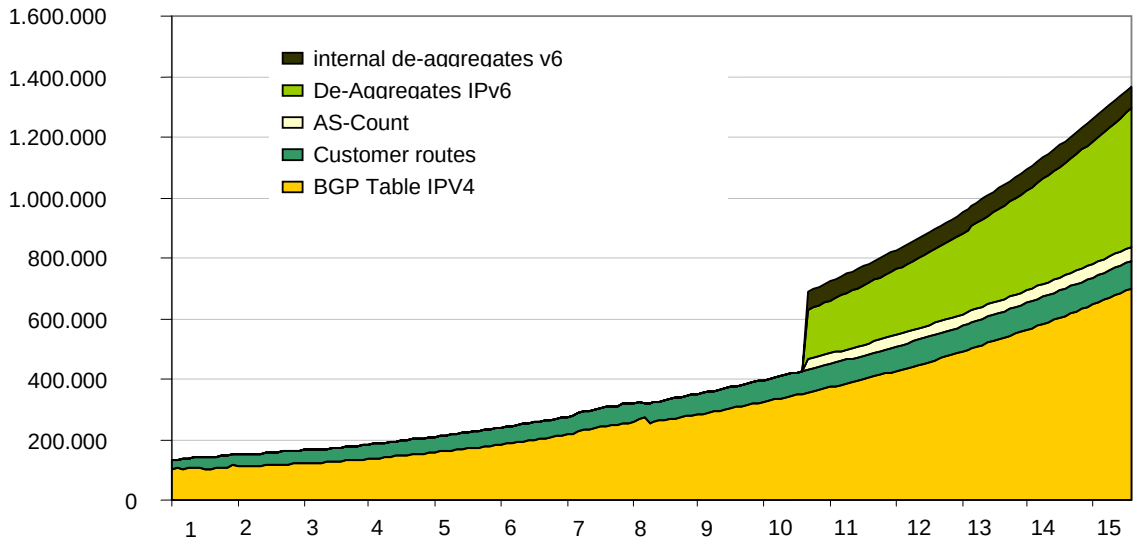

Figure 23: Growth of routing table size and routing table requirements

5.9 Traffic engineering

5.9.1 General

Internet traffic engineering is defined as the aspect of network engineering that addresses the issue of performance evaluation and performance optimization of operational IP networks. It encompasses the application of technology and scientific principles to the measurement, modeling, characterization and control of Internet traffic. A major goal of Internet Traffic Engineering is to facilitate efficient and reliable network operations while simultaneously optimizing network resource utilization and traffic performance. In this sense, the main objective of Traffic Engineering is the process of arranging how traffic flows through the network so that congestion caused by uneven network utilization can be avoided.

Effective traffic engineering has been difficult in conventional IP networks. The reasons are limited functional capabilities, e.g., (1) inadequacy of measurement functions and (2) limitations of intra-domain routing control functions. Sufficient measurement functions are indispensible in the generation of traffic matrixes, which are basic data sets needed to deploy traffic engineering. Standard IP routing is typically based on the destination address and simple metrics such as hop count or delay. Each router makes independent routing decisions using a local instantiation of a synchronized routing-area link state database. Route selection is based on shortest path computations using simple additive link metrics. The flaw is that these protocols do not consider the characteristics of offered traffic and network

capacity constraints when making routing decisions. This results in subsets of network resources becoming congested, while other resources along alternate paths remain under-utilized.

Traffic engineering is the ability to move trunks away from the path selected by the standard Interior Gateway Protocols (e.g., OSPF or IS-IS) and onto a different path. This allows an ISP to route traffic around known points of congestion in its network, thereby making more efficient use of the available bandwidth. In turn, this makes the ISP more competitive within its market by allowing the ISP to pass lower costs and better service on to its customers.

Multiprotocol Label Switching (MPLS) technology opens new possibilities to overcome the limitations of traditional IP systems concerning traffic engineering. This is because MPLS efficiently supports originating connection control through explicit Label Switched Paths (explicit LSPs). An explicit LSP is one whose route is determined at the origination node. Origination connection control permits explicit routes to be established which are independent of the destination-based IP shortest path routing model. Through explicit LSPs, MPLS permits a quasi circuit switching capability to be superimposed on the current Internet routing model. The method for determining an explicit LSP at the originating node can range from static provisioning, provisioning by a management system or an automatic routing process called Constraint-Based Routing.

5.9.2 Traffic Engineering Process Model

A number of stages can be identified in a generic Traffic Engineering process model:

1. The first stage is the formulation of a control policy. This policy may depend on economical factors, like the network cost structure, the operating constraints and some optimization factors.
2. The second stage is the observation of the network state through a set of monitoring functions. This is the feedback process which involves acquiring measurement data from the operational network.
3. The third stage is the characterization and the analysis of the network state. The results are used for network performance optimization, network operations control, network design and capacity planning.
4. The fourth stage is the optimization of network performance. The performance optimization phase generally involves a decision process which selects and implements a particular set of actions. This can range from short term to long term actions, like controlling the distribution of traffic across the network, increasing link capacity, deploying additional hardware or starting a network planning process to improve the network architecture and the network design.

Traffic engineering is an adaptive process. The four stages of the process model are repeated iteratively. It is obvious that a goal must be to minimize the level of manual intervention in traffic engineering by automating the tasks whenever possible [44].

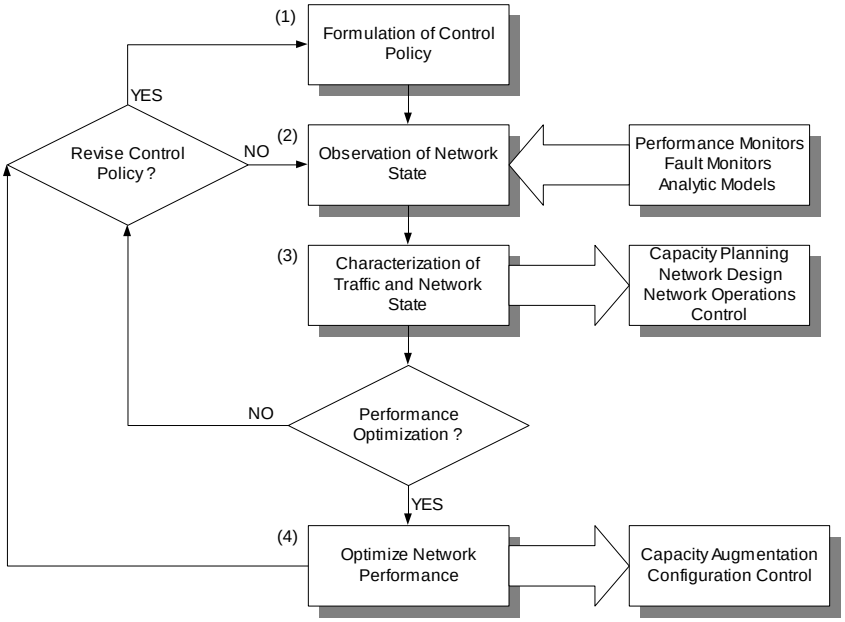

Figure 24: Traffic engineering process model

5.9.3 Constraint-based routing

In order to minimize the amount of administrative explicit path configuration and manual intervention, an automatic process is needed to find a feasible mapping of traffic trunks onto the physical network. This process is called

Constraint-Based Routing. Constraint-Based Routing enables a demand driven, resource reservation aware, routing paradigm to co-exist with current topology-driven hop-by-hop interior gateway protocols. Using the input data (traffic trunk attributes, resource attributes and topology state information) a process on each node automatically computes explicit LSP for each traffic trunk originated from the node. In practice, the network operator, will specify the endpoints of a traffic trunk and assign a set of attributes to the trunk which encapsulates the performance expectations and behavioral characteristics of the trunk. Constraint-Based Routing is then expected to find a feasible path to satisfy the expectations. If necessary, the network operator (with or without the aid of a traffic engineering support system) can the use administratively configured explicit LSPs to perform fine grained optimization.

As Constraint-Based Routing relies on the resource information, this additional information has to be flooded within the routing domain. There are extensions available to link-state IGPs (OSPF and IS-IS), which are standardized by the IETF.

5.9.4 Path setup

Once an explicit LSP is determined, either by administratively action or computed by a Constraint-Based Routing process, a signaling system is needed to establish the LSP (label distribution) and to make the appropriate resource reservation along the path. There are two different approaches in the standardization by the IETF, extended RSVP and CR-LDP.

5.9.5 Path maintenance

In the context of MPLS based Traffic Engineering the term Path Maintenance refers the following capabilities:

- Path Re-Optimization
- Path Restoration

In general, these capabilities deal with and should be part of a network survivability concept. Failure protection and restoration capabilities have become available from multiple layers, like SONET/SDH with self-healing ring topologies or APS and at the IP layer with its rerouting capability provided by standard dynamic routing protocols. In order to support advanced survivability requirements, MPLS as a path-oriented technology can be used to enhance to survivability of IP networks.

5.9.5.1 Path re-optimization

Due to the dynamic behavior, the network would become progressively sub-optimal every time a failure occurs. For this reason, periodical Path Re-Optimization is needed. There are two ways Path Re-Optimization takes place: via manual intervention, or via automated procedures performed by the head-end routers. In the first case, the network operator must periodically update the traffic model stored in the head-end routers, while in the second case, the head-end routers will periodically recalculate the paths for traffic trunks. The rerouting of traffic trunks as the result of re-optimization takes place in a seamless fashion. The head-end router first establishes the new LSP for the traffic trunk and then switches the traffic trunk onto the new LSP ("make-before-break"). An explicit trunk attribute, the "adaptability" attribute indicates whether a traffic trunk can be rerouted for the reason of re-optimization.

5.9.5.2 Path restoration

Path protection: The path of the protection LSP is completely disjointed from its working LSP. Path protection is controlled at the head-end router of the traffic trunk. When the head-end router detects that the LSP carrying the traffic trunk has failed, it performs the action specified by the "resilience" attribute of the trunk. The possible options are:

- None – No explicit path protection. In failure scenario, the packets will be forwarded based on standard IGP
- Fallback to pre-computed and pre-established backup-LSP. This provides the fastest path restoration at the expense of wasting bandwidth reserved by the backup-LSP.
- Establish a new LSP based on a pre-computed path and move the trunk onto the new LSP. The price for the bandwidth efficiency over the previous option is the additional delay incurred for establishing the new LSP.
- Compute and establish a new path, and move the trunk onto the new path. This option provides the slowest restoration but incurs no bandwidth or processing overhead.

Link protection: The path of a protection LSP is only disjointed from its working LSP at a particular link. Traffic on the working LSP is switched over to a protection path at the upstream LSR that connects to the failed link. This method is potentially the fastest. There are the same options to establish the protection path as described above.

5.10 Quality of Service / Class of Service (QoS/CoS)

5.10.1 General

Quality of Service (QoS) is supported by backbone networks. The following chapters describe the technical implementation and features which form the basis for building a set of new innovative products on top of it. It does not mean that each product has to exploit all provided features. Instead, keep it as simple as possible to earn the most efficient benefit and to create very fast innovative new products.

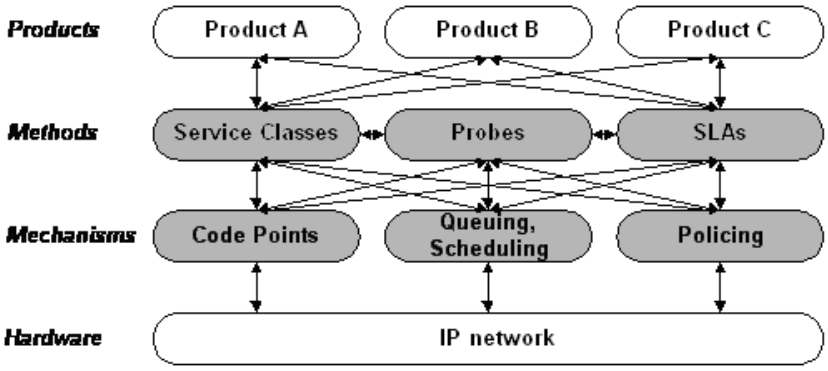

Figure 25: QoS support for new IP products

Scalability and stability are the main criteria for any extension of the network. It is absolutely necessary to aggregate IP streams with identical flow characteristic. The expression used for this solution is "service classes". Dedicated handling of single streams is only meaningful in special cases when high bandwidths are involved, and there are no plans for this solution to be introduced in the first instance.

The number of service classes should be strictly limited from the technical point of view. This is no restriction to the construction of various commercial products on top of it. Service level agreements (SLA) form the definition interface for the service which will be delivered to the customer by the provider. Parameters should describe a probability for a certain service and will be reported on as per class base.

Maximum 8 code points per path can be easily supported with respect to the intended MPLS solution. These are distinguished using the three experimental bits of the MPLS shim header. A large part of best-effort background traffic is required to produce efficient high quality service classes because DiffServ is based on relative priorities.

5.10.2 Service description

5.10.2.1 Requirements

QoS classes are in line with definitions given by the Metro Ethernet Forum (see also previous chapters on access/aggregation).

As required a basic way to provide QoS is the definition of service classes. Beside a standard class, which is characterized as the best-effort class, three premium classes will be introduced. They focus on different services. Upcoming trends show that there will be an additional need for a class which is specially designed to transport voice traffic (VoIP). A customer can describe to one or more service classes.

The following fundamental service requirements have to be fulfilled by the backbone:

5.10.2.2 Service Class - Service Description

Service classes differ from each other by their characterization and handling. The following are examples of technical quality parameters that have been identified:

- Throughput (bandwidth) between Service Creation Points (SCP1, 3)
- One-way transmission delay between Service Creation Points (SCP1, 3)
- One-way transmission delay variation (jitter) between Service Creation Points (SCP1, 3)
- Packet loss rate between Service Creation Points (SCP1, 3)

Beside this, administrative aspects are of importance, like:

- Availability

- SLA monitoring
- SLA reporting details
- Recovery periods
- Customer care

Service Class: Standard

The service class "Standard" is a best-effort service. There are no throughput or delay guarantees.

Service Class: Premium I

The service class "Premium I" supports mission critical services, typically based upon TCP. The QoS requirements within this class are prioritized in the following sequence:

- There are minimal packet losses up to the agreed throughput rate between two CPE routers
- The nominal throughput rate can be determined for each site (hose model)
- The maximum end-to-end packet delay can be estimated (delay model)

Service Class Premium II

The service class "Premium II" supports "multi-media" services, based upon TCP or UDP. The QoS requirements within this class are prioritized in the following sequence:

- The probable maximum end-to-end packet delay and packet jitter are improved compared to Premium I
- The nominal throughput rate can be determined for each site (hose model)
- The loss probability is higher than for Premium I

Service Class: PremiumVoice

The service class "PremiumVoice" supports "real-time" services, typically based UDP/RTP. The QoS requirements within this class are prioritized in the following sequence:

- This traffic is preferred handled within the backbone
- The probable maximum end-to-end packet delay and packet jitter is optimized
- The nominal throughput rate can be determined for each site (hose model)

5.11 Summary of minimal technical requirements for IP backbone devices

Usage Cluster		**LER**	**LSR**
Components			
Chassis		X	X
Memory		X	X
Switching fabric		X	X
Interfaces incl.		X	X
Service Cards		X	
HW Parameters			
Chassis	Type	Single	Single/(Multi)
	# of chassis	1	(up to 72 LC /8 FC)
	# of routing engine slots	1(2)	2
	# of switching fabric slots	3	8
Memory	Type	SDRAM	

	Max. Size	8Gb (PRP-3)	
Switching fabric	Type	Backplane	Midplane
	PPS	320Gb/s (1,2Tb/s)	1.2Tb/s
Line Card Slots	PPS	2.5G/10G/40G (12816)	up to 40Gb/s
	#	15(14)	16
Service Cards	Type	SBC, vFirewall, etc.	
	#		
	# of Intf / SC		
Interfaces	Type	GE, POS, etc.	GE,POS, WDM
	Memory	1GB Route/ 512 Packet	
Processor/CPU	#	1(2)	2
	Performance		
	Memory	4GB	8GB
Control Plane Protocols			
BGP	labeled BGP	Yes	Yes
	MP-BGP	Yes	Yes
LDP		Yes	Yes
P2MP LDP		Yes	Yes
IS-IS			
IS-IS TE		Yes	Yes
RSVP-TE		Yes	Yes
OSPF		Yes	Yes
PIM-xSM		Yes	Yes
IGMP / MLD		Yes	Yes
BFD		Yes	Yes
LMP		Yes	Yes
Data Plane Protocols			
IPv4	Support	Yes	Yes
	Forwarding performance	Line rate	Line rate
IPv6	Support	Yes	Yes
	Forwarding performance	Line rate	Line rate
Mix v4/v6	Support	Yes	Yes
	Forwarding performance	Line rate	Line rate
Multicast IPv4	Support	Yes	Yes
Multicast IPv6	Support	Yes	Yes
MPLS Support	Support	Yes	Yes

Other Service Functionalities			
QoS	HW Support	Yes	Yes
	DiffServ / Model	Yes	Yes
	# of queues	2,000 ingress queues, 8,000 egress queues per LC	8,000 per LC
VPN	L2 VPN Support	Yes (depends on LC)	Yes
	L3VPN Support	Yes (depends on LC)	Yes

6 Annex B: List of requirements and explanations

In order to give an overview, all requirements derived from the three different use case areas are summarized in the following Table 4. The definition of the use cases results in a mix of general and use case specific requirements. However, not all requirements are meaningfull and specific with respect to a Split Architecture or different implementations of OpenFlow. Therefore an evaluation of all requirements with respect to the potential impact on a Split Architecture and/or OpenFlow extensions is given. A high degree of impact is indicated by "**", some potential impact is indicate by "*".
So the most important requirements to be considered in the future discussion about the blue-print of a Split Architecture are those indicated with one or more "*".
In the defininition of the requirements the key words "must", "required", "shall", "should"and may" are to be interpreted as described in IETF RFC 2119. An additional column gives an indication about the respective "level of significance" of each requirement. To idicate the levels, the terms must, recommended and optional are used.

Definition given by IETF RFC 2119:

- MUST - This word, or the terms "REQUIRED" or "SHALL", mean that the definition is an absolute requirement of the specification.
- SHOULD - This word, or the adjective "RECOMMENDED", mean that there may exist valid reasons in particular circumstances to ignore a particular item, but the full implications must be understood and carefully weighed before choosing a different course.
- MAY - This word, or the adjective "OPTIONAL", mean that an item is truly optional. One vendor may choose to include the item because a particular marketplace requires it or because the vendor feels that it enhances the product while another vendor may omit the same item.

An implementation which does not include a particular option MUST be prepared to interoperate with another implementation which does include the option, though perhaps with reduced functionality. In the same vein an implementation which does include a particular option MUST be prepared to interoperate with another implementation which does not include the option (except, of course, for the feature the option provides.)

Table 4: List of requirements and analysis

No.	Requirement	Importance of Requirement	Missing in OpenFlow /SplitArchi.	Group of Requirement
R-1	A wide variety of services / service bundles should be supported.	**	-	
R-2	The Split Architecture should support multiple providers.	**	-	
R-3	The Split Architecture should allow sharing of a common infrastructure, to enable multi-service or multi-provider operation.	**	-	
R-4	The Split Architecture should avoid interdependencies of administrative domains in a multi-provider scenario.		**	
R-5	The Split Architecture should support operation of different protocols / protocol variations at the same OSI layer in case of shared networks.	**	**	A
R-6	The Split Architecture should support policy based network control of network characteristics.	**	**	B, C
R-7	The Split Architecture shall enforce requested policies.	**	**	B, C
R-8	The Split Architecture should support automatic transfer methods for distribution of customer profiles and policies in network devices.		**	
R-9	The Split Architecture should support TDM emulation and/or mobile backhaul.	*	**	
R-10	The Split Architecture should provide sufficient customer identification.	**	**	B

R-11	The Split Architecture should support best practices for QoS with differentiated, four classes according to the definition documented by the Metro Ethernet Forum.	**	*	
R-12	The Split Architecture should handle data not carrying QoS class identifier as default class.	**	**	D
R-13	The Split Architecture should map data carrying invalid QoS class identifier to a valid QoS class.	**	**	D
R-14	The Split Architecture must handle control data as highest priority class.	**		
R-15	The Split Architecture should support Source-Specific Multicast.			
R-16	The Split Architecture must control the access to the network and specific services on an individual service provider basis.	*	**	
R-17	The Split Architecture should provide mechanisms to control broadcast domains.			
R-18	The Split Architecture should support enforcement of traffic directions.			
R-19	The Split Architecture should support control mechanisms for identifier. This should include any kind of identifiers like addresses or protocols as well as limitation for send rate.		**	
R-20	The Split Architecture should prevent any kind of spoofing.	**		
R-21	The Split Architecture should monitor information required for management purposes.	**	**	D
R-22	The Split Architecture should generate traps when rules are violated.	**	**	
R-23	The Split Architecture should extract accounting information.	**	**	B
R-24	The Split Architecture should collect traffic statistics.	**	-	
R-25	The Split Architecture should support OAM mechanisms according to the applied data plane technologies.	**	**	C
R-26	The Split Architecture should make use of OAM functions provided by the interface.	**	**	C
R-27	The Split Architecture shall support the monitoring of links between interfaces.	**	**	C
R-28	The data path element should provide logically separated access to its internal forwarding and processing logic in order to control both independently.	**	**	A
R-29	It should be possible to define chains of processing functions to implement complex processing.	**	**	A
R-30	The Split Architecture shall support deployment of legacy and future protocol/service-aware processing function.	**	**	A
R-31	The introduction of a new protocol/service aware processing function should not necessitate the update of other functions.	**	**	A
R-32	The architecture of a data path element shall support loading of processing functions at run-time without service interruption.	**	**	A
R-33	A processing function instance should be controllable at run-time by the associated control entity in the control plane.	**	**	A
R-34	A processing function should expose module	**	**	A

	specific configuration parameters to an associated entity in the control plane.			
R-35	The Split Architecture should allow the exchange of opaque control information between a processing function on the data path element and the associated control entity with a well-defined protocol.	**	**	A
R-36	The level of detail exposed by a processing module is module and vendor specific. However, each processing module should support a common API for control purposes.	**	**	A
R-37	Urgent notifications sent by a data path element should be prioritized and not be delayed by data traffic to the controller.	**	**	A
R-38	A data path element classifier should be constructed in a protocol agnostic manner or should be at least flexible enough to load new classifier functionality as a firmware upgrade with identical performance.	**	**	A
R-39	The Split Architecture should introduce a clean split between processing and forwarding functionality.	**	**	A
R-40	The Split Architecture should provide means to control processing functions from a controlling entity on heterogeneous hardware platforms.	**	**	A
R-41	Providers should have a maximum degree of freedom in the choice of data centre technologies.	-	**	
R-42	Data centre virtualization must ensure high availability.	*	**	
R-43	The Private Cloud provisioning and management system shall have the ability to dedicate a specific share of resource per VPN.	**	-	
R-44	Each VPN may have the exclusive access to the specific share of resource.	-	-	
R-45	Each VPN shall have the ability to hold the requested resources without sharing with any other parties	**	-	
R-46	Each VPN may have the ability to limit the stored data mobility to a certain geographic region confinement (country/state).	-	**	
R-47	The restoration capability awareness should to be scalable.	-	-	
R-48	The virtualization functions QoS requirement should be synchronized with VPN service.	-	-	
R-49	The VPN extension should support the network condition to be used for the traffic balancing and congestion avoidance decision-making.	-	*	
R-50	The VPN resource requested by the server can be optimized by statistical multiplexing of the resource.	-	*	
R-51	The VPN Extension should support the automatic end-to-end network configuration.	-	*	
R-52	Quality of Experience management should to be supported.	-	*	
R-53	The data path element should expose to the load balancer/network controller the information regarding their availability, connections to other elements and load situations.	**	-	
R-54	The data path element should provide an API exposing mechanism that can be used to configure for switching/routing flows of packets.	**	-	

R-55	The Server/VM manager should expose to the load balancer/network controller the information regarding the operation of VMs including availability, load situation and the association to the servers.	**	-	
R-56	The Server/VM manager should provide an API exposing mechanisms that can be used to control the instantiation and migration of VMs across the server farm.	**	-	
R-57	The load-balancing solution should support L2-L7 flow detection/classification.	**	*	
R-58	The load-balancing solution should provide session persistence.	**	*	
R-59	The data path element should provide an API exposing mechanisms for switching the data path element between sleep/normal operation modes.	**	*	
R-60	The data path element should expose metrics that can be used by energy optimization algorithms.	**	**	C, D
R-61	The network management and configuration must provide predictable and consistent capabilities.	**	-	
R-62	The network management and configuration should provide a cost vs. benefit ratio better than today's approaches.	-	-	
R-63	The OF domain should be able to interact with other domains through EGP.	*	**	
R-64	Information of a lower layer has to be exposed to a higher layer appropriately.	**	**	A
R-65	A data path network element should enable control plane applications to poll detailed configuration attributes of circuit-switching capable interfaces.	**	**	A
R-66	A data path element should enable control plane application to set configuration attributes of circuit-switching capable interfaces.	**	**	A
R-67	The Split architecture may allow implementing additional OAM solution when the interface does not provide any.	*	**	

Abbreviations

3GPP	Third generation partnership program
ADSL	Asymmetric Digital Subscriber Line
AES	Advanced Encryption Standard
AGS	Aggregation Switch
ANDSF	Access Network Discovery Selection Function
AP	Access Point
API	Application Programming Interface
ARP	Address Resolution Protocol
AS	Autonomous System
ATM	Asynchronous Transfer Mode
AWG	Arrayed-waveguide Grating
BB	Broadband
BBA	Broadband Access
BBF	Broadband Forum
BB-RAR	Broadband Remote-Access-Router (SCP for Fast Internet)
BE	Best-Effort
BFD	Bidirectional Forwarding Detection
BG	Broadband Aggregation Gateway
BGP	Border Gateway Protocol; Distance Vector Routingprotocol der IETF (EGP)
BRAS	Broadband Remote Access Server / Service
BSS	Basic Service Set
CAR	Committed Access Rate
CBO	Class Based Queuing
CBWFQ	Class-Based Weighted Fair Queuing
CDMA	Code Division Multiple Access
CE	Control Element
CHG	Customer HG; HG in customer site
CIDR	Classless Inter-Domain Routing
CIPM	Cisco IP Manager
CIR	Committed Information Rate
CLI	Command Line Interface
CLNC	Customer LNS, LNS in customer site
CORBA	Common Object Request Broker Architecture
CoS	Class of Service
CP	Connectivity Provider
CPE	Customer Premise Equipment
CPU	Central Processing Unit
CRC	Cyclic Redundancy Check
CR-LDP	Constraint based LDP
DCF	Distributed Coordination Function
DHCP	Dynamic Host Configuration Protocol
DHT	Distributed Hash Table
DiffServ	Differentiated Services, IETF
DNS	Domain-Name-Server
DOCSIS	Data Over Cable Service Interface Specification
DPI	Deep Packet Inspection
DRR	Deficit Round Robin
DS	Differentiated Services
DSCP	Diff Serve Code Point
DSL	Digital Subscriber Line
DSLAM	Digital Subscriber Line Access Multiplexer (network side of ADSL line)
DWDM	(Dense) Wave-Division-Multiplex
dWRED	Distributed WRED
DXC	Digital Cross-Connect
ECMP	Equal Cost Multi Path
ECN	Explicit Congestion Notification
ECR	Egress Committed Rate
EGP	Exterior Gateway Protocol
EIGRP	Enhanced IGRP
ePDG	Evolved Packet Data Network Gateway
ESS	Extended Service Set
FE	Forwarding Element
FEC	Forwarding Equivalence Class
FEC	Forward Error Correction
FIB	Forwarding Information Base
FMC	Fixed Mobile Convergence
ForCES	Forwarding and Control Element Separation
FPGA	Field Programmable Gate Array
FSC	Fiber Switching
FTTCab	Fibre to the Cabinet
FTTH	Fibre to the Home
FTTLEx	Fibre to the Local Exchange
FW	Firewall
GbE	Gigabit Ethernet

GFP	Generic Framing Procedure
GLONASS	Globalnaja Nawigazionnaja Sputnikowaja Sistema (Russian satellite system)
GMPLS	Generalized Multiprotocol Label Switching
GNSS	Global Navigation Satellite System
GPON	Gigabit Passive Optical Network
GPS	Global Positioning System
GRE	Generic Route Encapsulation
GTS	Generic Traffic Shaping
GUI	Graphical User Interface
HCF	Hybrid Coordination Function
HDLC	High-level Data Link Control
HG	Home Gateway
IACD	Interface Adjustment Capability Descriptor
ICR	Ingress Committed Rate
ICT	Information and communication technology
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force (www.ietf.org)
IF	Interface
IGMP	Internet Group Management Protocol
IGP	Interior Gateway Protocol
IGRP	Interior Gateway Routing Protocol
IntServ	Integrated Services, IETF
IP	Internet Protocol
IPTV	IP television
ISCD	Interface Switching Capability Descriptor
ISDN	Integrated Services Digital Network
IS-IS	Intermediate System - Intermediate System; Link State Routingprotocol from OSI (IGP)
ISO	International Organization for Standardization
ISP	Internet Service Provider
ITIL	IT infrastructure library
ITU	International Telecommunication Union
L2F	Layer 2 Forwarding
L2TP	Layer 2 Tunnel Protocol
LAC	L2TP Access Concentrator
LAN	Local Area Network
LDP	Label Distribution Protocol
LER	Label Edge Router; MPLS-based router with MPLS, IP-VPN and QoS edge support
LER-BB	Broadband LER; LER for DS and higher
L-GW	Local gateway
L-LSP	Label inferred LSP
LMP	Link Management Protocol
LNS	L2TP Network Server
LSP	Label Switch Path
LSR	Label Switch Router; MPLS-based Router in the inner IP network. Only IGP knowledge.
LTE	Long Term Evolution
MAC	Media Access Control
MAN	Metropolitan Area Network
MEF	Metro Ethernet Forum
MGW	Media Gateway
MIB	Management Information Base
MLD	Multicast Listener Discovery
MME	Mobility Management Entity
MPLS	Multi Protocol Label Switching
MPLS-TP	MPLS Transport Profile
MSC	Mobile Switch controller
MTU	Maximum Transmission Unit
NGN	Next Generation Network
NIC	Network Interface Controller
NMS	Network Management System
NP	Network Provider
NTP	Network Time Protocol
OAM	Operation, Administration and Maintenance
ODU	Optical Data Unit
OF	Open Flow
OFDM	Orthogonal Frequency Division Multiplexing
OLT	Optical Line Termination
OSI	Open Systems Interconnection
OSNR	Optical Signal-to-Noise Ratio
OSPF	Open Shortest Path First; Link State Routingprotocol from IETF (IGP)
OTN	Optical Transport Network
OTU	Optical Transport Unit
OXC	Optical Cross-Connect
PANA	Protocol for carrying Authentication for Network Access
PBB-TE	Provider Backbone Bridge Traffic Engineering
PDU	Procotol Data Unit

PE	Provider Edge; Service Creation Point for IP-VPN
PER	Provider Edge Router
PGW	Packet Data Network Gateway
PIM	Protocol Independent Multicast
PIP	Physical Infrastructure Provider
PMIP	Proxy Mobile IP
PoP	Point of Presence
POTS	Plain Old Telephony Service
PPP	Point-to-Point Protocol
PPPoE	PPP over Ethernet
PSTN	Public Switched Telephone Network
PVC	Permanent Virtual Circuit (permanent L2 connection e.g., Frame Relay, ATM)
QoE	Quality of Experience
QoS	Quality-of-Service; general for differentiated quality of services or absolute quality of services.
QPSK	Quadrature phase-shift keying
RADIUS	Remote Authentication Dial-In User Service
RAR	Remote-Access-Router (SCP für OCN)
RARP	Reverse ARP
RFC	Request for Comment (in IETF)
RGW	Residential Gateway
RIB	Routing Information Bases
RIP	Routing Information Protocol; Distance Vector Routingprotocol from IETF (EGP)
ROADM	Reconfigurable Aptical Add-Drop Multiplexer
RR	Route Reflector for BGP/MP-BGP
RTP	Real Time Protocol
RTT	Round Trip Time
SAE	System Architecture Evolution
SBC	Session Border Controller
SCP	Service Creation Point
SDH	Synchronous Digital Hierarchy
SDN	Software Defined Networking
SGW	Serving Gateway
SHG	Separate HG, separate HG device with virtual HG
SIPTO	Selective IP Traffic Offload
SLA	Service Level Agreement
SLNS	Separate LNS, separate LNS device with virtual LNS
SMS	Service Management System
SNMP	Simple Network Management Protocol
SONET	Sznchronous Optical Network
SP	Service Provider
SP	Service Provider
SPARC	Split architecture for carrier grade networks
SSID	Service Set Identifier
SSM	Source Specific Multicast
STA	Station
STM	Synchronous Transfer Modul (STM-1: 155 Mbit/s, STM-4: 622 Mbit/s, STM-16: 2.5 Gbit/s; STM-64: 10 Gbit/s)
TCAM	Ternary Content Addressable Memory
TCM	Tandem Connection Monitoring
TCP	Transmission Control Protocol
TDM	Time Division Multiplexing
TE	Traffic Engineering
TKIP	Temporal Key Integrity Protocol
ToR	Top of the Rack
ToS	Type of Service
TR	Technical Report (from BBF)
TTL	Time to live
UDP	User Datagram Protocol
UNI	User Network Interface
VEB	Virtual Ethernet Bridges
VEPA	Virtual Ethernet Port Aggregator
VLAN	Virtual LAN
VM	Virtual Machine
vNIC	Virtual NIC
VoIP	Voice over IP
VPN	Virtual Private Network
VSI	Virtual Station Interface
WAN	Wide Area Network
WDM	Wavelength Division Multiplexing
WEP	Wired Equivalent Privacy
WFQ	Weighted Fair Queuing
WP	Work package
WT	Working Text (from BBF)

References

[1] Broadband Forum Technical Report TR-101 "Migration to Ethernet-Based DSL Aggregation", April 2006, available online. http://www.broadband-forum.org/technical/download/TR-101.pdf

[2] Seamless MPLS (draft-leymann-mpls-seamless-mpls-02), available online http://www.ietf.org/id/draft-leymann-mpls-seamless-mpls-02.txt

[3] IEEE 802.11 - IEEE Standard for Information technology – Telecommunications and information exchange between systems – Local and metropolitan area networks – Specific requirements / Part 11: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications, available online: http://standards.ieee.org/getieee802/802.11.html

[4] IEEE 802.11k - IEEE Standard for Information technology – Telecommunications and information exchange between systems – Local and metropolitan area networks – Specific requirements / Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications / Amendment 1: Radio Resource Measurement of Wireless LANs, available online: http://standards.ieee.org/getieee802/802.11.html

[5] IEEE 802.11k - IEEE Standard for Information technology – Telecommunications and information exchange between systems – Local and metropolitan area networks – Specific requirements / Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications / Amendment 2: Fast Basic Service Set (BSS) Transition, available online: http://standards.ieee.org/getieee802/802.11.html

[6] 3GPP-23.402v10.1.0 - 3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Architecture enhancements for non-3GPP accesses (Release 10), http://www.3gpp.org/ftp/Specs/html-info/23402.htm

[7] Broadband Forum TR-069, CPE WAN Management Protocol v1.1, Release 3.0, Amendment 2, Available online: http://www.broadband-forum.org/technical/download/TR-069_Amendment-2.pdf

[8] Broadband Forum TR-181, Device Data Model for TR-069, Issue 1+2, Available online: http://www.broadband-forum.org/technical/download/TR-181_Issue-2.pdf

[9] DSL Forum, Technical Report TR-144, "Broadband Multi-Service Architecture & Framework", Available online: http://www.broadband-forum.org/technical/download/TR-144.pdf

[10] Broadband Forum, Working Text WT-145, "Multi-service Broadband Network Functional Modules and Architecture", Revision 16, August 2010

[11] DSL Forum, Technical Report TR-092, "Broadband Remote Access Server (BRAS) Requirements Document", August 2004, Available online: http://www.broadband-forum.org/technical/download/TR-092.pdf

[12] DSL Forum, Technical Report TR-059, "DSL Evolution – Architecture Requirements for the Support of QoS-Enabled IP Services", September 2003, Available online: http://www.broadband-forum.org/technical/download/TR-059.pdf

[13] Broadband Forum Working Text WT-178, "Multi-service Broadband Network Architecture and Nodal Requirements", Revision 00, September 2010

[14] Broadband Forum Working Text WT-203, "Interworking between Next Generation Fixed and 3GPP Wireless Access", Revision 06, October 2010

[15] Broadband Forum Working Text WT-221, "MPLS based mobile network for LTE", Revision 06, July 2010

[16] Broadband Forum Working Text WT-223, "Technical Specifications for MPLS in Mobile Backhaul Networks", Revision 04, June 2010

[17] IST OASE (FP7 – ICT– GA 249025), "Deliverable D6.1: Overview of Tools and Methods and Identification of Value Networks", 29.10.2010

[18] Deutsche Telekom, "T-Mobile UK and 3 create Britain's largest 3G network", http://www.deutschetelekom.com/dtag/cms/content/dt/en/529162;jsessionid=C08B5A0EF472D3F94451019F0ADD2357?printversion=true, 18. December 2007

[19] Arnaud Cauvin et al., "Next Generation Mobile Networks Optimised Backhaul Requirements", NGMN alliance, August 14th 2008

[20] 3GPP-23.829v1.2.0 - 3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Local IP Access and Selected IP Traffic Offload (Release 10), available online: http://www.3gpp.org/ftp/Specs/html-info/23829.htm

[21] 3GPP-24.312v10.0.0 - 3rd Generation Partnership Project; Technical Specification Group Core Network and Terminals; Access Network Discovery and Selection Function (ANDSF) Management Object (MO) (Release 10), available online: http://www.3gpp.org/ftp/Specs/html-info/24312.htm

[22] Ripcord: A Modular Platform for Data Centre Networking, Technical Report No. UCB/EECS-2010-93, available online: http://www.eecs.berkeley.edu/Pubs/TechRpts/2010/EECS-2010-93.html, June 7, 2010

[23] IEEE 802.1Qbg – Edge Virtual Bridging, Draft 1.2, available online: http://www.ieee802.org/1/pages/802.1bg.html

[24] IEEE 802.1Qbh – Bridge Port Extension, Draft 0.5, available online: http://www.ieee802.org/1/pages/802.1bh.html

[25] Greenberg, A. et al., "VL2: A scalable and flexible data centre network", ACM SIGCOMM Computer Communication Review, vol 9, pp. 51-62, 2009, URL: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.156.6990&rep=rep1&type=pdf

[26] Greenberg, A. et al., "Towards a next generation data centre architecture: scalability and commoditization", in Proceedings of the ACM workshop on Programmable routers for extensible services of tomorrow, pp. 57-62, 2009, URL: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.153.2831&rep=rep1&type=pdf

[27] IETF TRILL Working Group, URL: http://tools.ietf.org/wg/trill/

[28] DCE http://www.cisco.com/en/US/netsol/ns783/index.html

[29] Mysore, R.N. et al., "Portland: a scalable fault-tolerant layer 2 data centre network fabric", ACM SIGCOMM Computer Communication Review, vol. 39, pp. 39-60, 2009, URL: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.167.1946&rep=rep1&type=pdf

[30] Leiserson, Charles E., "Fat-Trees: Universal Networks for hardware-efficient supercomputing", IEEE Transaction on Computers, Vol. 34, No. 10, p. 892, Oct. 1985

[31] Adrian Farrel, Igor Bryskin: "GMPLS – architecture and applications", Elsevier 2006, p. 329

[32] OpenFlow specification 1.1.0, available http://openflow.org/documents/openflow1.1-allmerged-draft2.pdf

[33] Saurav Das: Extensions to OpenFlow to support Circuit Switching. Addendum to OpenFlow specification 0.8.9 (rc01), available at http://pnrl.stanford.edu/openflow/Draft_Exp_Extn_OFcs.pdf

[34] D. Papadimitriou, M. Vigoureux, K. Shiomoto, D. Brungard, "Generalized MPLS (GMPLS) Protocol Extensions for Multi-Layer and Multi-Region Networks (MLN/MRN)", RFC6001

[35] ITU-T Series-Y: Global Information Infrastructure, Internet Protocol aspects and Next-Generation-Networks – Frameworks and functional architecture models, "General overview of NGN", available online: http://www.itu.int/rec/T-REC-Y.2001/en

[36] IETF; "A Border Gateway Protocol (BGP)"; RFC 1163; ftp://ftp.isi.edu/in-notes/rfc1163.txt

[37] IETF; "Application of the Border Gateway Protocol in the Internet"; RFC 1164; ftp://ftp.isi.edu/in-notes/rfc1164.txt

[38] IETF; "BGP Route Reflection An alternative to full mesh IBGP"; RFC 1966; ftp://ftp.isi.edu/in-notes/rfc1966.txt

[39] IETF; "BGP Communities Attribute"; RFC 1997; ftp://ftp.isi.edu/in-notes/rfc1997.txt

[40] IETF; "IS-IS extensions for Traffic Engineering"; Internet-Draft; http://www.ietf.org/internet-drafts/draft-ietf-isis-traffic-01.txt

[41] IETF; "Constraint-Based LSP Setup using LDP"; Internet-Draft; http://www.ietf.org/internet-drafts/draft-ietf-mpls-cr-ldp-03.txt

[42] IETF; "Extensions to RSVP for LSP Tunnels"; Internet-Draft ; http://www.ietf.org/internet-drafts/draft-ietf-mpls-rsvp-lsp-tunnel-04.txt

[43] IETF; „LDP Specification"; Internet-Draft; http://www.ietf.org/internet-drafts/draft-ietf-mpls-ldp-06.txt

[44] IETF; “A Framework for Internet Traffic Engineering”; Internet-Draft; http://www.ietf.org/internet-drafts/draft-ietf-tewg-framework-00.txt

[45] ITU-T Series-G: Transmission systems and media, digital systems and networks, “Interfaces for the Optical Transport Network (OTN)”, available online: http://www.itu.int/rec/T-REC-G.709/en

[46] ITU-T Series-G: Transmission systems and media, digital systems and networks, “Generic functional architecture of transport networks”, available online: http://www.itu.int/rec/T-REC-G.805/en

[47] ITU-T Series-G: Transmission systems and media, digital systems and networks, “Functional architecture of connectionless layer networks”, available online: http://www.itu.int/rec/T-REC-G.809/en