Project Number:	**CELTIC / CP2012-2-5**
Project Title:	SDN Concept in Generalized Mobile Network Architectures– SIGMONA
Document Type:	P (Public)

Document Identifier:	D 2.1
Document Title:	**Control and management of mobile and transport networks in SDMN**
Source Activity:	WP 2
Main Editor:	Zoltán Vincze
Authors:	See author list.
Status / Version:	1.0
Date Last changes:	18.03.2016
File Name:	D2.1 Control and management of mobile and transport networks in SDMN.doc

Abstract:	This document introduces concepts for streamlining the management of mobile and transport networks in the era of SDN-enabled and virtualized mobile networking environments.

Keywords:	SDN, NFV, mobile backhaul, automated network optimization, virtualization, multipath L2 networks

Document History:	

Table of Contents

Authors

Partner	Name	Phone / Fax / e-mail
6WIND	François-Frédéric Ozog	
	Phone:	+90 216 987 6386
	e-mail:	ff.ozog@6wind.com
AVEA	Engin Zeydan	
	Phone:	+90 216 987 6386
	e-mail:	Engin.Zeydan@turktelekom.com.tr
Coriant	Juha-Petteri Nieminen	
	Phone:	+358 40 8673208
	e-mail:	juha-petteri.nieminen@coriant.com
Nokia Solutions and Networks Kft.	Zoltán Vincze	
	Phone:	+36 20 977 7797
	e-mail:	zoltan.vincze@nokia.com
VTT	Jorma Kilpi	
	Phone:	+358 40 7195134
	e-mail:	jorma.kilpi@vtt.fi
VTT	Kari Seppänen	
	Phone:	+358 40 7006887
	e-mail:	kari.seppanen@vtt.fi
VTT	Tapio Suihko	
	Phone:	+358 40 5529646
	e-mail:	tapio.suihko@vtt.fi

Executive Summary

The flat rate pricing applied in case of mobile broadband services, the continuous proliferation of mobile broadband services and the fierce competition introduce contrary requirements on mobile network operators: they need to provide continuously increasing service quality for continuously increasing amount of data at a continuously decreasing or constant price. This makes the operators to replace network management and monitoring processes well-known and understood from the voice traffic dominant era with new ones which can provide highly cost-efficient operation for the data traffic dominant era. In parallel operators and vendors search for ways to change mobile network architecture to be better aligned with the aforementioned requirements. To achieve these goals, operators and vendors look into ways to deploy SDN and network virtualization technologies in mobile networks which are seen as enablers for streamlining and making future-proof the operation of mobile networks. This document proposes concepts in the area of management of mobile and transport networks that could enhance network operation to meet the demands of serving exponentially growing traffic without major operating cost increase. First, it is described how SDN and virtualization impacts MBH networks and after that four conceptual solution are proposed:

1. Integrating WMNs under SDN control
2. Integrating RAN and MBH control
3. Multipath L2 networks
4. Harmonized network and resource management in SDMN

List of terms, acronyms and abbreviations

API	Application Programming Interface
ARP	Address Resolution Protocol
ASIC	Application-Specific Integrated Circuit
BFD	Bidirectional Forwarding Detection
BSS	Business Support System
BTP	Backhaul Transport Provider
CDN	Content Delivery Network
CoMP	Coordinated Multi-Point
CPU	Central Processing Unit
CQI	Channel Quality Indicator
DPDK	Data Plane Development Kit
DSCP	DiffServ Code Point
E2E	End-to-End
eNB	eNodeB
EPC	Evolved Packet Core
EPS	Evolved Packet System
ETSI	European Telecommunications Standards Institute
GS	Glue Switch
GW	Gateway
HSS	Home Subscriber System
HW	Hardware
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
ICIC	Intercell Interference Coordination
IP	Internet Protocol
ISL	Inter-Switch Link
LAN	Local Area Network
LINP	Logically Independent Network Partitions
LSP	Label Switched Path
LTE	Long Term Evolution
LTE-A	LTE-Advanced
LXC	Linux Containers
MAC	Media Access Control
MC-LAG	Multi-Chassis Link Aggregation Group
MBH	Mobile Backhaul
MBHC	MBH Controller
MME	Mobility Management Entity
MNO	Mobile Network Operator
MPLS	Multiprotocol Label Switching
MST	Mono Spanning-Tree
NB	Northbound

NFV	Network Function Virtualization
NPU	Network Processor Unit
NV	Network Virtualization
mmW	milimeter Wave
OAM	Operation and Maintenance
OF	OpenFlow
ONF	Open Networking Foundation
ODP	OpenDataPlane
OSS	Operation Support System
OVS	Open vSwitch
PC	Personal Computer
PCP	Priority Code Point
PCRF	Policy and Charging Rules Function
PDN	Packet Data Network
PGW	Packet Data Gateway
PVST	Per-VLAN Spanning Tree
QoE	Quality of Experience
QoS	Quality of Service
RAN	Radio Access Network
RANC	RAN Controller
RAT	Radio Access Technology
REST	Representational state transfer
RF	Radio Frequency
RFC	Request For Comments
RRM	Radio Resource Management
SB	Southbound
SDMN	Software-Defined Mobile Network
SDN	Software-Defined Networking
SMA	SON for MBH Agent
SGW	Serving Gateway
SoC	System on a Chip
SON	Self-Organizing Network
STP	Spanning Tree Protocol
TCP	Transport Control Protocol
TLV	Type Length Value
TNO	Transport Network Operator
UE	User Equipment
VLAN	Virtual LAN
VM	Virtual Machine
VNF	Virtual Network Function
VS	Virtual Switch
WFS	WMN Front-end Switch
WMF	WMN Mediator Function
WMN	Wireless Mesh Network

List of Figures

List of Tables

1. Introduction

The evolution of radio access technologies in mobile networks has created the possibility to provide true mobile broadband experience for customers who, regardless of their location, expect a level of service quality and user experience similar to what is provided by the fixed broadband access when running popular Internet based services and applications. The capability of providing this experience to the customers is a key requirement whenever mobile broadband is to be offered as a valid alternative to the fixed broadband. However, cost efficient solutions are essential for mobile network operators in order to maintain their competitiveness in the telecommunications business. High resource utilization achieved by creating a configuration that is end-to-end optimized is one important element of cost efficient operation. This is difficult to implement in mobile networks due to factors such as the unpredictability of actual traffic demand in the network introduced by user mobility, the diverse QoS (Quality of Service) and QoE (Quality of Experience) requirements of the popular applications, the increasing number of network nodes (e.g., small cell deployments), the rigid resource allocation schemes, the coexistence of different mobile network technologies over shared backhaul network (e.g., HetNet deployments), leased resources, etc. Accordingly, the network configuration and management should be based on adaptive, autonomous and flexible mechanisms that are capable of adjusting the system to the fluctuating load and user mobility.

Software Defined Networking (SDN) and network virtualization (NV) are seen as two crucial technologies that can turn mobile networks into Software Defined Mobile Networks (SDMN), which are able to provide highly flexible and efficient network operation which can accommodate the foreseen significant increase in the amount of transmitted data. Though these technologies have similarities there are significant differences and they can be seen as complementary ones that can provide the required flexibility by being introduced together in the network. SDN's goal is to separate the control plane from the data plane in networks. This would enable to introduce a logically centralized controller which is able to control the devices in the network via an API, thus basically the setup and configuration of the whole network would become a programming task. This provides the possibility to re-use formal design and verification methods already developed for software developing in networks and simplify the whole network configuration process. This centralized controller based operation can be an enabler for implementing virtualization solutions developed for computers in networks in mobile networks also.

The scope of this document is to analyze what impacts these changes have on the management of mobile and transport networks and to propose novel control and management solutions utilizing the innovation possibilities opened up SDN and NFV in mobile networks. Section 2.1 will first introduce how SDN can be integrated into mobile backhaul networks and how mobile backhaul network elements can be virtualized. After that a concept is introduced that aims at integrating wireless mesh networks (WMNs) into SDN control. This is not a simple task, as the legacy WMN management functions (like load balancing) requires very low latency control loop, whereas delegating those functions to a central SDN controller would result in too high latency possibly ruining the operation of those functions. The concept introduced in section 2.3 will address this issue. The solution proposed by Section 2.4 aims at adding multipath L2 networks support for OVS. Section 3.2 introduces the concept of integrated RAN and mobile backhaul controller that has the benefit of considering backhaul network status in RAN management. The concept of harmonized network and resource management (HNRM) is introduced in Section 4.2. This concept looks at cross domain and cross layer optimization possibilities of mobile networks.

2. Software Defined Mobile Backhaul

2.1 Architectural overview

2.1.1.1 Architecture building blocks

The key building blocks of SDN mobile backhaul networks are network elements, SDN controller functions, measurement functions and orchestration functions, as illustrated in Figure 1.

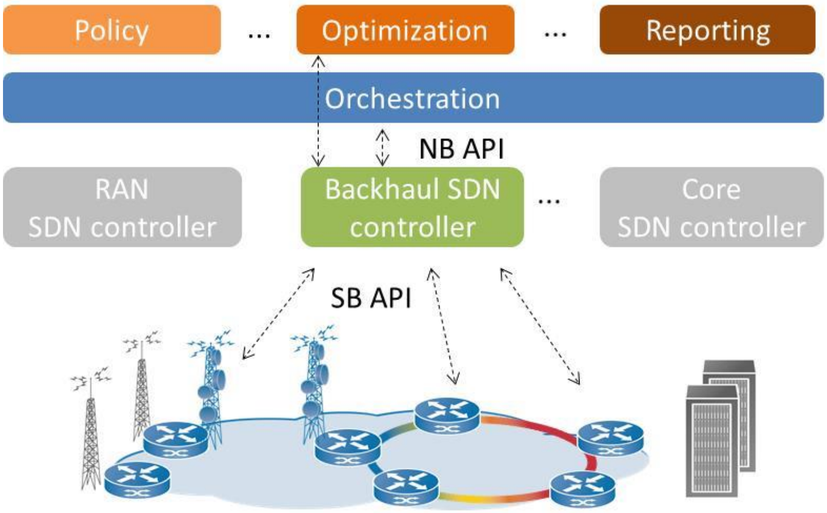


Figure 1: Architecture building blocks

The key task of network elements is to forward traffic. They also host measurement functions and provide data for reporting, analytics, optimization, etc. purposes. Network elements provide API (often referred to as southbound API, or SB API for short, as network elements are “south” from SDN controllers) for control and data retrieval purposes.

SDN controllers program network elements to set up backhaul connections. They provide API (often referred to as northbound API, or NB API for short) for higher layers components such as orchestration, reporting, etc. functions to request services such as new backhaul connections and read performance reports etc.

Measurement functions gather data (capture certain types of packets, collect performance statistics, etc.) or perform test operations such test packets transmission. Measurement functions may be hosted by network elements or separate appliances (often called probes). Measurement and other data such as statistics, faults, event logs, etc. may be processed by data analytics functions e.g. for optimization purposes.

Orchestration functions coordinate overall mobile network system including radio network, backhaul, core functions (gateways), service functions (CDNs, security, etc.), etc. From the backhaul network point of view orchestrators typically request backhaul connection services.

2.1.1.2 Network elements

SDN backhaul networks are typically packet switched. Under packet switching layer, they utilize various transmission technologies such as Ethernet, radio transmission and optical wavelength switching. Transmission technologies may be controlled by SDN controller, for instance in order to control all network functionality from one controller or achieve better overall optimization, as discussed in Chapter 2.1.1.4, or alternatively they be left outside SDN control, e.g. in case of simple point-to-point Ethernet links.

Figure 1 illustrates use case where SDN controller in addition to packet layer configures optical switching layer in aggregation network and radio links. This enables, for instance, to adjust packet forwarding based on radio link capacities (that may change dynamically) and optimize optical switching topology. Another example of multiplayer is to have protection only one layer and not all, which is often the case in traditional networks.

SDN network elements vary significantly on how much functionality is retained in network elements and how much is done in the controller. In one extreme case, the network elements have forwarding plane and minimal amount software to expose it via a low level, typically OpenFlow [OF], and all control plane functions are performed in SDN controller. In another extreme case, the network elements maintain current control plane functions (typically IP/MPLS) and SDN controller participates only a small portion of tasks, such as traffic engineering and optimization.

It is seen likely that optimal solution is neither extreme, but something in between so that tasks that require short response times and can be done based on local information are implemented in network elements. Examples of such functions are protection switchover decisions between preconfigured alternate routes and responding control messages that have local significance, such as ARPs.

Another thing that varies a lot is the abstraction level of APIs. Low-level protocols such as OpenFlow enable fine-grained configuration of network elements, but on the other hand require SDN controller to specify every detail. That is especially difficult when different types of network elements have different restrictions due to HW or SW limitations. This is one of the reasons why SDN controller needs adapter layer, as discussed in Chapter 2.1.1.3.

Higher level APIs provide less flexibility but on the other hand make configuration easier as there are less things to configure and different network elements are likely to provide compactible functionality. Examples of higher level APIs are YANG models [YANG] exposed via Netconf [NETCONF] and Restconf [RESTCONF]. It is possible, though expose low level YANG models via Netconf and Restconf.

Again, best solution is neither extreme, but utilize low level APIs for functions that benefit from flexibility, such as making service level forwarding decisions, and utilize higher level APIs for functions for other functions, such as Operation and Maintanence (OAM) etc. measurement functions.

2.1.1.3 SDN controllers

On a high-level mobile view backhaul SDN controller consist of controller core functions (often referred to as network operating system), network element adapters and applications, as shown in Figure 2.

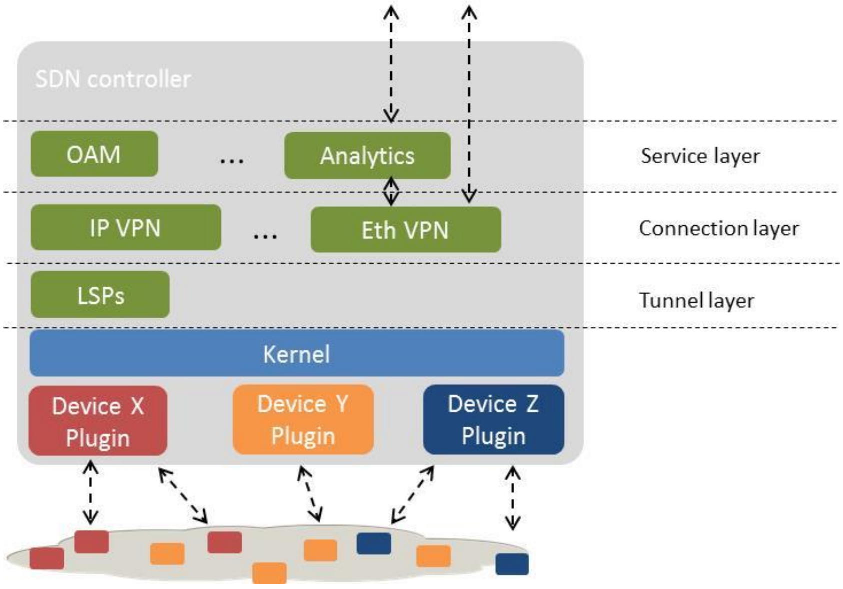


Figure 2: SDN controller architecture

The role of the controller core (kernel) is similar to operating system in computers. It provides a standardized (sockets, message passing, etc.) way for applications to communicate with network elements as well as each other. It also maintains a shared database for network resources and configurations, and by that way acts as resource broker.

The role of network element adapters resembles much the role of HW drivers in operating systems, i.e. they hide the differences of hardware components, in this case network elements. This enables to use different API protocols for different network elements. They also hide implementation differences, different protocol versions, etc.

Like in the computer operating systems, applications perform the actual user visible tasks. The set of required applications varies from the use case to another, but is likely that following basic applications exist in most SDN backhaul systems:

- Tunnel management application
    - Calculate tunnel (e.g. MPLS LSP) paths and configure to network
    - Typically include alternate path and protection and/or load balancing configurations
- Connection management applications
    - Different types of applications for different types of backhaul connections, e.g. for IP routed connections, transparent L2 connections, or user specific flow connections to support user mobility by backhaul network
- Test, measurement and data collection applications
    - Test and measure network entities such as connections, tunnels, links, etc.
    - Collect data such as statistics and event logs
    - Provide processed (analyzed) data to other controller applications such as connection management applications as well as other uses such as visualizations to operator’s personnel.

Applications utilize different abstractions of network. For example, tunnel management application needs to see all network elements and links so that it can route tunnels whereas connection management applications only want to see user facing equipment and tunnels connecting them, as illustrated in Figure 3.

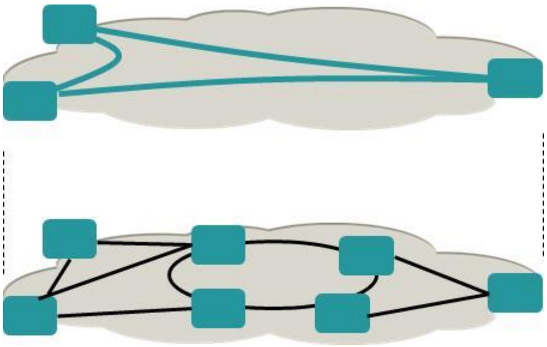


Figure 3: Network topology layering within SDN controller

Tunnel and connection management applications together with measurement and data collection applications may form a self-optimizing feedback and control loop, as shown in Figure 4. By this means SDN is way to implement a SON functions to backhaul network.

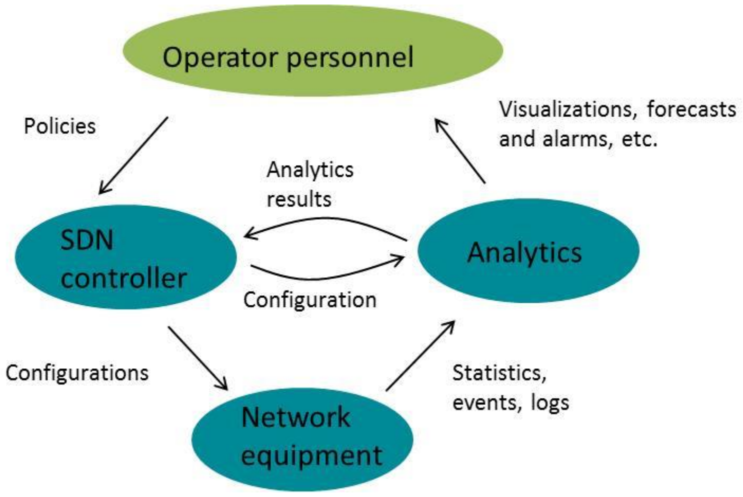


Figure 4: Self-optimization loops in SDN mobile backhaul

Applications provide northbound API for upper layers, which is in backhaul case typically for some form of orchestration function or dashboard for manual connection creation. Northbound APIs typically utilize web technologies such as REST [REST].

2.1.1.4 Measurement and data analytics functions

There may be multiple types of test, measurement and data analytics functions for different uses. OAM protocols such as Ethernet OAM [ETH OAM] or Bidirectional Forwarding Detection (BFD) [BFD] are used for service reporting as well as triggering actions, such as protection switchovers. When used for triggering actions corresponding applications (tunnel application in this case) configure protocol instances and actions according to overall configuration (in this case BFD to monitor each tunnel and trigger switchover in case loss of connection).

Test and measurement functions may also be used for debugging, for instance to implement network level debugger described in [NW DEBUG]. The example illustrates one big benefit of SDN for testing and debugging use; SDN controller knows all network configuration so test measurement results may be easily correlated with appropriate network resources (such as missing lookup entry).

Data collection functions gather data such as statistics, logs and events from network for analysis. Analytics results are provided via APIs to other control functions for optimization purposes, as was discussed in Chapter 2.1.1.3.

2.1.1.5 Orchestration functions

Orchestration functions control overall mobile service, consisting of radio networks, backhaul, core (gateways) and other functions such as content service functions. From backhaul point of view orchestration function is an entity that requests new connections, modifies existing and deletes unnecessary. Orchestration functions also retrieve processed/analyzed performance etc. data for overall service optimization.

Orchestration function may, for example, allocate virtualized mobile core functions to different locations and based on core functions placements create/modify/delete backhaul connections accordingly.

2.2 Virtualization of MBH elements

2.2.1 Background

Mobile backhaul network elements are nowadays typically vendor specific HW appliances that utilize Application-Specific Integrated Circuits (ASICs) or Network Processor Units (NPUs) for packet forwarding. They also typically have vendor specific operating systems. This prevents from running generic code in different network elements as well as running multiple functions (such as core site backhaul router and mobile gateway functions) in parallel in shared hardware.

The rise of modern virtualization technologies and multicore processor has triggered interest in implementing network functions by using general purpose computing hardware. This is often referred to as Network Functions Virtualization (NFV) according to the white paper that triggered the movement [NFV WP]. As the white paper indicates, the initial driver has been higher level processing (such as mobile gateways) than mobile backhaul, but high emerging performance and low power consumption multicore processors may make NFV good candidate for backhaul as well.

2.2.2 Virtualized network element architecture

NFV network element architecture resembles virtualized server architecture, or more specifically is a virtualized server optimized for packet processing. The system has a host operating system (typically Linux) and hypervisor that allocates resources to Virtual Machines (VMs) that run guest operating systems (again typically Linux). Virtual machines run Virtualized Network functions (VNFs) that perform actual packet processing tasks as illustrated in Figure 5. Depending on the use case functionality of an application, say backhaul router, may be in one or more VMs.

Instead of VMs, virtualization may also be done by using container virtualization technologies such as Linux Containers (LXC). Containers differ from VMs in that all containers share kernel with the host operating system whereas all VMs have their own kernel. This reduces overhead but on the other hand, all VNFs must use the same kernel version.

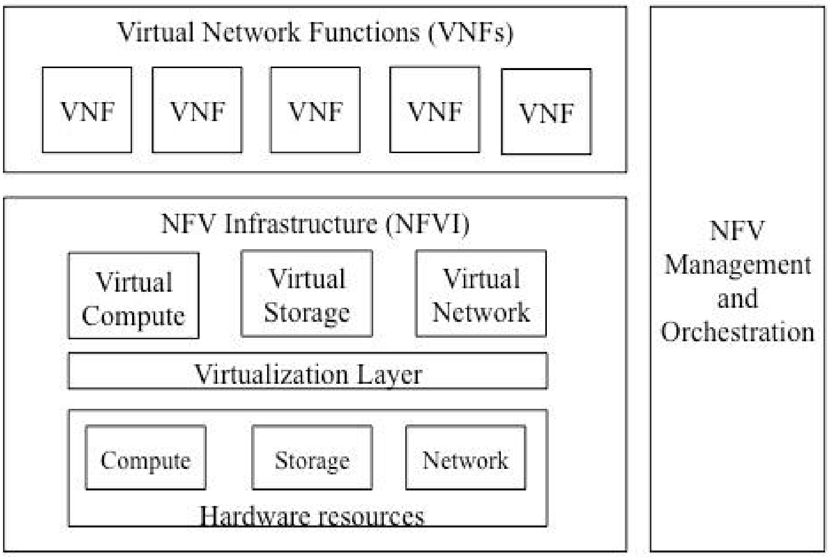


Figure 5: NFV architecture (source ETSI)

2.2.3 Packet forwarding software

Packet forwarding by using general purposes Central Processing Units (CPUs) and the normal operating system procedures, e.g. by copying data from kernel space (where network drivers reside) to user space (where processing happens), is not feasible, so various solutions have been proposed to optimize performance as well as provide common basis for forwarding code.

The most well-known frameworks are DPDK [DPDK] and ODP [ODP]. The former is originally by Intel and thus built for x86 architecture whereas the latter is driven mainly by System on a Chip (SoC) vendors using ARM cores. Both can, however, be used in other than the original architectures.

Both frameworks provide basic primitives such as packet receive and transmit buffering. Actual forwarding code is not part of frameworks and needs to written for each application.

2.2.4 Implementing mobile backhaul network by using virtualized network elements

A mobile backhaul network has a large variety of network elements from cell sites to gateway sites. Capacity requirement vary from tens of megabits per second at small cell sites to hundreds of gigabits per second at large core sites. Therefore one solution does not fit all, but implementation architectures are different at different locations.

It is seen likely that low capacity NFV elements, used e.g. at cell sites, are implemented by using single multicore processor or SoC. In addition to mobile backhaul functions, the same physical element may host other functions, such as base station or content caching functions.

In the high capacity end, the most promising use case for NFV is seen in applications where mobile core functions have been virtualized to server pools. In such cases virtualizing also mobile backhaul router functions allows operator to run all functions in general purpose hardware.

2.3 Integrating WMN based mobile backhaul with SDN control

2.3.1 Introduction

Increasing capacity demands in broadband mobile networks call for new solutions especially in dense urban areas. To meet these needs, the traditional macro-cell architecture has to be augmented with small cells that cover the hot-spot areas. One of the major problems in small-cell deployment is the lack of suitable fixed wireline backhaul connections in many potential installation sites like lamp posts and bus stops. Furthermore, this development can easily lead to multiplication of the number of cells by a factor of 10. This means that the cost of small-cell installation must be brought down as low as possible with, e.g., zero-configuration [NGMN-BHR].

Installing new fiber optic network connections for small cells is very costly in dense urban areas and, in many cases, also very time consuming because of the required planning and official permits. Thus, the use of wireless backhaul connections is a natural choice for these kinds of small-cell scenarios. However, the capacity requirements for LTE-A and forthcoming 5G are such (>= 1 Gbit/s per base station) that they are hard to meet with current wireless systems. Millimeter wave (mmW) RF systems (e.g., 60 GHz or 71-88 GHz) can provide ample capacity to meet the requirements.

Using mmW RF technology makes it necessary to apply narrow-beam directed point-to-point links between stations to provide sufficient link budgets for usable link spans. This is actually an advantage as it increases the total system capacity compared to omni-directional transmissions. However, narrow mmW beams are rather vulnerable to disturbances and thus the reliability of an end-station with only one point-to-point link could be quite low. The solution for this problem is to have mesh connectivity between WMN nodes and gateways. Moreover, using WMN for backhaul connectivity allows for having reliable multi-hop paths between small cells and WMN gateways and thus extending the area that can be covered with a single WMN.

SDN techniques facilitate the provisioning of network services in a deterministic, dynamic, and scalable manner. The software programmability enables agile and automated network configuration and traffic management that is vendor neutral and based on open standards. Network operators are able to dynamically adjust the network’s traffic flows to meet the changing needs while optimizing the network resource usage. An OpenFlow-based SDN is formed by switches that forward data packets and communicate with one or more controllers using the OpenFlow protocol.

The concept that is proposed here integrates an existing mmW WMN backhaul solution with SDN-based centralized transport network control. The main goals of the concept are to combine local and centralized control as well as to provide “plug-and-play” style incremental network extension. Furthermore, the capability of sharing the network resources among multiple mobile network operators (MNOs) is a very important target.

2.3.2 WMN-based small-cell backhaul

Small-cell backhaul should be seen as a part of the whole mobile network infrastructure and the WMN portion as a last mile segment of the backhaul connection [NGMN-BHR]. Thus, events in WMN, like failures, can affect the rest of the network by, e.g., triggering handovers between base stations and protection switching at the fixed network side. This means that, to get best advantages from alternative backup paths that WMN provides, the fault recovery mechanisms in the WMN should operate, in the most of the cases, faster than the "normal" telecom grade protection (50 ms).

There are already concepts that integrate mmW radio links in backhaul and SDN ideas, e.g., hybrid wireless optical MBH described in [BSCW13]. The mobile backhaul concept proposed here is based on a novel mmW WMN system that has been developed in various earlier projects [Taipale12] [MEVICOD32]. This WMN is not limited to repeater (or relay) configurations but it supports multi-hop paths (the limiting factor for hop count is delay tolerance) allowing better coverage and more alternative routes. As a single WMN can be fairly large, multiple gateways to the fixed network are also supported.

Packet routing in the proposed WMN concept is performed at flow granularity. Flows are identified by inspecting L2 and/or L3 headers, e.g., Ethernet MAC addresses together with VLAN Id and PCP (Priority Code Point), or IP addresses and DSCP (Differentiated Services Code Point) field. Each WMN flow can be assigned to a separate path and these assignments can change dynamically based on network state. In case of congestion or link failure, high priority traffic can get better (guaranteed) service while the best effort traffic suffers most of the damage. It is also possible to split one traffic flow to multiple paths within the WMN. Due to related processing overheads, such splitting of flows is usually applied only for "fat" non-realtime traffic flows.

2.3.3 Local versus centralized control

As explained earlier, current and future mobile backhaul requirements are such that it is of paramount importance to hide all WMN impairments as perfectly as possible. In the best case, the WMN portion of the backhaul connection would be seen as a reliable bit pipe – with somewhat elastic capacity. To achieve this kind of performance, fault management mechanisms inside WMN have to react to failures and other events much faster than other fault management mechanisms in the network. In practice, this means just few 10s of ms time scales. As the delays inside WMN could be something around 1 ms per hop, the only viable way to achieve the required reaction speeds is to use local protection and recovery mechanisms. One of the common main ideas of SDN is the optimization of the whole network configuration, as the network state is (in principle) known by a single centralized control entity. This would make the usage of network resources more efficient and, at the same time, make it possible to utilize simple and cheap network equipment.

The main problem with WMN based backhaul and centralized control is that WMN is potentially very dynamic environment. Moreover, optimizing its operation requires lots of quite specific information from each link as well as detailed system specific understanding. Thus, centralized control for WMN would require transferring considerable amounts of status information and configuration commands between WMN nodes and a centralized control entity. In practice, the centralized controller would have to replicate the WMN's currently existing distributed control plane to provide the necessary functionality.

All this would cause extra traffic in the network and additional delays to fault protection operations. As the same functions can be handled locally in the WMN, centralization of control is hard to justify. However, centralized control is very attractive alternative for configuring and controlling end-to-end traffic flows and backhaul connections. Thus, it would be beneficial if the local and centralized control could be made to live together by utilizing the best features from both.

2.3.4 SDN-configurable WMN

The approach to the problem of integrating WMN MBH with SDN control is to leave the most of the WMN functions as they are in the current WMN concept and to use SDN only to configure end-to-end connections. However, the SDN controller cannot be allowed to configure routing inside the WMN as that would mess up the fault recovery, load-balancing and other WMN operations and, vice versa, any self-configuration action taken by WMN would confuse the SDN controller. Thus, a key part of the solution is to hide the WMN internal structure and operations from the SDN layer.

The WMN backhaul solution will be a part of larger backhaul system that includes also fixed legacy and SDN transport network portions and covers the whole backhaul connection from base stations to MNO's mobile core network (e.g., EPC). One of the main ideas in this backhaul system is to provide virtualized network slices to multiple operators.

2.3.5 WMN abstraction and virtualization

Network (or topology) abstraction is a powerful tool that allows construction of hierarchies in SDN [MRFRW13]. The main idea is the same as with abstractions in programming in general: to give the programmer an access to the needed information while hiding all the internals from accidental manipulations. This abstraction principle is exactly what is needed in hiding WMN operations from the SDN controller. The key elements in the abstraction model are that the whole WMN domain is represented as a single virtual SDN switch and that each WMN node port connected to a small cell is shown as a separate (virtual) port in that switch. The abstracted view can be further virtualized such that separate slices of the abstracted network are visible to relevant parties, e.g., to different MNOs (see Figure 6).

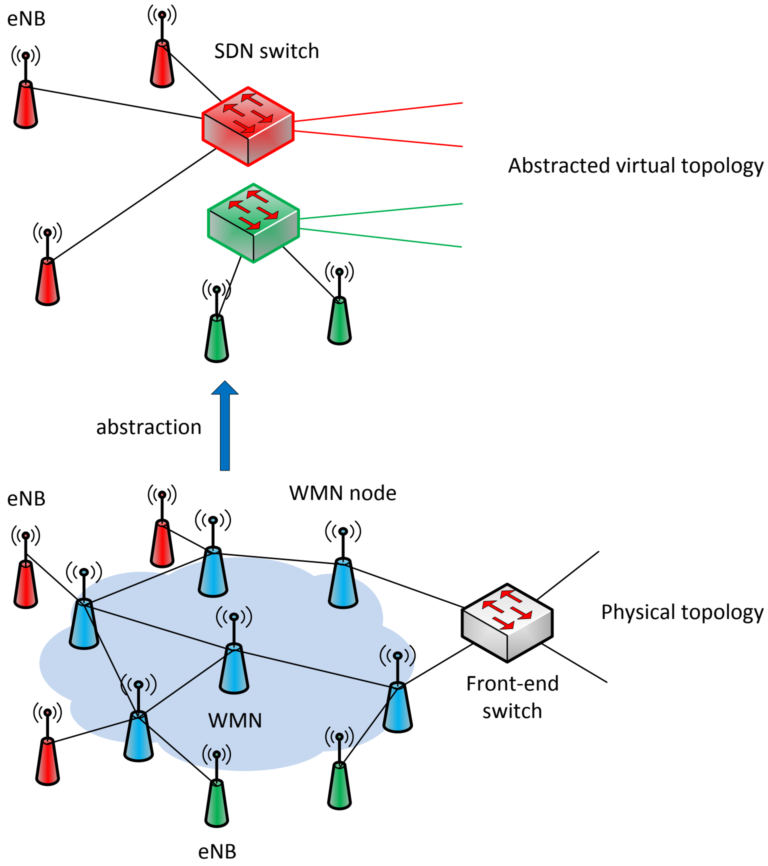


Figure 6: WMN abstraction and virtualization

This effectively hides all the WMN functionality from the upper layers while it, at the same time, offers full control to configure all traffic flows from and to the small cell. This abstraction model can also be used to hide the existence of multiple WMN gateways (GW) in a single domain and, which is quite useful, hide all such protection mechanisms inside WMN that could cause moving traffic flows from one gateway to another.

Hiding the existence of multiple GWs and traffic flow rerouting from one GW to another requires some extra functionality between WMN and the fixed transport network. A WMN Front-end Switch (WFS) is introduced in order to handle this functionality, i.e., switching the WMN traffic flows so that they appear to originate from a single virtual port. The WMN abstraction is provided by WMN Mediator Function (WMF). At the control plane, WMF takes the responsibility of offering the SDN control interface and interpreting SDN commands given to WFS and translating them into WMN's control operations (see Figure 7).

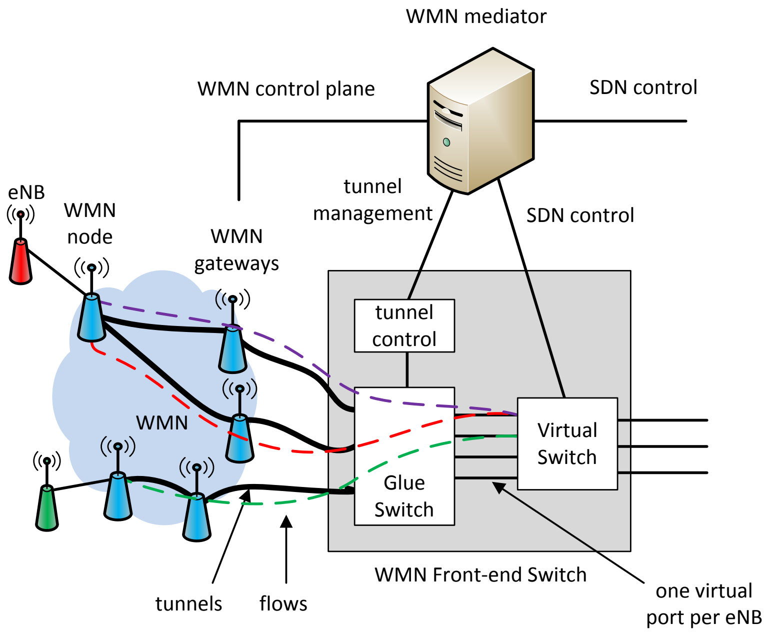


Figure 7: WMN Front-end switch architecture

In the WMN system, the traffic flows are, in practice, tunneled between WMN nodes and GWs, and the paths that these tunnels take are changing dynamically (down to ms scale). Even the target GW for a flow can change if necessary. In a simple legacy network model, the GWs would terminate these tunnels and forward the customer payload to the fixed network. However, in the abstraction model, the peculiarities of WMN need to be hidden from the SDN control plane, e.g., flows moved from a GW to another and flows from one base station passing through different GWs. For this purpose, there is additional tunneling between GWs and WFS. The main purpose of this tunneling is to carry information about flow identification (especially the source WMN node) to WFS.

The changes in traffic flow routing over WMN can be detected in two ways: the WMN control plane sends a notification about path re-selection to WMF or WFS detects that a traffic flow has moved from one GW-WFS tunnel to another. The latter case can be handled in OpenFlow (OF) like manner: WFS sends the “unknown” packet to WMF, which makes the WFS reconfiguration after inspecting the packet header (in this case, tunnel specific header). In the WMN system, downstream and upstream traffic could have separate paths and even via separate GWs. However, in this case, each flow is forced to use the same GW in both directions. This allows that also downstream GW change can be triggered by a WMN node. Thus, when the change of the GW for the upstream traffic is detected, WFS is reconfigured also to reroute downstream to the same GW. The GWs automatically adapt to this change and the flow paths for upstream and downstream traffic within the WMN (between GW and the WMN node) can still be different.

The system will also support direct base station to base station connectivity (e.g., for LTE X2 traffic). The idea is, that when the mediator identifies an SDN command that tries to configure a connection between two WMN side ports in WFS, it asks WMN control plane to configure a direct connection inside WMN. Thus, the traffic can take the shortest route instead of being hauled over WFS. However, this causes also some problems with port statistics: it is not sufficient to just return the counters from WFS but these values have to be merged with intra-WMN traffic counters.

The current design is based on two virtual switch instances inside WFS (as shown in Figure 7). The “Glue Switch” (GS) is taking care of routing WMN flows between GWs and WFS virtual ports. These virtual ports are, in fact, WMN side ports of the second switch instance, Virtual switch (VS). The two-switch approach makes it possible to use some of the existing virtual switches as VS, e.g., Open vSwitch [OVS] or Indigo Virtual Switch [IVS]. In this configuration, WMF can pass most of the OF commands directly to VS and VS has all the required functionality to provide the abstracted network view for SDN controllers. Furthermore, all effects of the changes in traffic flows in WMN side are limited to GS.

2.3.6 Incremental network infrastructure extension

In network extension, a new small cell is installed to a hot-spot location and, and after power-on, the new small cell should be brought into active state automatically. The new small-cell base station can be connected to an existing WMN node or the WMN node can be installed at the same time (as a separate co-located unit or integrated to the cellular base station). In the latter case, WMN self-configuration procedures will first initialize the WMN node, which can then provide transport network connectivity for the small-cell base station. In any case, the network infrastructure extension should not need any human interaction besides the actual physical installation procedure and powering up the new equipment.

During the WMN node configuration, the WFS is also configured to facilitate network extension. WMN control plane notifies WMF about the new WMN node and its configuration. Using this information, WMF adds new virtual ports to WFS virtual switch and configures each port so that all small-cell authentication related messages are forwarded to authentication function ("AAA" in Figure 8). All other traffic can be dropped by default until further configuration.

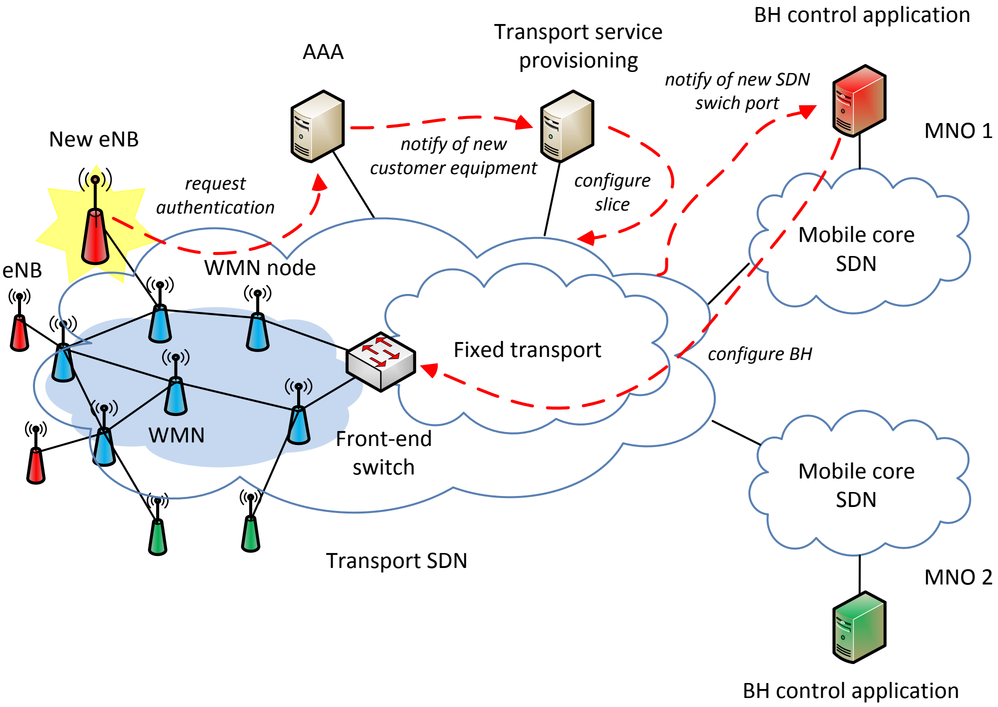


Figure 8: Infrastructure extension use case

When new small-cell base station is installed and switched on, it should first try to authenticate itself and get some basic network configuration information. At the first phase, authentication packets are received by Transport Network Operator's (TNO's) "AAA" service that identifies the owner of the new base station. If the identification and authorization is successful, TNO's provisioning element will reconfigure network virtualization so that the virtual port, to which the new base station is attached, will be added to the network view of the correct MNO. As a result, each MNO should have its own virtualized view to the abstracted WMN (as shown in Figure 6).

When MNO's network controller is notified about the new port and thus about the new base station, MNO can continue with its own authentication procedures. When authentication has been passed, MNO can activate its "backhaul control application" that configures connections between the new base station and mobile core network.

For WMN virtualization, FlowVisor (FV) [SGYA+09] should provide sufficient functionality, but it is unclear if it meets the needs of the whole backhaul virtualization.

2.3.7 Issues in fault tolerance and QoS assurance

There are some issues about reliability in the current WMN abstraction scheme. Only one WFS between WMN and fixed transport network is a single point of failure. It is true that failure rates at fixed network side hardware are much lower compared to, e.g., WMN link failures – especially if high-availability equipment is used. However, this situation is not satisfactory if, above all, WMN is used as a part of mobile backhaul network. This problem cannot be solved just by adding a second WFS in parallel, as it would break down the WMN abstraction. One possibility is to mimic some kind of MC-LAG (Multi-Chassis Link Aggregation Group) functionality (similar constructions are already used in current network edge realizations) and hide that functionality inside the WMN abstraction.

Network virtualization in SDN is quite commonly understood simply as just slicing physical resources (e.g., OpenFlow switches) to provide somewhat isolated network slices for multiple SDN controllers. In this simple virtualization model, all controllers can see the actual physical topology of their network slice. In the proposed concept, the network abstraction hides the actual physical topology and, in this sense, it is not directly compatible with the current "controller should know everything" models. The above-mentioned ideas of network virtualization are closer to, e.g., ITU-T Y.3011 model of Logically Independent Network Partitions (LINPs) and the development of the WMN backhaul concept may need to be steered more towards that direction [Y3011].

When the WMN is shown as a virtual switch to the MNO, each switchable connection (which is actually a path or even a collection of paths) has some resiliency metric (path availability) due to mesh connectivity and some path E2E delay metric(s) due to predictable link scheduling. Given the routing and the scheduling of the WMN, these metrics have some computable optimal targets and measurable realizations.

One potential problem with the WMN abstraction model is that there is no clear capacity concept inside the WMN. The link throughputs are not the only varying factor as the dynamic path selection is also changing path allocations. Thus, the capacity that one flow "sees" might fluctuate all the time. Furthermore, the flows between base station and virtual port seen by SDN control can have separate paths and thus they do not share the same fate. The first problem is with SDN flow configuration: if the MNO wants do capacity reservations, then what capacity value should be given to each virtual port. The second problem is with capacity fluctuations: if MNO tries to optimize SDN flow routing by monitoring flows, it is necessary to quickly identify which impairments are due to WMN (cannot do anything) and which ones caused by rest of the MBH.

When two or more MNOs have a slice of the same WMN the capacity problem has another dimension: temporarily the capacity of the WMN can be lower than the sum of the slices sold to the MNOs. There should be some business oriented but fair approach to diminish the available capacity of all MNOs.

A possible solution is to define the marketable capacity of the WMN as a function of the (total) available capacity of the gateway(s') WMN-side links. This means that capacity fluctuations of the gateway links only are taken into account. Capacity fluctuations of all other links are ignored. This information would be readily available at the gateway(s), without signaling delays. As far as the mesh topology assumption holds this approximative approach is actually quite justified, even for pathwise decisions, but in real life there will also be non-mesh topologies, e.g., tail sites and ignoring the capacity fluctuations of these unprotected tail links is less justified.

2.4 Transport and security at the data-plane level

The goal of the concept introduced in the next subsections is to allow virtual applications to participate into multipath L2 networks and establishing security through IPSec concentrators.

2.4.1 Multipath L2 networks

2.4.1.1 Background

Wireless mesh architectures can grow to very large flat networks building on experience of several fixed networks. This allows significant operating cost reductions but rely on technologies to exploit multipath capabilities of the network. As a matter of fact, when using a single broadcast domain, there can be a single L2 path between two points of the network. This is simple from an operations perspective but does not leverage multiple physical links that may exist between those two points. As a consequence, the industry, led by Cisco, created PVST (Per-VLAN Spanning Tree) and then PVST+ that allows each VLAN to run its own spanning-tree protocol which in turn allowed to use all physical links between two locations across all VLANs. Applying this to virtualized environments is not straightforward as it requires at least continuity of physical VLAN to virtualized VLANs in the host (OVS).

2.4.1.2 Implementation considerations

Per-VLAN Spanning Tree (PVST) is a Cisco proprietary version of STP and maintains a spanning tree instance for each VLAN configured in the network. It uses Inter-Switch Link (ISL) Trunking and allows a VLAN trunk to be forwarding for some VLANs while blocking for other VLANs. Since PVST treats each VLAN as a separate network, it has the ability to load balance traffic (at layer-2) by forwarding some VLANs on one trunk and other VLANs on another trunk without causing a Spanning Tree loop.

PVST+ allow interoperability with non-Cisco devices. Non-Cisco devices usually implement what is called “MST” (Mono Spanning Tree). Mono Spanning Tree means that there is one single spanning tree that applies to all the VLANs. A port is either forwarding or blocking, it can’t be forwarding for one vlan and blocking for another VLAN. PVST+ was recently supported by OVS, added by David Marchand from 6WIND. The aim is to support high bandwidth multipath L2 networking over a wide interconnect of radio antenna.

With current STP support, OVS handles the STP frames to run a STP instance per bridge instance. All ports belonging to this bridge have an associated state (blocking, listening, learning, forwarding, disabled). The disabled state means that this port is not part of the STP instance, yet, this port can still forward packets.

The ports states are used:

- to invalidate the mac learning table
- to avoid flooding on disabled ports (when a mac is unknown)
- by the composed_output_action action which sends packet on a port

We will rely on this STP implementation in OVS to avoid code duplication since the PVST+ instances run the same states machine. The only peculiarity resides in the fact that by default, a Cisco switch sends "normal" STP frames to discover other bridges. No PVST+ frames are sent unless another STP peer is noticed. OVS daemon will run PVST+ engines that will encapsulate STP instances (one for each) and maintain states for all ports belonging to the associated vlan. To run these engines, OVS daemon will have to send PVST+/STP frames. Whatever the configuration, each OVS port belonging to an OVS switch that runs PVST+ must send "normal" STP frames _and_ PVST+ frames.

The choice at the moment is as follows:

- always send a STP frame which will be associated to the PVST+ instance linked to the "native" VLAN of the port
- send PVST+ frames for each PVST+ instance on the port with a 802.1Q header, with the only special case for the "native" which will send a PVST+ without a 802.1Q header.

On the other hand, STP and PVST+ frames will come from other peers, so on reception,

- a STP frame is associated to the right PVST+ instance based on the "native" VLAN of the port
- a PVST+ frame is first checked so that the "originating vlan" (from the cisco TLV) is consistent with either the 802.1Q tag, or with the native vlan of the port. If this check fails, a warning message is logged and the frame is dropped. If this check succeeds
- a PVST+ frame without a 802.1Q tag is ignored since it should only refer to the "native" vlan of a port. This frame is a duplicate of the STP frame that are received.
- a PVST+ frame with a 802.1Q tag is handled if the tag belongs to the trunk configuration of the port. It is then associated to the right PVST+ instance.

2.4.2 Secure backhauling

Mobile networks bandwidth is continuously rising and as smartphone bandwidth can go over 100Mbps, the total amount of bandwidth at the backbone edge can be several tens of gigabits if not hundreds of gigabits. Up until recently only very high end dedicated equipment offered such capacity but at a cost incompatible with the business models of the mobile operators.

6WIND recently validated that a single 4U x86 server could actually handle over 130Gbps of IPSec leveraging 6WIND Turbo IPSec VMs and 6WIND Virtual Accelerator complementing an installed OVS. The test was conducted using web server application and Spirent HTTP load tester to ensure the bandwidth was fully accessible at the application layer, not just at the packet layer. As a matter of fact, the latency induced by some IPSec stacks results in TCP congestion algorithms triggering and self-limiting traffic emitted.

3. Software Defined RAN

Softwarization of radio-access network relies on modified controller functions augmented with network- and radio environment-awareness as opposed to a limited and fragmented view of radio and core network segments. SDN RAN requires an advanced RAN controller (RANC) architecture which can monitor and act according to various system state information such as resource requirements, mobility and QoS. Moreover, such operation has to be integrated with mobile backhaul counterpart to meet end-to-end performance requirements in a hierarchical architecture.

3.1 Architectural overview

The RAN architecture works on a global hierarchical architecture as being part of a lower hierarchy that is part of an end-to-end cellular network architecture in a SDN-enabled LTE network.

The architectural part of the software defined LTE in the RAN has three main components. In the infrastructure layer, there are radio access nodes, i.e. small cells or base stations, while in the middle layer there is the RAN controller (RANC). In the application layer, there are applications that runs on top of RAN controller.

The infrastructure layer elements (base stations, access points, small cells, etc.) collect RAN related parameters and send them periodically to the RANC via the southbound interface denoted as *RAN Interface API*. The southbound interface is like a new *OpenFlow-like protocol* communicating with network devices with "purpose-built" functionalities.

RANC controls the monitoring and management of the radio access network. It handles the necessary coordination between multiple base stations for radio related functions such as inter-cell interference cancellation (ICIC), coordinated multipoint (CoMP) transmission, and joint scheduling. For improved performance, RANC coordinates multiple eNB's transmissions in order to minimize their interference to each other using latest LTE technology releases in the form of CoMP transmission and ICIC. The aim of the RANC is to facilitate central control of eNBs in a coordinated manner for a given geography. As part of a lower hierarchy, RAN controller sends abstracted feedback to higher controller (in this case the mobile backhaul (MBH) controller) based on the higher controller's polling request.

In the application layer, core network functionalities of the wireless networks such as radio access technology selections, radio resource management, radio QoS, and policies are all realized as multiple RAN control applications running on top of the RANC. In this way, by selecting an appropriate RAN control application, flows are distinguished by channel qualities of users, mobility conditions, network frequency, mobile phone capability and other network related states that maintain the radio access network in working state.

3.2 Design and integration of RAN controller with MBH controller

RAN Controller (RANC) is a controller entity in SIGMONA's concept of mobile network architecture for RAN segment. The integration of RANC aims to provide the following capabilities in RAN in addition to fundamental radio resource management and allocation:

1. Optimization of radio resources: RANC monitors the radio resources of RAN in a spatio-temporal setting and manages the radio resource allocation based on QoS, mobility and network policies.

2. Better QoS/QoE for mobile users: Environment-awareness due to RAN monitoring and utilization of user profiles enable better QoS/QoE support and provisioning of differentiated service to mobile users.

3. Backhaul-Radio Segment parity: The resource allocation and management in radio access segment is supposed to match the resource allocation in mobile backhaul for a consistent end-to-end user experience. For instance, a radio resource provisioned in the radio link can be problematic if the backhaul does not entail the necessary transmission capabilities at that time instant.

For a mobile RAN covering a large geographical area, RAN controllers may be deployed in a distributed fashion. These entities may be working as peer entities or may select one of them as the *super-RANC*. That latter one is then responsible for the integration with the rest of the network control plane. However, the delay and scalability aspects are important for such a distributed deployment. A distributed deployment is favorable for minimizing delay and micro-scale visibility of RAN while it may incur scalability issues due to communication and cooperation overheads..The general architecture for RANC-integrated SDMN is shown in Figure 9. The southbound interface *RIA* towards RAN elements collects system information such as congestion, link states, and service requests.

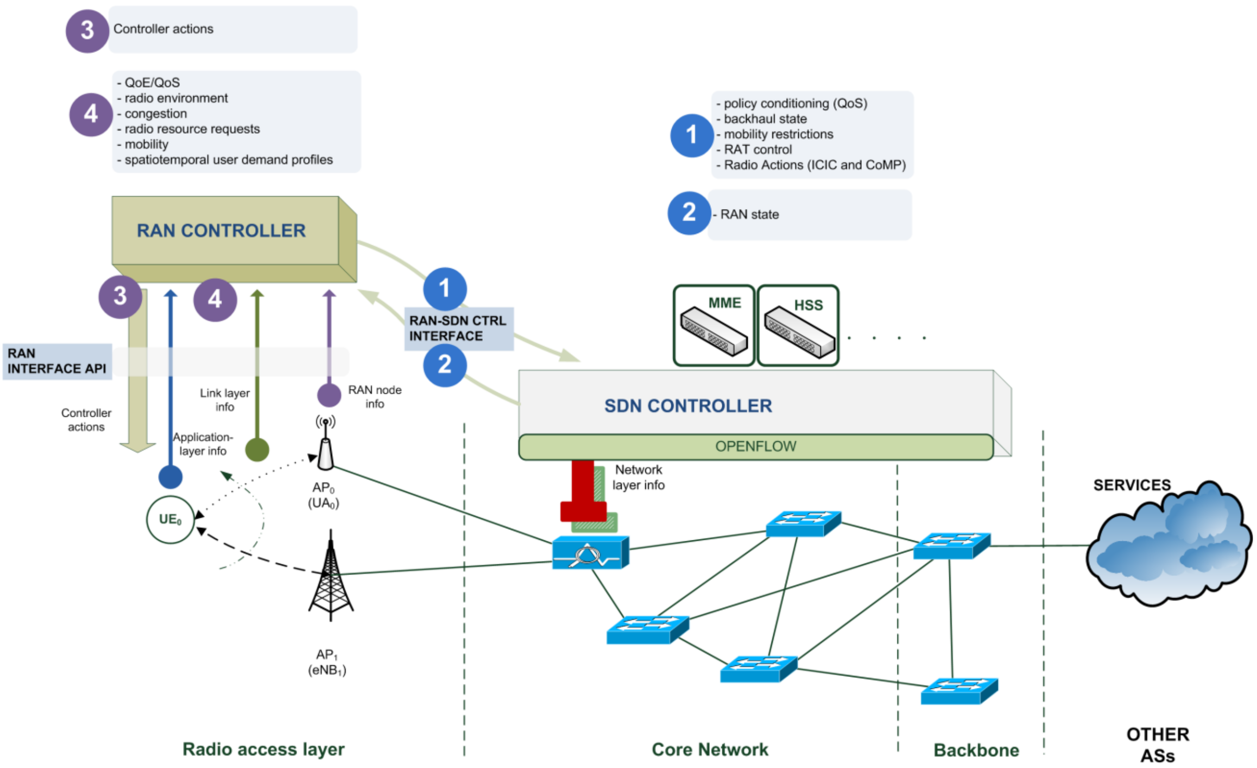


Figure 9: The general architecture for RANC-integrated SDMN

MBH integration allows the feedback of RANC information to the MBH controller. This integration has the following purposes

- MBHC control over RANC as a higher layer in the controller hierarchy: MBHC may poll RANC for retrieving RAN state and control RANC as the higher-in-command in the control plane.
- Sychronization of RAN and MBH resource allocation: An efficient control plane should provide an end-to-end control of network resources and entities. Accordingly, different network segments should be managed in an integrated and cosistent manner.
- QoS control based on MBH state: This aspect is related to previous item and requires MBHC and RANC cooperation.
- Resilience and fault-tolerance: MBHC can switch to different RANC or execute load-balancing actions for evading RANC failure cases.

Therefore, it requires the following data elements in the RANC interfaces:

- Policy set (QoS)
- Mobility restrictions
- RAT selection rules/control
- CQI and channel state information
- RAN node congestion
- <Action, Rule> tuples

These information sets may be per UE, per group of UE or global. Moreover, they may be applicable to per eNB, per group of eNB or all eNBs in that RANC's dominion. RANC operation is controlled by the configuration parameters listed in *Table 1*:

Table 1: Configuration Parameters for RANC operation

Parameter	**Description**
S	set of connected RAN nodes
Tc	Data collection period
Ts	Decision logic period
M	set of connected MBH controllers
L	set of connected other RANCs
C	set of supported data elements for RAN monitoring
A	set of supported RAN actions

The configuration of a specific RANC depends on the data elements it can consume for RAN monitoring such as RAN node congestion state, namely *C*, and actions it can take, namely A. These two dimensions determine the dynamic behavior of the RANC. Moreover, the control plane topology *(S* set of connected RAN nodes, *M* set of connected MBH controllers, and *L* set of connected other RANCs) is also a fundamental component of system configuration.

RANC design is shown in *Figure 10*. It consists of the following modules:

- *Connectors*: This part is responsible for inbound/outbound communications such as connection management and interfacing different peers.

- *Data adapters*: This module converts the output from modules for connector consumption and the input from external entities for upper modules.

- *RRM Engine*: RRM engine monitors the resource allocation and interference in the radio environment

- *QoS Analyzer*: It examines the radio and system related inputs from RAN nodes and monitors QoS.

- *Mobility*: Mobility module manages mobility related decisions such as handover decisions or restricted mobility where a user is not allowed to roam outside a specific zone

- *Radio Environment Map*: This module fuses the different radio environment information coming from different RAN nodes and constructs a REM for the zone controlled by RANC.

- *Decision Logic*: Main algorithmic components are executed in this module and it initiates the control actions according to input from other modules.

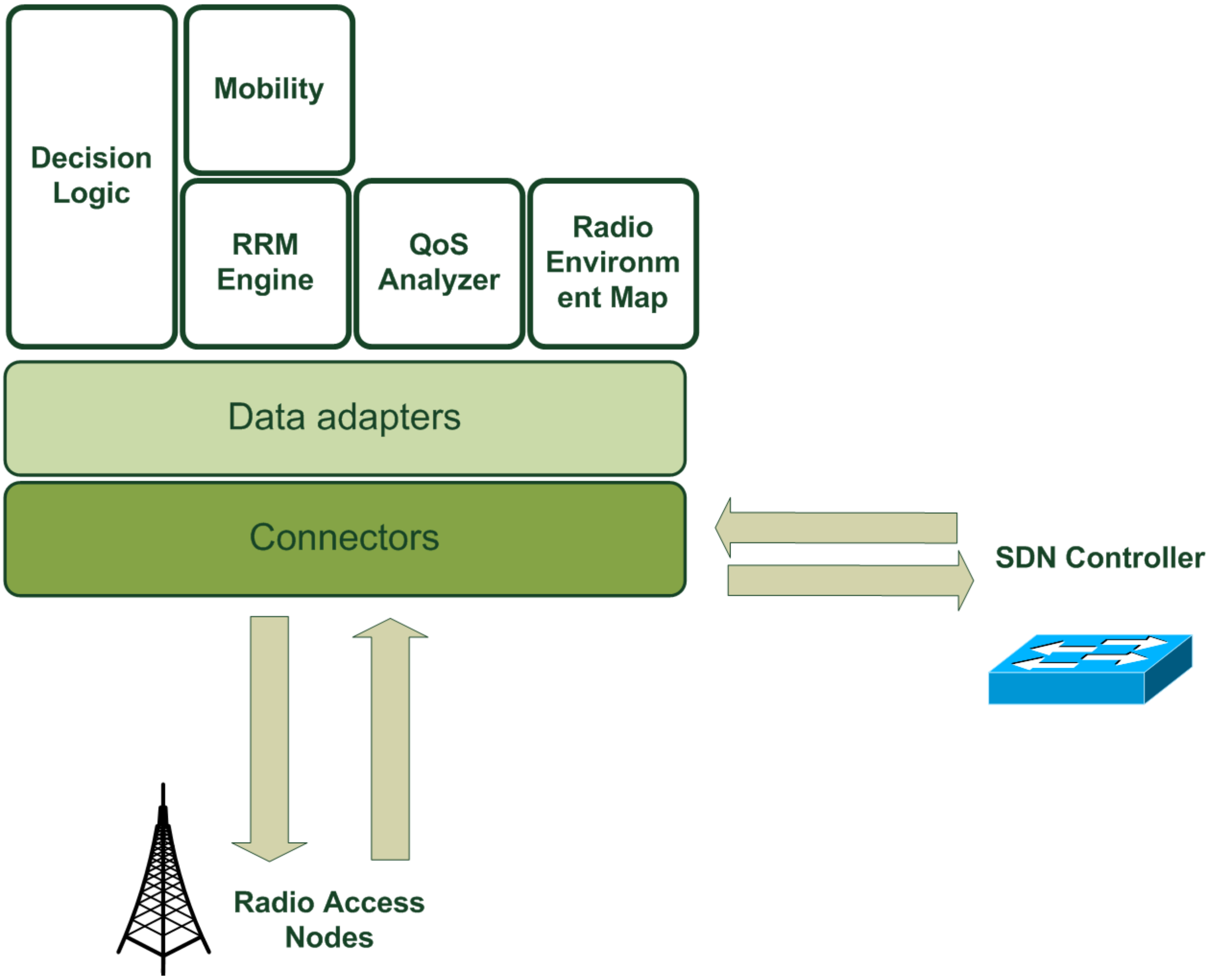


Figure 10: RANC design

3.3 RAN management

Local RANC periodically collects data from RAN elements about CQI, congestion, traffic, interference, etc. The time scale of data gathering period Tc is around seconds. However, this system parameter is flexible and can be altered according to the dynamism of the RAN as shown in Table 1. As shown in Figure 9, the duplex interconnection between the RAN controller and RAN network elements through the southbound interface layer (RIA) provides the monitoring and control mechanisms for resource management and radio-specific functions, namely ICIC and CoMP. The RIA protocol between RANC and the network elements (i.e. eNodeB, switch or router) is used to monitor the resources in the RAN network elements as well as radio optimization handling enforced from the controller.

RANC also interfaces backhaul controller via the interconnection between RANC and the SDN-Controller through a northbound API layer (RASCI). This interface allows the mobile backhaul controller to improve its operation according to RAN state. Moreover, it provides the means to MBH controller to control RAN Controller inline with its position in the controller hierarchy.

Scalability is an important issue for RANC. For RIA and RASCI, scalability issues emerge in three major dimensions:

1. Control data scalability: More advanced control functionalities and distributed architectures rely on more data exchange and "data deluge" may occur.

2. Computational scalability: The computation and data processing for inbound and outbound data and decision logic may become over-complex due to complexity of the system.

3. Communication scalability: Number of RAN nodes controlled by the local RANC may have scalability issues due to communication overhead and delay. The radio related functions are pushed to the edge (radio access nodes) in Evolved Packet System (EPS) in LTE to optimize delay performance under dynamic changes in wireless environment. The centralization of those functions in SDMN is another challenge for delay performance.

The RAN management via a controller plane consisting of SDN-inspired controller exhibit the fundamental trade-off of distributed vs. centralized architectures. Distributed architectures provide resilient and delay-minimized operation while they also suffer from synchronization and communication overheads. Meanwhile, centralized systems exploit global view of the system and can enable more concentrated optimization for performance objectives. However, they are more delay-prone and may suffer from resilience issues due to single-point-of-failure situations.

3.4 Network Virtualization for Mobile Operators using QoS-Aware Schedulers in SDN based LTE Networks

SDN provides powerful and simple approach to manage complex networks, by creating programmable, dynamic and flexible architecture, abstraction from hardware and centralized controller structure. In addition to SDN, network virtualization is another important paradigm for using network resources efficiently. It can provide features such as sharing of resources for breaking up large resource into multiple pieces [RJ13], isolation for protection from other tenants that are sharing the same resources, aggregation for combining many resources into one, dynamism for fast deployment, scalability and mobility support and easy of management for deployment, testing purposes. Relatively, resource allocation plays a fundamental role to meet QoS requirements of user's applications. Depending on the application types (voice over IP, video conferencing, streaming media etc.), the requirements differ and can be mapped to common parameters such as minimum guaranteed data rate, transmission delay, jitter and packet loss rate.

Network virtualization provides many benefits including decoupling of physical and virtual networks and multi-tenancy. In this section we introduce an SDN based virtualization controller architecture for mobile backhaul and radio access network (RAN) sharing. In the developed architecture, satisfied-user ratio, achievable maximum data rate and fairness performances of QoS-aware resource allocation algorithms which are max-min fairness and rate guarantee schedulers are investigated. The main novelty can be summarized as follows:

- An SDN-based Evolved Packet System (EPS) architecture is proposed that can supply benefit to not only MNOs, but also to Backhaul Transport Providers (BTPs) who are responsible for setting up and maintaining Evolved Packet Core (EPC), backhaul and RAN.

- Using the proposed architecture, different QoS-aware scheduling algorithms are investigated in terms of multiple MNO’s satisfaction rates based on their varying demands. Achievable maximum data rate and fairness issues are considered as well.

3.4.1 Use case: virtualization controller for mobile backhaul and ran sharing

In this scenario (see Figure 11), a network virtualization controller that is directly connected to the SDN controllers of each MNO is used to adaptively perform resource sharing between different MNOs. In this architecture, virtualization is performed in two levels. First, BTPs will manage the network slices assigned to each MNO using network virtualization controller. Second, sub-virtualization for all MNO’s applications can be performed within a mobile operator’s slice. In this SDN-based EPS architecture, traffic of multiple MNOs is converged to run on a common network infrastructure while each stream of mobile operators is kept virtually separate. Note that using this scenario, sharing of the network by multiple MNOs results in lower capital expenses and operational costs.

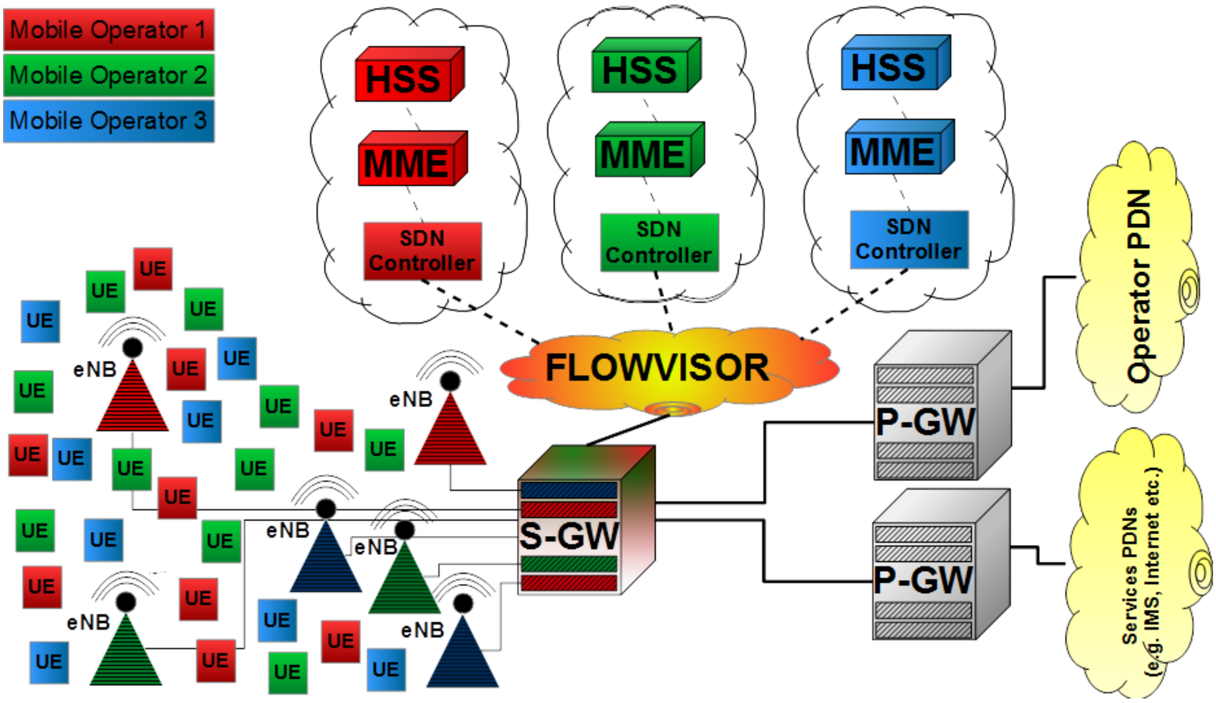


Figure 11: The shared SDN based LTE mobile architecture.

The network virtualization readily applies to the provisioning of a shared EPS network where the streams of different MNOs are isolated from one another and each MNO can control its own allocated slice of the network without any regard to the other MNOs sharing the network. The network slices allocated to the individual MNOs is managed by the BTP via the Virtualization Controller (e.g. FlowVisor). The SDN framework allows for the BTP to act as a broker in this setting to modify and adapt the slices in real time based on the agreements between the BTP and the MNOs.

SDN allows adaptive virtualization based on different scenarios including topology, hardware, device CPU and bandwidth of the individual links with priority settings within the network amongst MNOs. The individual MNOs can then control their own slices via their dedicated control plane architectures (i.e. via their own Mobility Management Entity (MME), Home Subscriber Station (HSS) and Policy and Charging Rules Function (PCRF)). Every time a new rule needs to be pushed by MNOs controller, the virtualization controller first checks the integrity and validity of the rule and then forwards the rule to the corresponding forwarders in the network.

The SDN framework with the virtualization controller allows all nodes, including the network forwarding hardware and network gateways (Serving Gateways (SGWs) and Packet Data Gateways (PGWs)), packet data networks (PDNs) and backhaul to be shared by the MNOs. It also provides granularity in what is shared in the network. In the shared network, the MNOs may maintain their own evolved NodeBs (eNodeB), gateways, and PDNs and they may also share some of the gateway elements and PDNs. All MNOs participating in the shared network maintain their own control plane (MME, PCRF and HSS) and this is used to control the network slice they are allocated by the virtualization controller which is maintained by the BTP.

3.4.2 Scheduling algorithms

In the proposed framework the programmable radio resources can be distributed among the users belonging to different MNOs based on different radio scheduling algorithms. The distribution of the backhaul resources between the MNOs shall follow the radio resources distribution.

In literature, several resource allocation algorithms exist. Their performance and complexity vary depending on the input parameters and calculation mechanisms [FC13]. Schedulers distribute the resources to users according to their allocation mechanisms which basically allocates k^{th} resource to j^{th} user, if its metric (m_j) is the biggest one such as $m_j = \mathrm{argmax}_i\{m_i\}$. In general, schedulers may be classified into mainly three categories, namely, channel-unaware, channel-aware/QoS-unaware and channel-aware/QoS-aware.

Channel-unaware schedulers, (i.e. round robin which has fair allocation mechanism serving in a cyclic order) use simple algorithms to allocate the resource to users without considering channel state information or QoS requirements. In contrast, channel-aware schedulers require channel state information which is periodically provided by channel quality indicator reporting [FC13]. Maximum throughput and proportional fair schedulers fall into this category. Maximum throughput scheduler considers instantaneous achievable data rate $(R(t))$ as metric and the analytical expression of the algorithm can be written as $m_j = \mathrm{argmax}_i\{R_i(t)\}$. It maximizes the total throughput of the system via providing user diversity gain. However, it is totally unfair. On the other hand, proportional fair scheduler partially satisfies both system throughput and fairness among users. Proportional fair uses past average throughput of individual users as weighting factor while allocating resources for next transmission time interval (TTI) period. The metric of proportional fair can be written as $R_i(t)/\lambda_i(t)$ where $\lambda_i(t)$ denotes average throughput of i^{th} user and calculated by $(1 - 1/\tau)\lambda_i(t - \Delta_t) + R_i(t)/\tau$ where τ is time constant of smooth filter and Δ_t is TTI period. However, none of maximum throughput and proportional fair schedulers considers QoS requirements.

In general, QoS schedulers [FC13] use the metric defined as $R_i(t)/U_i'\big(\lambda_i(t)\big)$, where $U_i\big(\lambda_i(t)\big)$, represents utility functions, provided by [PH02]. For rate guarantee scheduler, defined in [FK97], utility function for QoS-aware users is designed as

$$U(\lambda_i) = \lambda_{i_{\min}} \left(\log(\lambda_i) + 1 - e^{\left(-\beta_i \frac{\lambda_i - \lambda_{i_{\min}}}{\lambda_{i_{\min}}}\right)} \right)$$

where β_i is positive constant that controls the aggressiveness with respect to ratio of amount of allocated sources and demands. Another QoS-aware scheduler is max-min fairness algorithm [KS07], in which resources are allocated to users in the order of their increasing demands (i.e., data rates) and unsatisfied users are equally allocated with remaining resource. In [ON16] the system level performance of the max-min fairness and rate guarantee scheduling operation has been investigated, please refer to that paper for detailed results.

4. Mobile network management

4.1 Network- and resource management framework

The network- and resource management proposed from SDMNs is shown in Figure 12. First, the general concept of the framework is described, and then different deployment scenarios are given. The framework uses the service of the underlying SDN/virtualized mobile network. The virtualization layer may not exist in all segments, thus the framework has to be able to cooperate with legacy network segments and management solutions too. It collects information relevant for network- and resource management like the request originated by the different mobile network management functionalities (e.g. mobility management requests), actions taken by the different segment specific optimization process, external demands, i.e. demands arriving to the mobile network operator, etc. It also responsible for accepting the resource demands, optimization requests automatically derived from the mobile operator policies or accepting optimization requests generated by the human network operator.

Based on the received and collected information the framework has three main responsibilities:

- Consolidation of the request/triggers/demands received from either in-network functionalities or from external sources.
- Make resource allocations/re-optimization based on the consolidated demand database.
- Coordinating the optimization activities performed in the SDMN.

These tasks should be performed by the framework taking into account information from Business Intelligence/Business Support System/Operation Support System tools. This makes it possible to evaluate the proposed optimization/resource management decisions form cost/business point of view too and thus, only perform actions that are proven to be feasible both from technical and economical point of view.

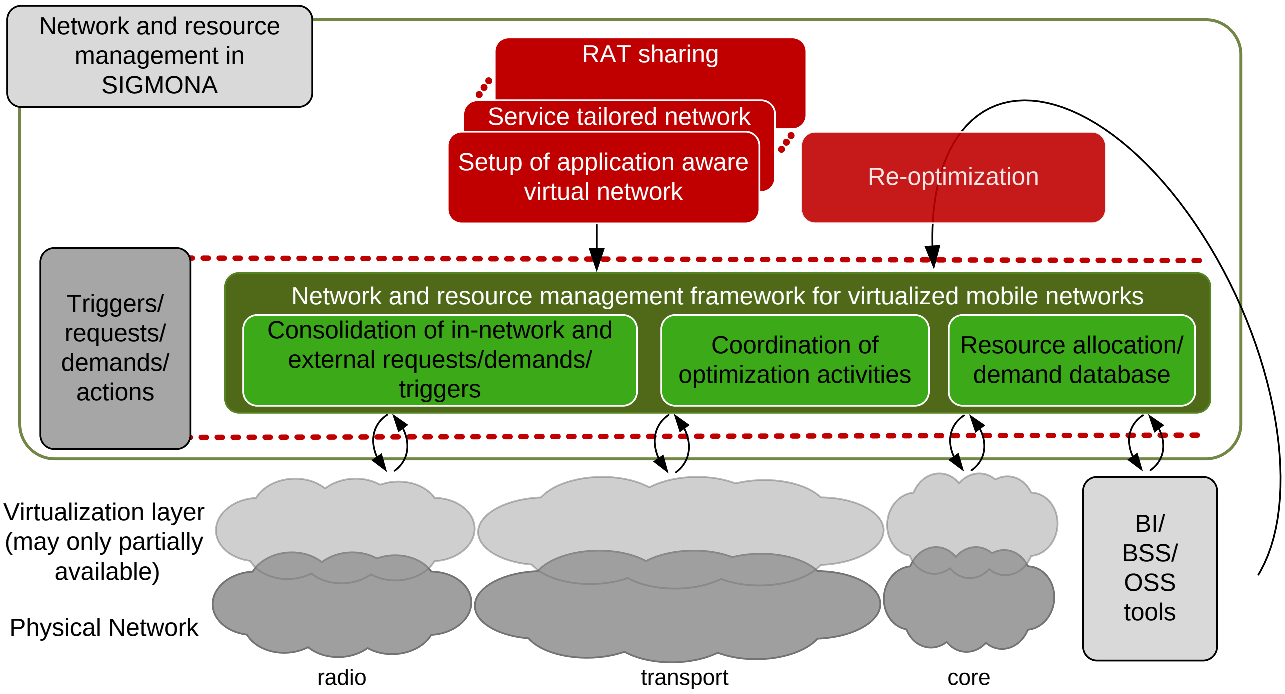

Network and resource management in SIGMONA
RAT sharing
Service tailored network
Setup of application aware virtual network
Re-optimization
Triggers/ requests/ demands/ actions
Network and resource management framework for virtualized mobile networks
Consolidation of in-network and external requests/demands/ triggers
Coordination of optimization activities
Resource allocation/ demand database
Virtualization layer (may only partially available)
Physical Network
BI/ BSS/ OSS tools
radio
transport
core

Figure 12: Network- and resource management framework in SDMNs

There are three deployment alternatives for the network- and resource management framework:

- Centralized deployment (Figure 13)
- Distributed deployment (Figure 14)
- Hybrid deployment (Figure 15)

In the centralized case the framework is deployed to the Mobile network operating system. This provides a fully centralized view on the network and resource status of the mobile network. Such centralization, however also raises scalability issues and also requires harmonization with low level or independent optimization processes running in the different network segments. E.g. radio SON functions may perform optimizations which are contrary to the actions of the framework. For those cases the necessary interfaces have to be designed.

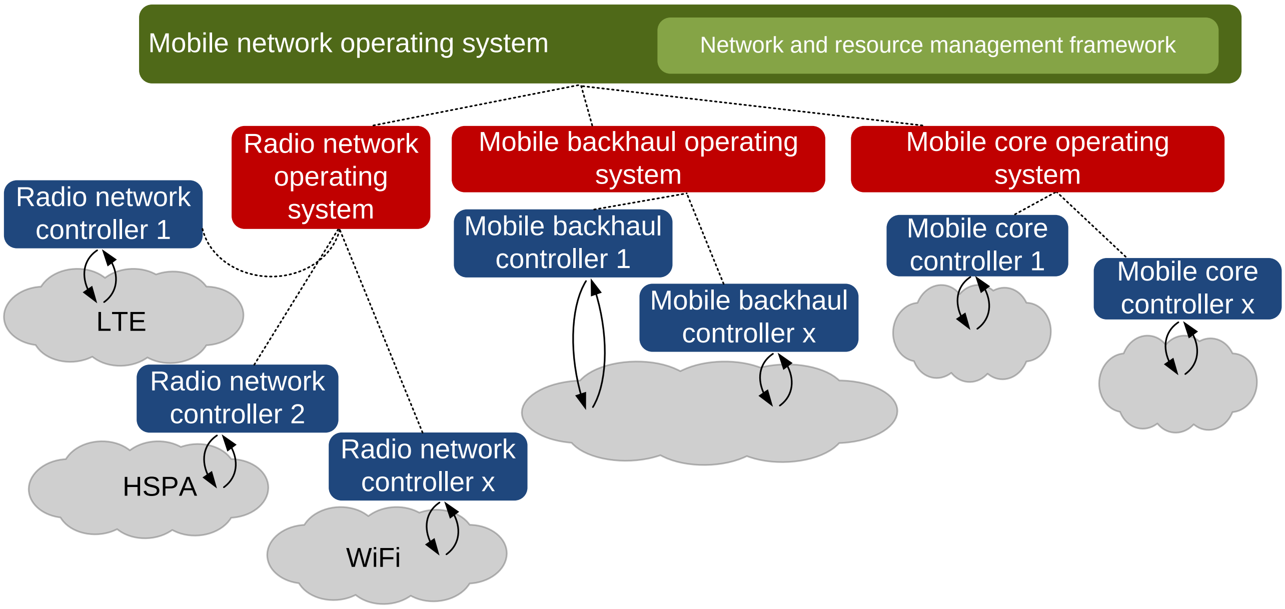

Mobile network operating system
Network and resource management framework
Radio network operating system
Mobile backhaul operating system
Mobile core operating system
Radio network controller 1
LTE
Radio network controller 2
HSPA
Radio network controller x
WiFi
Mobile backhaul controller 1
Mobile backhaul controller x
Mobile core controller 1
Mobile core controller x

Figure 13: Centralized deployment of the framework

In the distributed case the network- and resource management framework is deployed in the network segment specific operating systems, i.e. in the radio, backhaul and core operation systems. This increases the scalability of the solution, however the coordination of the network specific optimization process has to be implemented; this can be done either by introducing horizontal interface between the different network segment specific operation systems or by implementing cross domain communication via the mobile network operation system. Anyway, the price of such solutions is the added complexity due to additional interfaces and limitations on the optimization potentials introduced by the cross domain communication needs.

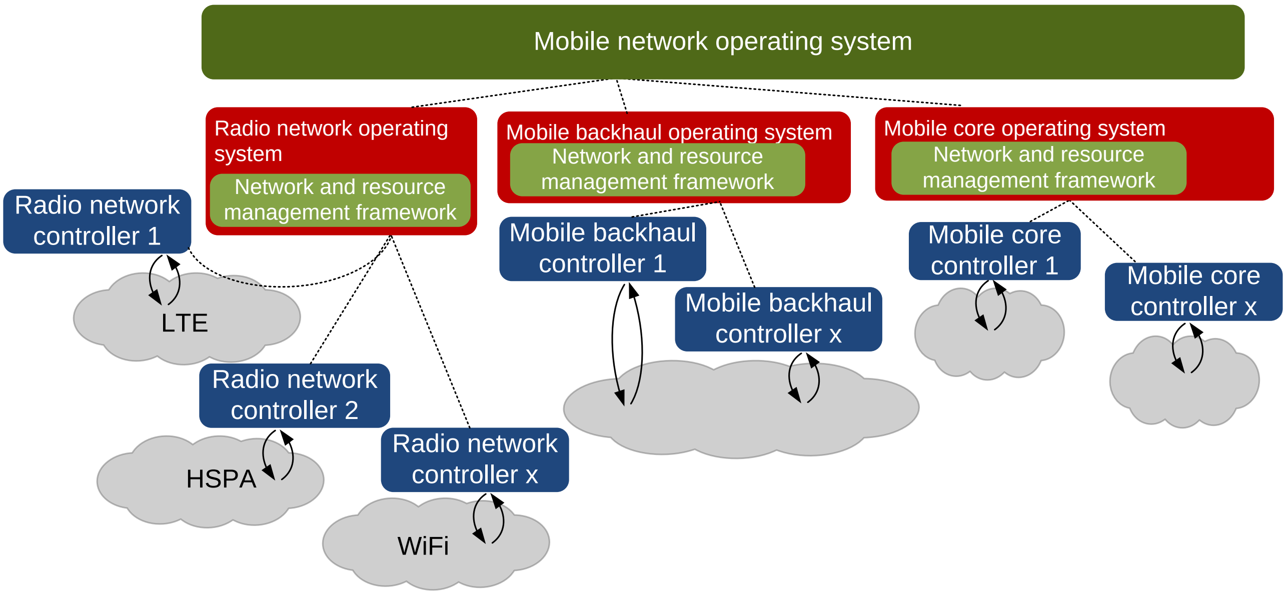

Mobile network operating system
Radio network operating system
Network and resource management framework
Mobile backhaul operating system
Network and resource management framework
Mobile core operating system
Network and resource management framework
Radio network controller 1
LTE
Radio network controller 2
HSPA
Radio network controller x
WiFi
Mobile backhaul controller 1
Mobile backhaul controller x
Mobile core controller 1
Mobile core controller x

Figure 14: Distributed deployment of the framework

The hybrid deployment merges the previous two approaches, the network- and resource management framework is deployed both in the mobile network operating system and in the network segment specific operating systems. In this design it should be selected carefully how the internal functionalities of the framework is split between the different layers. It is proposed to follow the already mentioned approach ("Centralize what you can and distribute what you must"), i.e. functionalities which do not cause scalability issues if centralized should be implemented in the mobile network operating system whereas complex, high "state processing" requirements (that require to store a lot of state information during its operation) should be implemented in the network domain specific operation systems.

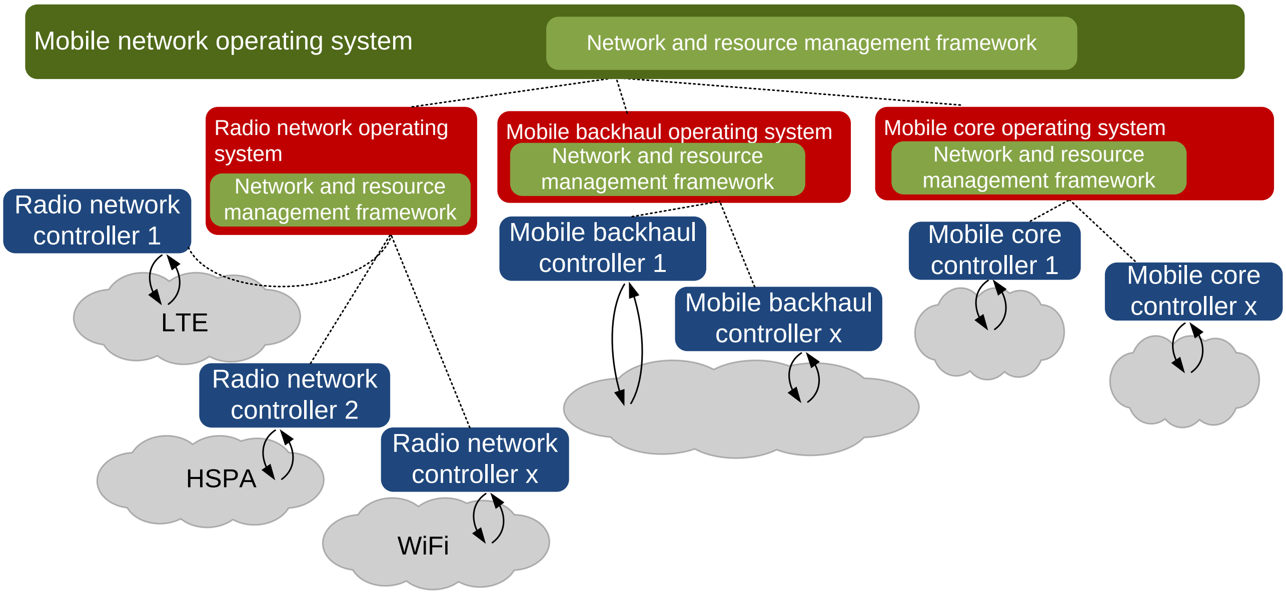

Mobile network operating system
Network and resource management framework
Radio network operating system
Network and resource management framework
Mobile backhaul operating system
Network and resource management framework
Mobile core operating system
Network and resource management framework
Radio network controller 1
LTE
Radio network controller 2
HSPA
Radio network controller x
WiFi
Mobile backhaul controller 1
Mobile backhaul controller x
Mobile core controller 1
Mobile core controller x

Figure 15: Hybrid deployment of the framework

4.2 Concept for harmonized network and resource management

This section will describe the HNRM concept in detail; Figure 16 shows its building blocks. The concept is described for LTE network, however it is straightforward to apply it to other mobile network types like HSPA or to cases where the network operator owns/operates several networks, e.g. LTE, HSPA and WiFi and uses HNRM to manage them.

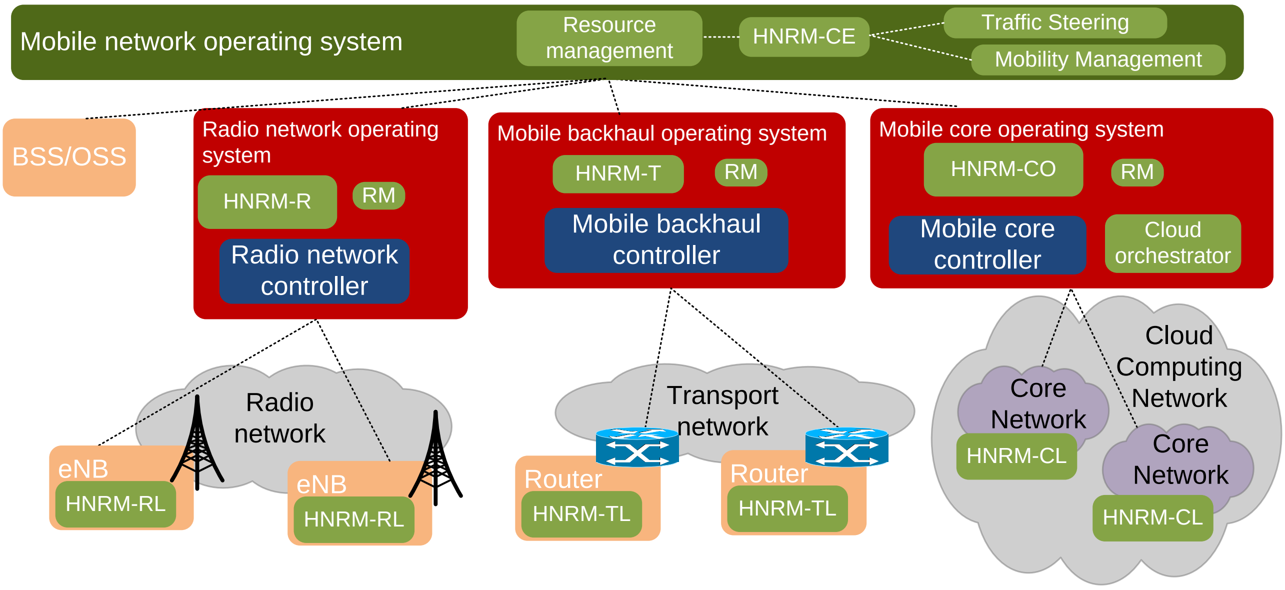


Figure 16: The HNRM concept

For the HNRM concept we decided to use the hybrid deployment model (described in section 4.1 and shown on Figure 15). The reason for this design choice is that it gives the best possibilities for controlling the network and also it provides the best architecture for handling locally what does not have to be handled centrally and thus has the best scalability.

The operation of the HNRM is distributed on three levels:

- Mobile network operating system
- Mobile network domain specific operating systems
- Mobile network device elements

The next subsections will describe how the operation of the HNRM is distributed between those levels and how they interact.

4.2.1 Mobile network operating system

In SDMN, the mobile network operating system (MNOS) is responsible for the mobile network wide management. It oversees the operation and status of the different mobile network domains, i.e. radio access network, mobile backhaul network and the core network, and manages their operation via the domain specific operating systems.

The MNOS hosts the HNRM-CE (HNRM-Central) entity. This entity is responsible for orchestrating the operation of the mobile domain network specific HNRM entities (described in section 4.2.2); it has the centralized view of the HNRM operation. It is also connected to three main mobile network management functionalities:

- Mobile network resource management (RM)
- Mobility management (MM)
- Traffic Steering (TS)

In case of basic integration of the HNRM and the listed management functionalities, the HNRM has the responsibility to harmonize its operation with the functionalities. For this, it is required that it can gain relevant information regarding the internal operation of the RM, MM and TS. In case of tighter integration the HNRM shall be able to influence or even to control the operation of the RM, MM and TS.

HNRM is also connected to the BSS/OSS (Business Support System/Operation Support System) of the mobile operator. The HNRM collects information from those systems in order to evaluate if an optimization action can be justified from cost/revenue, user experience, user impact point of view, e.g. prioritize optimizations for base stations/area where it is economically more favourable.

4.2.2 Mobile network domain specific operating systems

Mobile networks can be divided into three network domains: radio network, mobile backhaul and core network. In SDMN, each domain can have its own network operating system. The next subsections will describe how the HNRM entities are mapped to these domain specific operating systems.

4.2.2.1 Radio network operating system

The radio network operating system (RNOS) is responsible for controlling and managing the radio network it is assigned to. It main entities relevant for HNMR are:

- radio network controller
- resource management

The radio network controller has the means to control and manage the operation of the radio network while resource management is responsible for managing the radio resources. The RNOS also includes the HNRM-Radio (HNRM-R) entity which is responsible for managing radio resource related operations and optimizations.

There are two scenarios for the RNOS and radio access technology (RAT) assignment: either the operator has a separate RNOS for all the RATs it is operating or integrates the management of RATs into one RNOS. In the latter case the RNOS includes multiple radio network controllers and radio resource management entities whereas in the former case the RNOS has only one controller. The two alternatives are shown by Figure 17 and Figure 18.

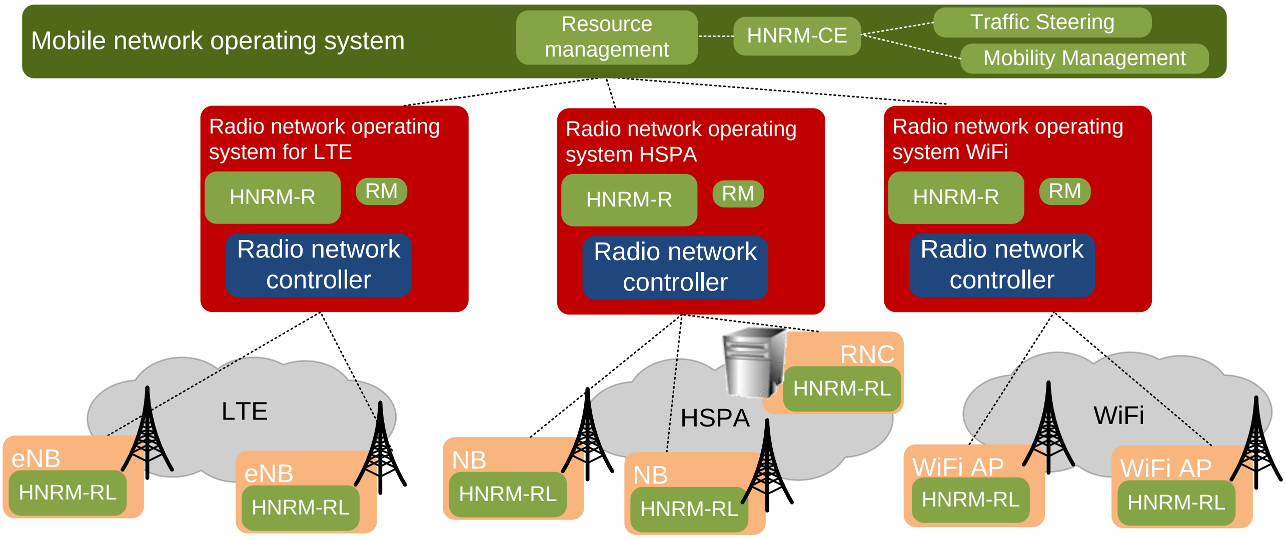


Figure 17: Separate RNOS for different RATs

In case there are separate RNOSs for the RATs managed by the operator, then HNRM-R residing in the RNOS can control and manage the optimization of the assigned radio network only, thus inter-RAT optimizations have to be delegated to the HNRM-CE.

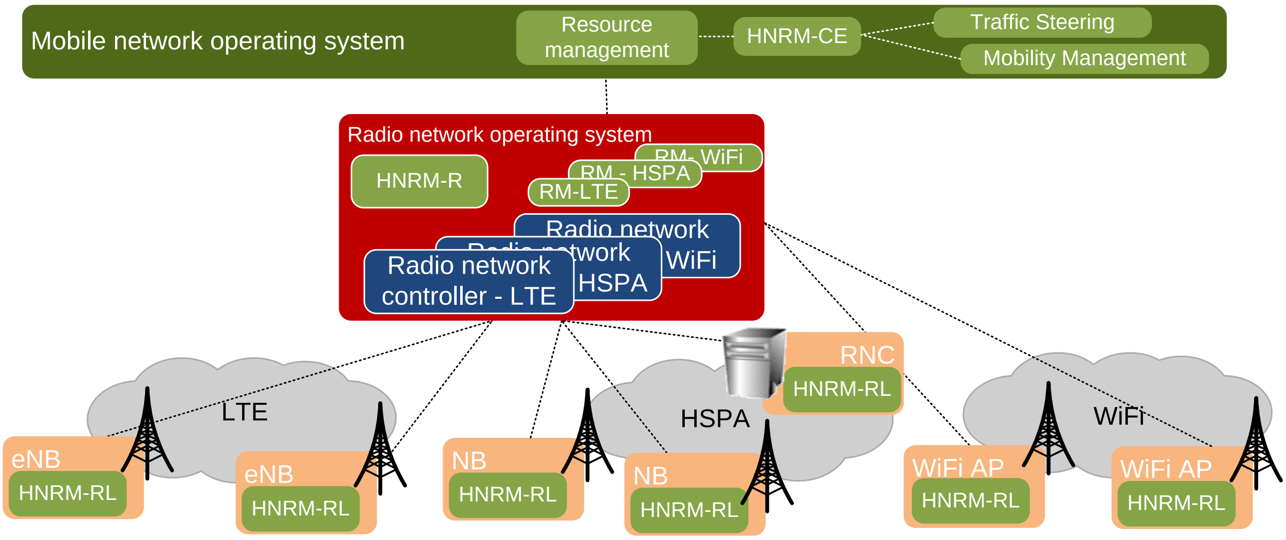


Figure 18: Joint RNOS for different RATs

If the operator integrates the radio network controllers and radio resource management of the different RATs into one RNOS, then a single HNRM-R entity can control and manage the resources and optimizations of the RATs jointly, thus contrary to the previous case the HNRM-R can handle the inter-RAT activities locally and only those radio optimizations have to be delegated to the HNRM-CE which require insight into backhaul and/or core network operation.

4.2.2.2 Mobile backhaul operating system

The mobile backhaul network operating system (MBHNOS) is responsible for controlling and managing the mobile backhaul connecting the radio and core network. It main entities relevant for HNMR are:

- mobile backhaul controller
- resource management

The mobile backhaul controller is responsible for controlling the operation of the mobile backhaul devices while resource management's scope is to manage the backhaul resources (e.g. path capacity). The MBHNOS also includes the HNRM-Transport (HNRM-T) which is responsible for mobile backhaul resource related operations and optimizations. There can be two scenarios for implementing the MBHNOS that influences the HNRM operation also.

In the first scenario, the mobile operator has a different MBHNOS for all network part composed by the same equipment vendor (Figure 19). This might be required or seen as appealing by the operator as there are already vendor specific SDN controllers (having some vendor specific features on top of the standardized OpenFlow specification – e.g. Cisco introduced the eXtensible Network Controller [2]) and the separate MBHNOS for the specific SDN controllers can better exploit the vendor specific features. In this case there is a HNRM-T entity in each of the MBHNOS-s. That means that for mobile backhaul level optimization some features shall be delegated to the HNRM-CE running in the MNOS. This makes it possible that the optimization actions run by the different MBHNOSs are harmonized.

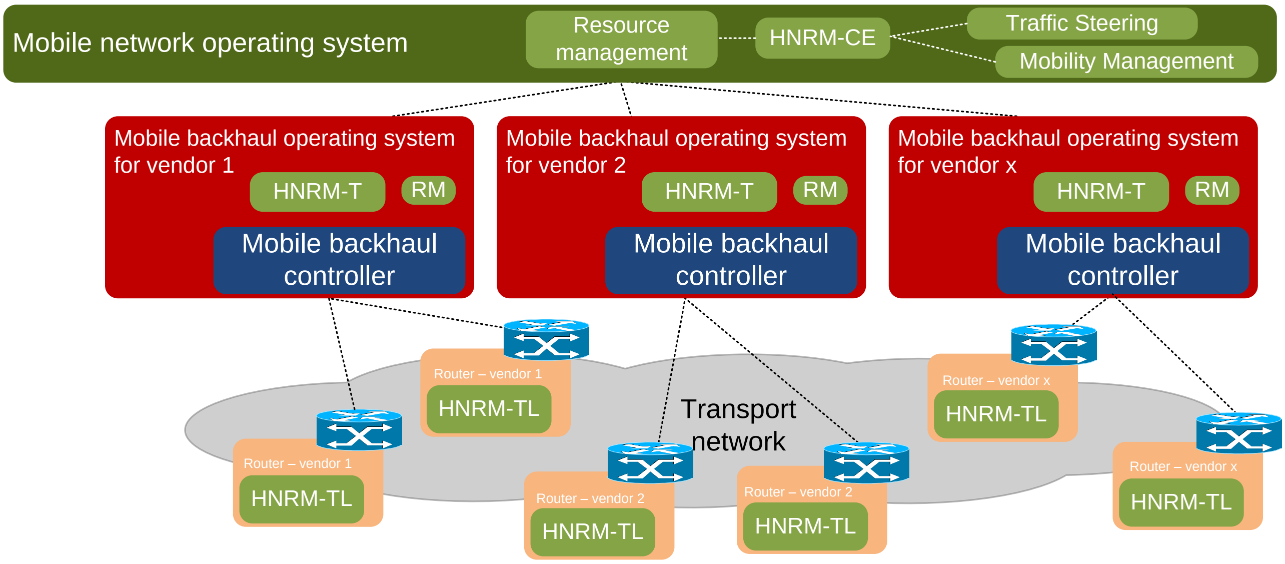


Figure 19: Different MBHNOS for different equipment vendors

Figure 20 shows the second scenario where the mobile operator has one MBHNOS for the whole mobile backhaul. In this case there can be several MBH controllers, each of them controlling the devices of one vendor. There is only one HNRM-T entity is required to run in the MBHNOS and this single one can control and harmonize the resource management and optimizations for the whole mobile backhaul. It needs to cooperate with the HNRM-CE entity only in cases when the its planned actions can have impact on radio or core network also.

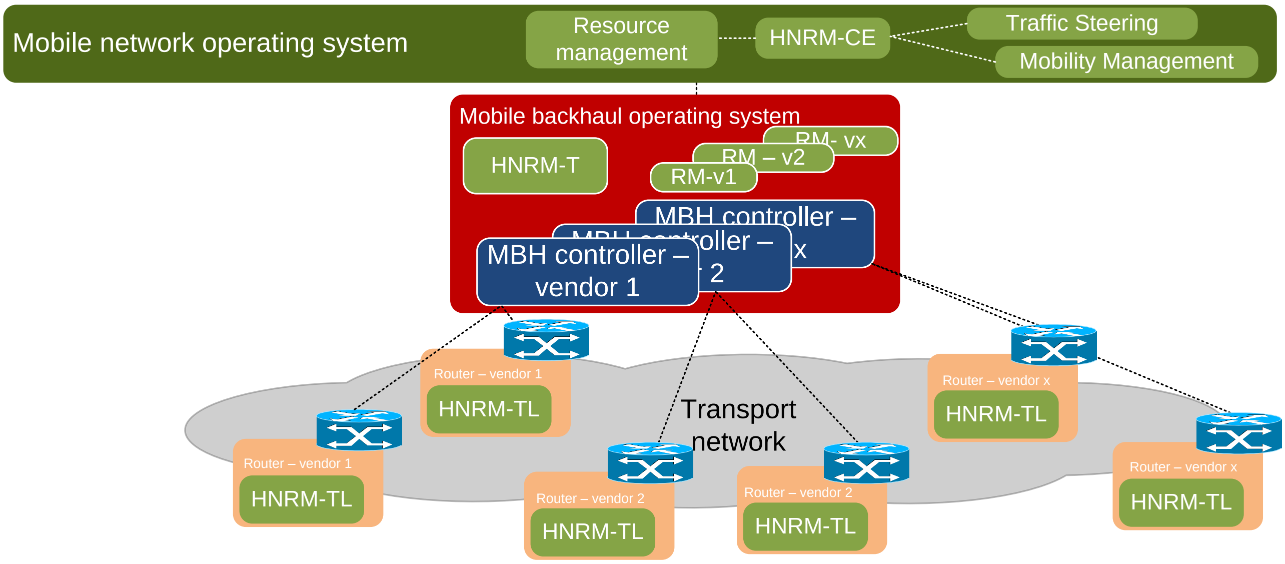


Figure 20: Joint MBHNOS for the whole mobile backhaul

4.2.2.3 Mobile core network operating system

The mobile core network operating system (MCOS) is responsible for controlling and managing the core network operation. It main entities relevant for HNMR are:

- mobile core controller
- cloud orchestrator
- resource management

The mobile core controller is responsible for controlling the operation of the network elements, e.g. SAE-GW, MME, etc. in case of LTE networks. The cloud orchestrator is responsible for providing the necessary cloud (computing, network) resources for the operation of the core functionalities. Resource management is responsible for managing core resources (e.g. MME capacity) so as it can meet the demands of the mobile network. The MCOS includes also the HNRM-CO entity which has the responsibility to manage and harmonize core network related optimizations.

The mobile network operator may decide to run separate MCOS for the different radio access technologies. This case is shown in Figure 21. This requires that each MCOS runs a separate HNRM-CO entity; those entities can harmonize the resource management inside one core network but cannot do it for inter-core network resource management. The latter one requires that those responsibilities are delegated to the HNRM-CE entity running in the MNOS.

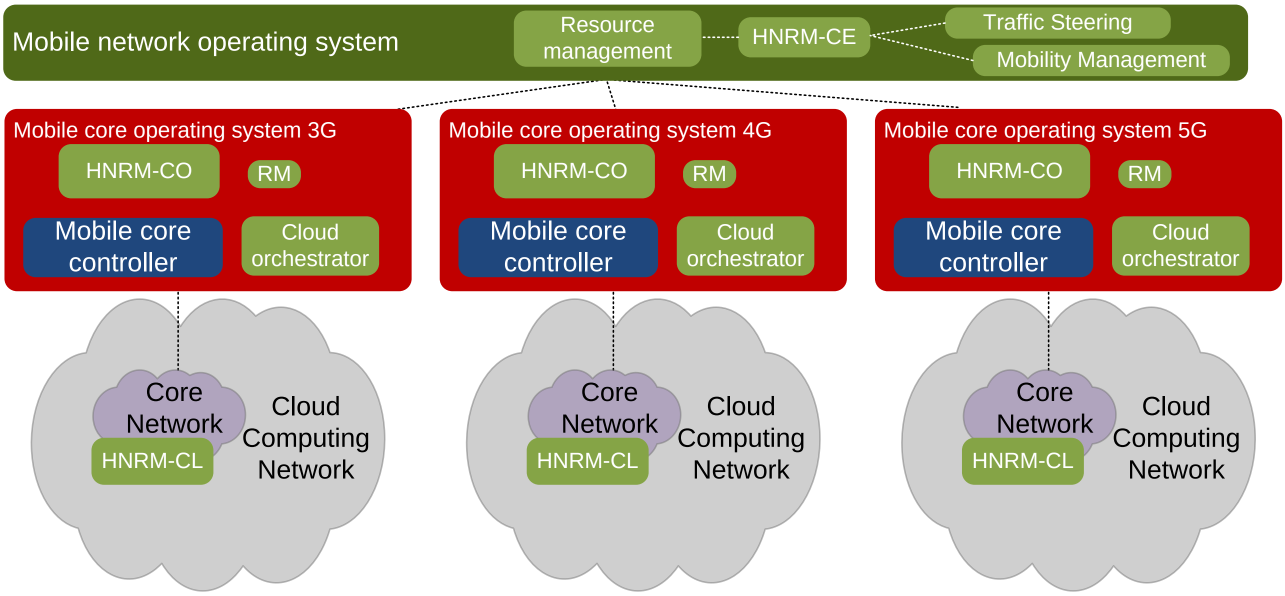


Figure 21: Separate MCOS for different RAT cores

The alternative choice of the mobile network operator can be that it runs only one MCOS for all the radio access technologies it is operating. In this case there are several mobile core controllers in one MCOS and the HNMR-CO is able to do inter-core harmonization also, thus this responsibility does not have to be delegated to the HNRM-CE. HNRM-CE has to be involved only in those optimizations that have impact on the mobile backhaul or on the radio network.

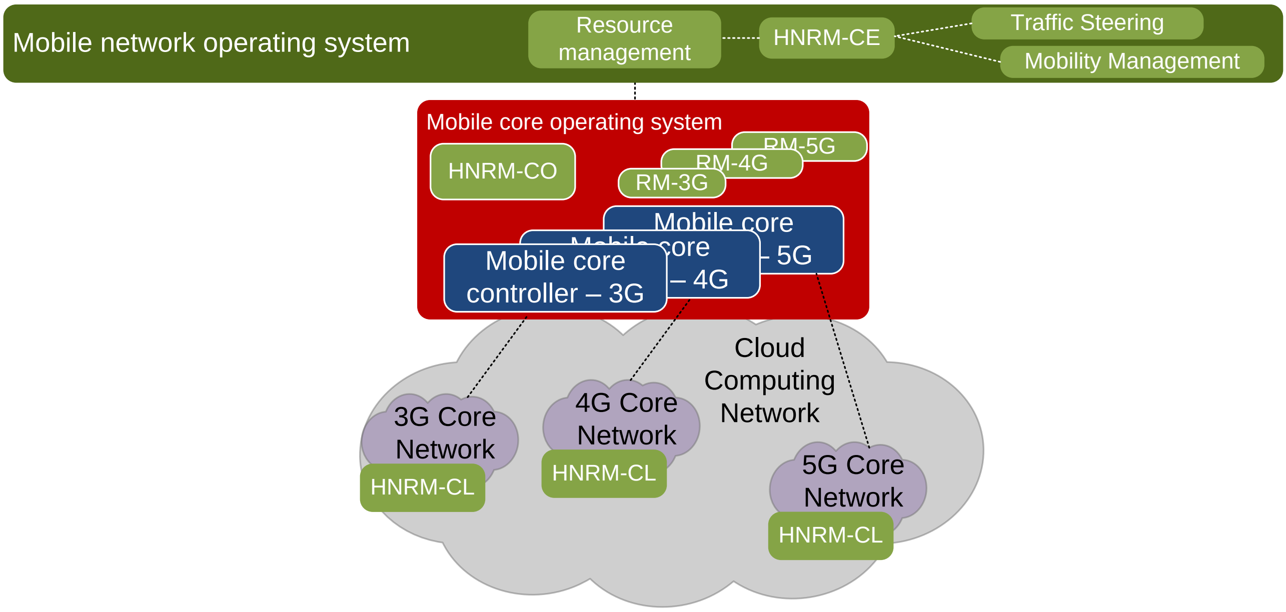


Figure 22: Joint MCOS for different RAT cores

4.2.3 Mobile network device elements

The lowest level entities in the HNRM concept are the entities running on the network devices. There are three types of HNRM devices on this level (see also Figure 16):

- HNRM-RL (HNRM Radio Local)
- HNRM-TL (HNRM Transport Local)
- HNRM-CL (HNRM Core Local)

These entities are running on the mobile network devices and have responsibilities with local scope. They should carry out measurements, provide information for upper layer HNRM entities, and perform local optimizations. Their operation is further detailed in the next section.

4.2.4 General operation of HNRM entities

This section describes the general operation of the HNRM entities on the different operation levels. Figure 23 shows the main functionalities of the entities.

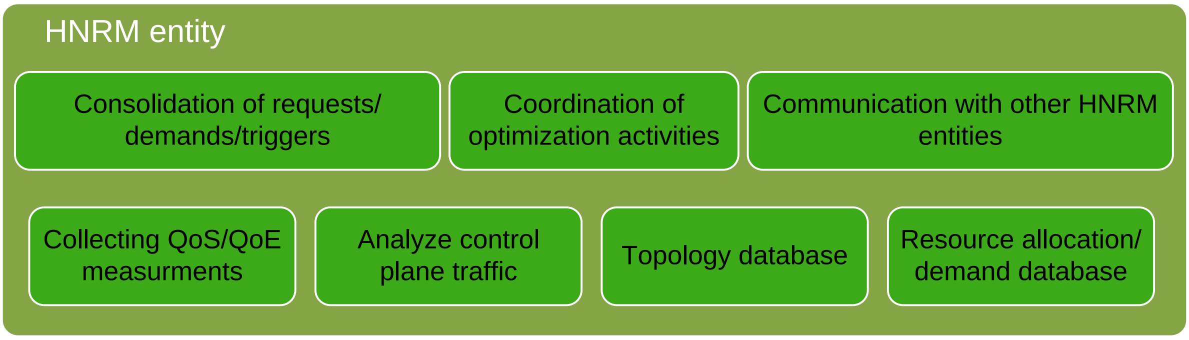

HNRM entity
Consolidation of requests/ demands/triggers
Coordination of optimization activities
Communication with other HNRM entities
Collecting QoS/QoE measurments
Analyze control plane traffic
Topology database
Resource allocation/ demand database

Figure 23: The main functionalities of the HNRM entity

The entities are collecting QoS/QoE measurements on different aggregation levels (e.g. per flow, per traffic class, etc), they analyze control plane traffic; they also communicate with other HNRM entities (vertically and horizontally too) this enables them to evaluate the status of the system belonging to their responsibility (i.e. network device, network domain, mobile network). The entities have two databases: topology and resource allocation/demand ones. The topology database stores the network topology relevant for the HNRM entity, i.e. the HNRM entities in the different levels have different view on the topology, i.e. a HNRM-RL entity might know only the neighboring base stations whereas the HNMR-R entity knows the whole radio topology. Similarly, in the resource allocation/demand database only that level information is stored which relevant for the operation of the entity, i.e. HNRM-RL entity might have info only about the base station's resource allocation it is running on whereas the HNRM-R entity has information on radio resource usage on radio network level.

Based on the collected information, analysis and available databases has three main functionalities:

- consolidation of requests/demands/triggers
- coordination of optimization activities
- communication with other HNRM entities

An operating system in SMDN (e.g. RNOS) can receive request/triggers from various sources regarding resource management; those requests and triggers can easily be contra dictionary ones or requiring complex optimization actions. The scope of the consolidation of requests/demands/triggers is to filter them so as to eliminate actions which could be totally overlapping, to create an execution order for the actions which is optimal and to resolve contra dictionary optimization demands.

The scope of the coordination of the optimization activities task is to coordinate and harmonize the optimization activities being run by the different management entities on the level of the HNRM entity (e.g. HNRM-R can harmonize eICIC/TS operation).

The communication with other HNRM entities has two directions: horizontal and vertical. In case of horizontal communication HNRM entities running on the same level communicate, e.g. HNRM-RLs in different base stations. In case of vertical communication HNRM entities on different levels are communicating, e.g. HNRM-RL and HNRM-R entities communicate with each other. This is required for HNRM to be able to build a consistent network view and for efficient harmonization of the network and resource management tasks.

4.3 Mapping of HNRM to network architecture models

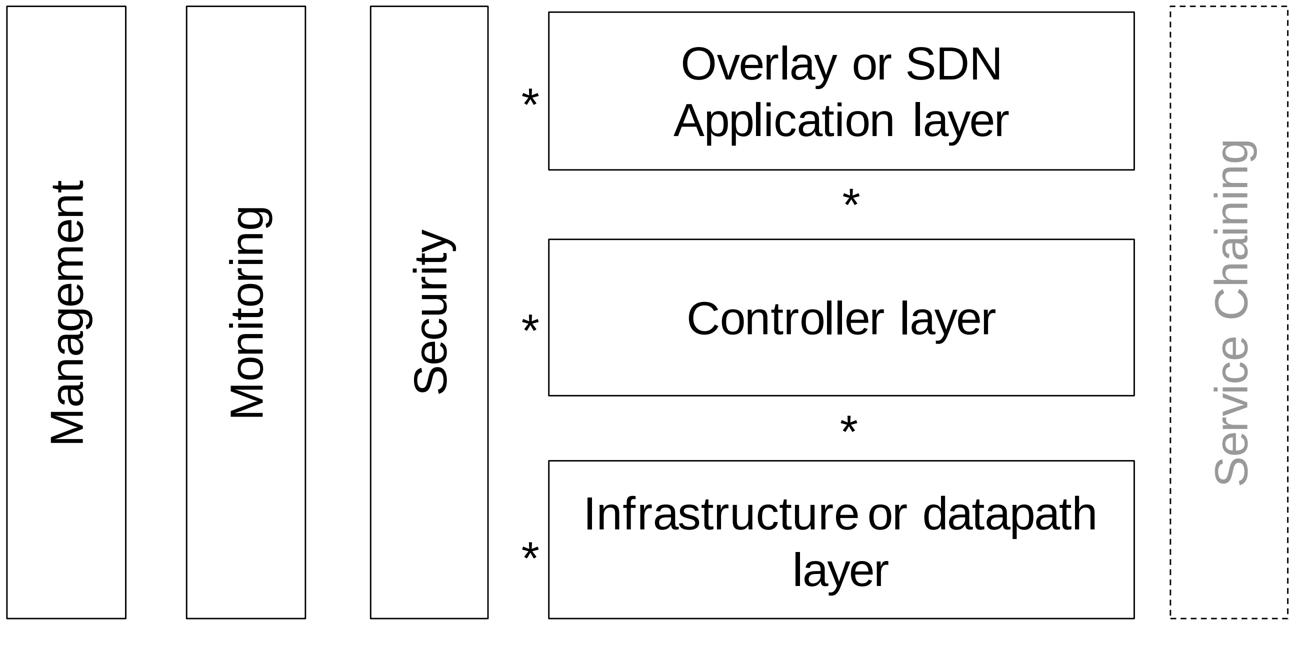
The next subsections will analyze how the HNRM concept can be mapped to different SDMN architecture models. This mapping helps to understand the requirements and the implementation possibilities of the concept.

4.3.1 Mapping of HNRM to layered architecture model

4.3.1.1 Introduction of the layered architecture model

Figure 24 shows the layered network model of a SDMN; it consists of seven layers:

- **SDN application layer**: it represents the applications that are built on top of the services provided by the SDN controllers.
- **Controller layer**: it includes the SDN controllers and their functionalities that present in the network.
- **Infrastructure or datapath layer**: this layer includes the physical and virtual network devices that are present in the network.
- **Security layer**: this includes the security related functionalities in SDMN.
- **Monitoring layer**: this layer is the place for the network monitoring related functionalities in SDMN.
- **Management layer**: this layer is responsible for carrying out mobile network related management activities.
- **Service chaining**: this is representing the solutions with the scope to implement service chaining in a streamlined way (e.g. directing selected traffic via specified services – firewall, content optimization, etc.)



* reference point description

Figure 24: Layered SDMN network model

4.3.1.2 Actual mapping to the model

Figure 25 shows the mapping of the HNRM to the layered network model.

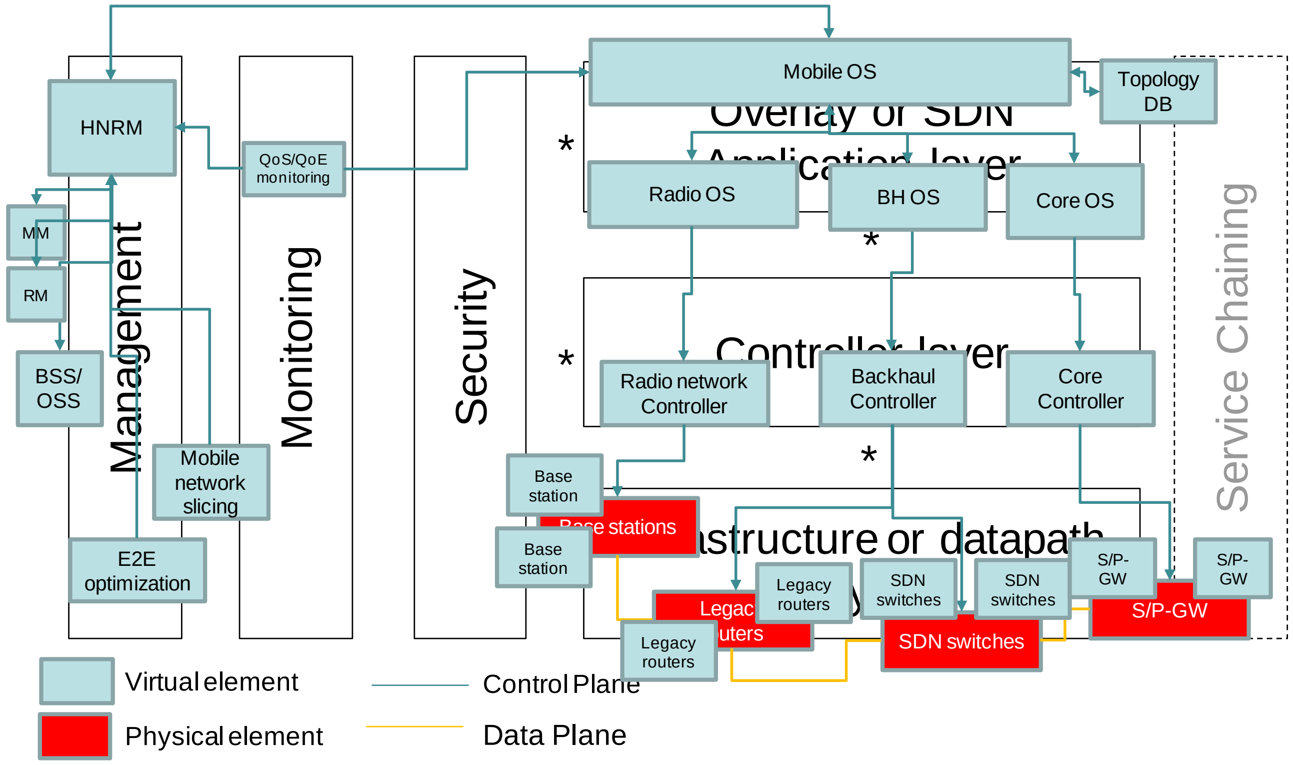


Figure 25: Mapping of the HMNR to the layered network model

The following HNRM related functionalities can be mapped to the SDN Application layer: Mobile Operating System, Radio Operating System, Backhaul Operating System, Core Operating System, topology database.

The following HNRM related functionalities can be mapped to the controller layer: radio network controller, backhaul controller, core controller.

The monitoring layer provides services about the QoS/QoE monitoring of the system on different aggregation levels (e.g. flow/base station/radio access network, etc.).

The following HNRM related functionalities can be mapped to the management layer: Mobility Management, Resource Management, BSS/OSS, Harmonized Network and Resource Management, Mobile Network Slicing, E2E optimization.

The datapath includes legacy and SDN-ready routers and switches. QoS shall be manageable in the datapath.

4.3.2 Mapping to function based architecture model

4.3.2.1 Introduction of the function based architecture model

Figure 26 shows a function based network model for SDMNs. The upper part of the figure shows the core functionalities of the SDMNs.; the bottom part of the figure shows the network elements and most important interfaces between the functions and the network elements.

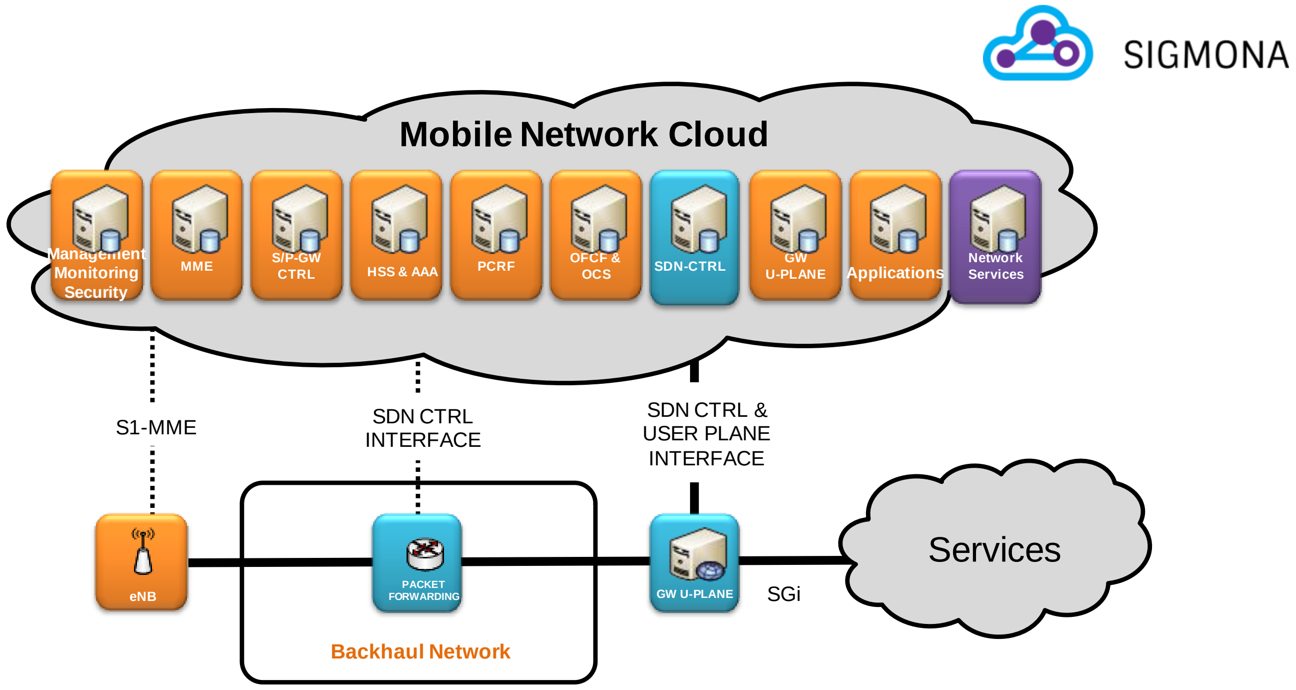


Figure 26: Function based SDMN network model

The MME (Mobility Management Entity), HSS (Home Subscriber System), AAA (Authentication, Authorization and Accounting), PCRF (Policy and Charging Rules Function), OCF (Online Charging Function) & OCS (Online Charging System) are standardized LTE functionalities. The SAE-GW is split into two parts: one (GW U-plane) that is responsible for handling user plane traffic whereas another one (S/P-GW CTRL) is responsible for controlling the operation of the GW U-plane and the latter one is running in the telco cloud. The SDN-CTRL functionality is hosting all the SDN related controller functionalities; the network services function collects the services that the mobile network can provide to upper layers (e.g. load balancing on alternate paths between an eNB and SAE-GW). The Management, Monitoring and Security (MMS) function is collecting all the network management and monitoring related functions in the mobile network.

4.3.2.2 Actual mapping to the model

Figure 27 shows the mapping of HNRM to the function based model. The HNRM can be mapped to the MMS functionality; also the Topology DB, Demand DB and BSS/OSS can be mapped into the MMS. The radio, backhaul and core controllers can be the part of the SDN-CTRL functionality,

Automated mobile network slicing can be thought of as a service of the HNRM; based on the received demands (from the network operator, or from other operators), network measurements and information received from the BSS/OSS systems, it is managing the network slices of the mobile network, thus this can be mapped to the Network services functionality. Another example for the Network service related to the HNRM is E2E optimization; this can be thought of as a service of the HNRM; this service is carrying out optimizations that go beyond the boundary of network domains.

We propose to add there the SDN control interface for the radio networks also, that could provide means for efficient controlling and managing of the radio networks, thus to enable flexible, automated E2E management and optimization.

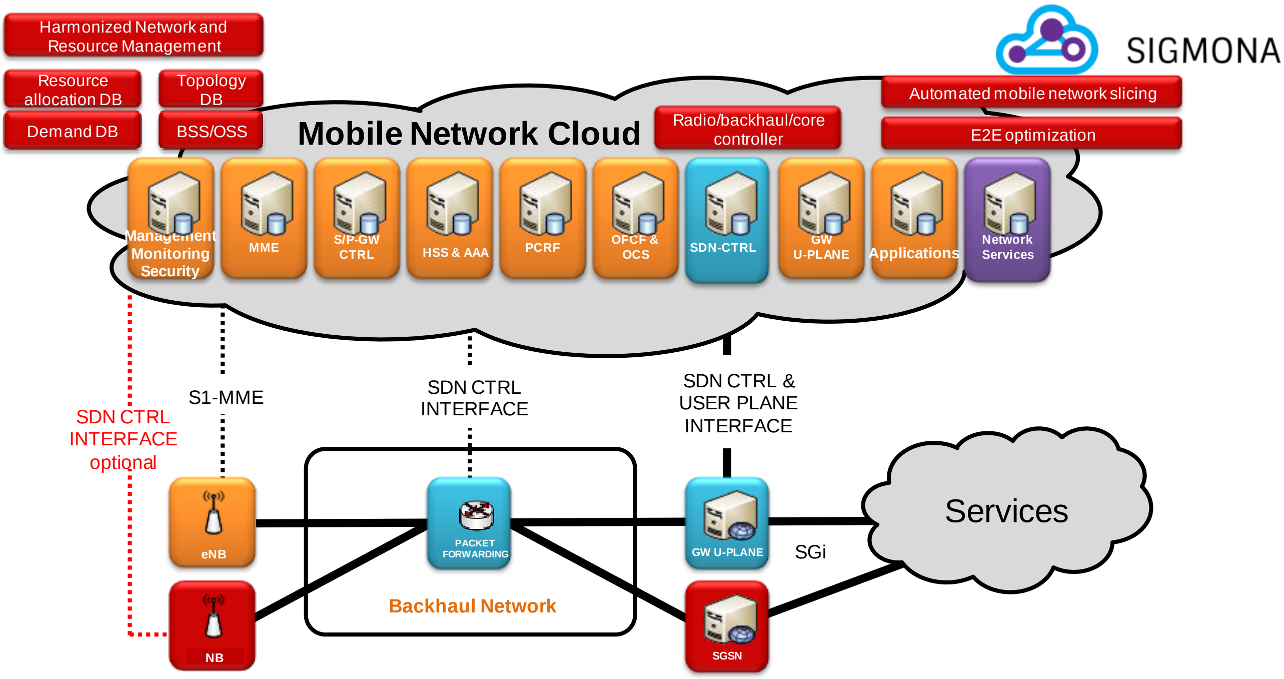


Figure 27: Mapping of the HMNR to the function based network model

4.3.3 Mapping to NFV model

4.3.3.1 Introduction of the NFV architecture model

Figure 28 shows the network architecture model defined by the NFV. Detailed description of the model can be found in [3].

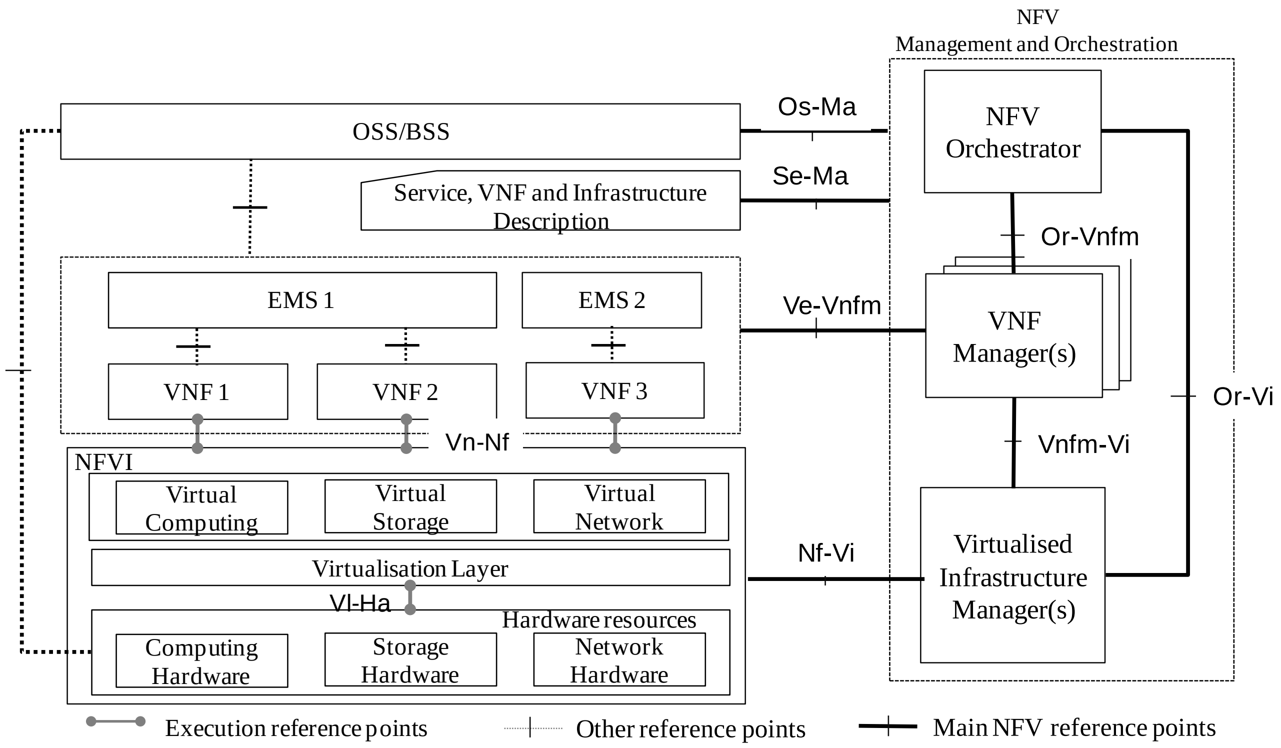


Figure 28: Network architecture model defined by NFV

4.3.3.2 Actual mapping to the model

Figure 29 shows the mapping of the HNRM concept to the NFV architecture.

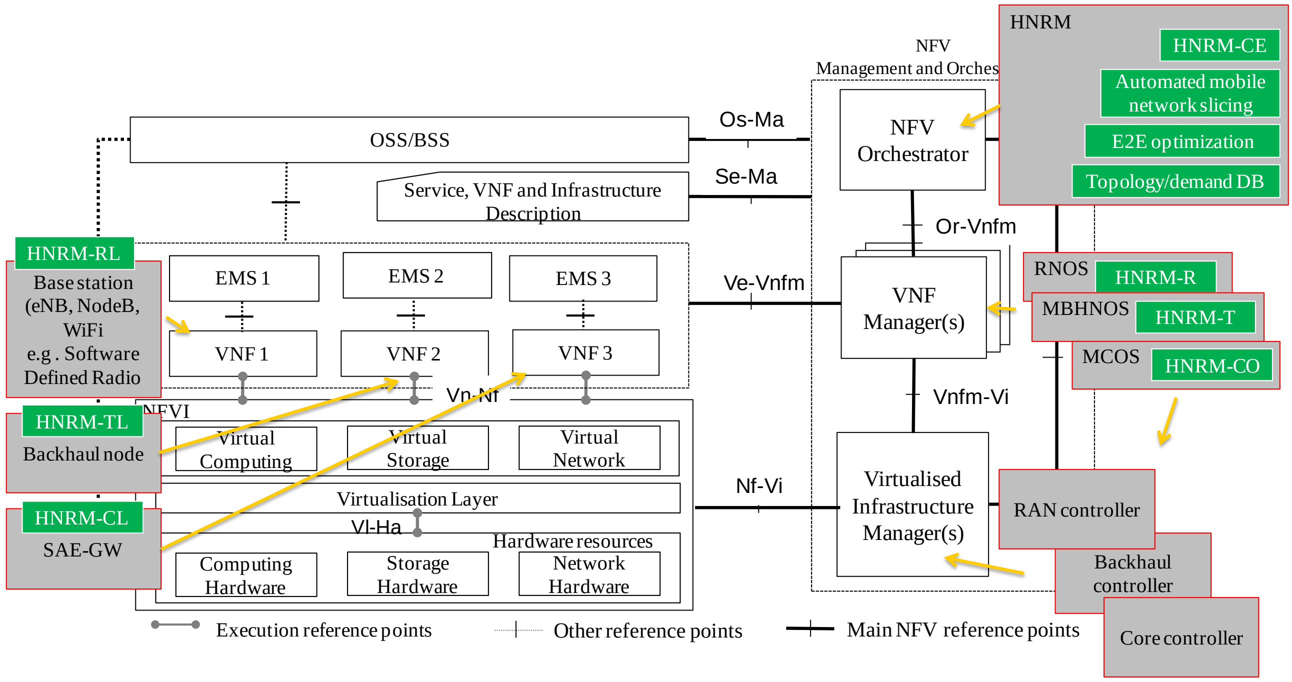


Figure 29: Mapping of HNRM to the NFV architecture

The HNRM-CE, Topology/demand DB can be part of the NFV Orchestrator, also the services provided by the HRNM can be logically part of the NFV Orchestrator. The mobile network domain specific network operating systems (RNOS, MHBNOS, MCOS) can be mapped to the VNF managers; in line with that also the HNRM-R, HNMR-T, HNRM-CO entities could be implemented in the VNF manager. The RAN, backhaul and core controller can be part of the Virtualized Infrastructure Manager(s). The local HNRM entities (HNRM-RL, HNRM-TL, HNRM-CL) can be mapped to the VNFs (Virtual Network Function) which are running the eNB, GW, router functionalities.

4.3.4 Summary of the HNRM mapping to different architectures model

The HNRM concept have been mapped to three different virtualized mobile network architectures. The mapping showed that the elements of the HNRM concept can be mapped to the different architecture models quite straightforward. This shows that currently, no obstacles are seen that could prevent the efficient implementation of the concept in virtualized mobile networks.

4.4 Automated network operation supported by HNRM

4.4.1 Automated, adaptive eNB-side TCP Proxy

The proper operation of eNB side TCP proxy requires that the data buffered by the proxy (and available only in the Proxy anymore) has to be forwarded to the target eNB during handover without any loss. [PSZ14] showed a solution which can grant that all the crucial TCP segments are forwarded by the source eNB to the target eNB during the handover. Unfortunately, there is no guarantee that all the forwarded segments will be received by the target eNB as on the transport connectivity packets can be dropped (e.g. due to RED operation or congestion) and there is no means in the LTE handover forwarding procedure to recover the lost packets.

NM can help in improving the situation by decreasing the probability of packet losses or can even grant 100% success rate depending on the solution inside NM. Figure 30 illustrates the eNB side TCP proxy issue and the NM entities involved in solving it. The solution requires involvement of the NM-CE, NM-R, NM-RL and NM-T entities. We identified two major operations modes depending on the mobility pattern of the users served by the eNB: reactive and proactive operation. In the former case the NM activates the eNB TCP Proxy related operation only when it is detected/predicted that a user with ongoing TCP Proxied connection will have handover; this suits well to cases where the majority of the users are nomadic, i.e. quasi-static users. The proactive operation suits well to eNBs serving high-mobility users, i.e. it is probable that a large number of users could be impacted by the handover problem.

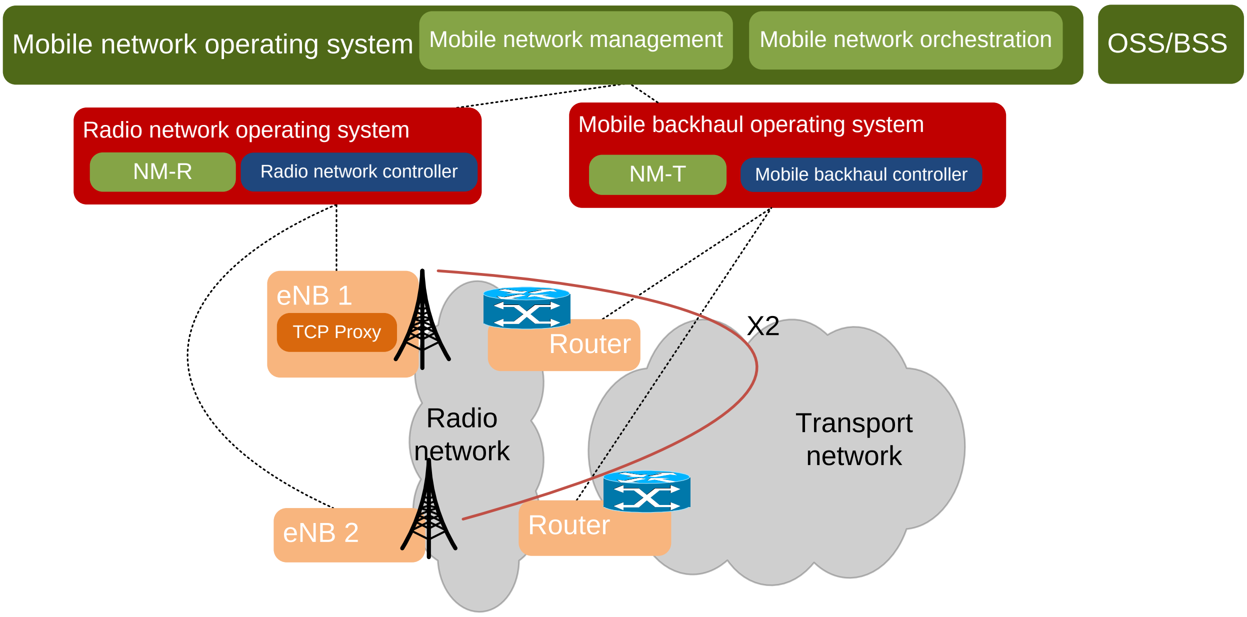

Mobile network operating system
Mobile network management
Mobile network orchestration
OSS/BSS
Radio network operating system
NM-R
Radio network controller
Mobile backhaul operating system
NM-T
Mobile backhaul controller
eNB 1
TCP Proxy
Router
X2
Radio network
Transport network
eNB 2
Router

Figure 30: NM handling the eNB TCP Proxy use-case

4.4.1.1 NM Reactive operation for the eNB TCP Proxy use-case

The flowchart of the proactive operation is shown in Figure 31. Whenever the LTE RRM detects/predicts that a user should perform handover the NM-RL entity is notified about it. The NM-RL checks if the user has an ongoing TCP proxied connection. In case it has, it notifies the NM-R about it; first it is checked from relevant sources (e.g. PCRF/BSS/OSS) if the user is eligible for special treatment (e.g. it is a high valued customer). In case it is not eligible then the procedure stops, otherwise the NM-R selects the method which should be used for eliminating the handover problem. After the method is selected the NM-R notifies the NM-RL at the source and if needed at the target eNB also about the selected method. Based on that the NM-RL entities carry out the necessary steps in the eNBs. In the reactive operation we consider only radio side method and skip involving NM-T and the MBHNOS as very prompt reaction is required. The radio side methods that the NM-RL can apply are introduced in the next section.

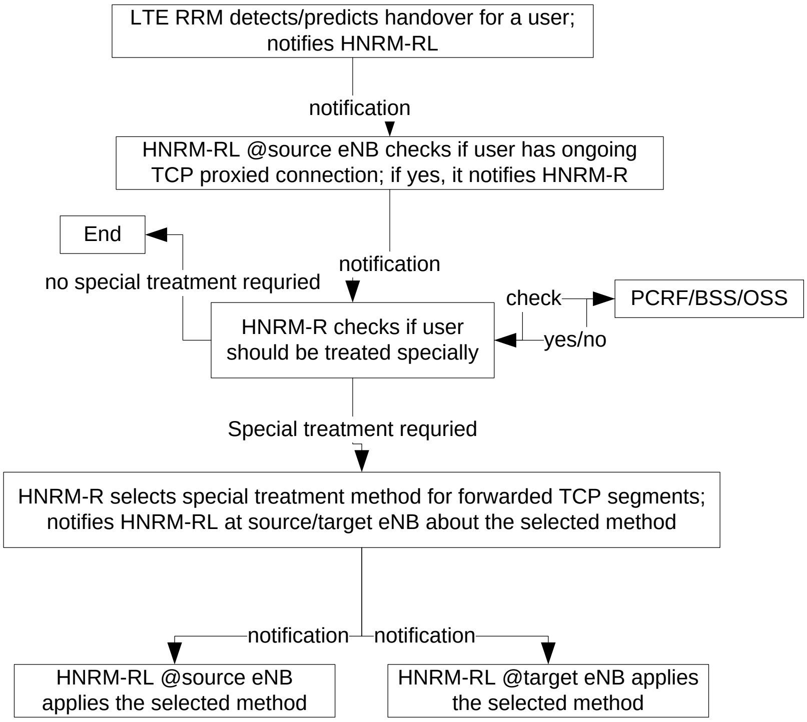

LTE RRM detects/predicts handover for a user; notifies HNRM-RL
notification
HNRM-RL @source eNB checks if user has ongoing TCP proxied connection; if yes, it notifies HNRM-R
End
notification
no special treatment requried
check
PCRF/BSS/OSS
HNRM-R checks if user should be treated specially
yes/no
Special treatment requried
HNRM-R selects special treatment method for forwarded TCP segments; notifies HNRM-RL at source/target eNB about the selected method
notification
notification
HNRM-RL @source eNB applies the selected method
HNRM-RL @target eNB applies the selected method

Figure 31: Reactive operation

4.4.1.2 Radio side methods for reliable data transfer through the X2

One option for reliable data transfer is the prioritization of the forwarded traffic on the X2 interface. This mechanism requires support only from the source eNB. In this case the NM-RL at the source eNB changes the X2 traffic settings so as the X2 traffic traverses the mobile backhaul prioritized.

The prioritization of the forwarded X2 traffic on the transport network is a capable mechanism to significantly reduce the risk of packet discards on the transport network that may occur due to high data bursts or buffer management mechanisms such as Random Early Detection (RED). As the volume and intensity of the X2 traffic is low, the forwarded packets are not likely to cause transport buffer overflow provided that they are handled separately from high volume bulk traffic. Prioritization requires that the transport network infrastructure implements QoS queuing in the transport schedulers so that the prioritization is effective in congestion situations. The prioritization may be performed by setting the DiffServ Code Point (DSCP) class in the outer IP address of the X2 GTP-U IP packets that carry the forwarded data from the source eNB to the target eNB. An Assured Forwarding (AF) class with sufficiently large relative weight configuration can be an effective setting. Simplicity and availability are also the advantage of the prioritization as this mechanism is implemented in virtually all transport devices.

In addition to the prioritization, alternative mechanisms may also be considered to provide extra or explicit guarantees for lossless X2 forwarding (or if prioritization is not available as alternatives the NM-R may decide to use these methods). One possible mechanism is to extend the forwarded segments with Forward Error Correction (FEC) codes in such a way that enables the receiver to reconstruct a certain amount of lost segments from the added redundancy in the FEC parts of the received packets. Another alternative is to simply duplicate each forwarded packet to provide 2N redundancy, i.e., receiving either of the two identical copies results in the successful transmission of the original segment. Similar techniques are already being considered for improving the TCP recovery [TF13]. Additionally, explicit ARQ retransmission mechanisms may also be implemented to guarantee completely error-free X2 transfer. The above extra functionalities require decoding, de-duplication or any inverse mechanism to recover or restore the received data stream to its original form. The previous solutions may be implemented as additional software blocks on the eNB; when the NM-RL receives the notification from the NM-R about the selected method, it can initiate and start the necessary software blocks. This can be achieved for example by starting a VM running the necessary functions.

4.4.1.3 NM proactive operation for the eNB TCP Proxy use-case

In case of the proactive operation the NM grants the transport resources required for proper TCP Proxy operation for handover cases before starting the TCP operation, Figure 32 shows exact operation.

When TCP Proxy activation is requested for an eNB, the NM-R is notified about it. It checks from eNB related mobility information what the usual handover pattern for the eNB is. If it is used mainly by nomadic users, i.e. the number of expected handovers is low, then NM-R concludes that it is more beneficial to use reactive operation and the TCP Proxy operation can be started in the eNB. If the eNB has frequent handovers then the number of users possible impacted by the handover issue of the TCP proxy is high, thus the NM-R concludes that proactive operation is required. In the next step the NM-R checks towards which eNBs does the eNB have significant number of handovers; based on this it requests X2 connectivity between the eNB where the TCP proxy should run and identified relevant neighbour eNBs from the NM-CE. In the connectivity request the NM-R also provides information about the required X2 bandwidth; this is a higher value compared to eNBs without TCP proxy and in this case the amount of to be forwarded traffic is higher as the buffer of the TCP proxy has to be transferred also. For the calculation of the X2 bandwidth the NM-R can utilize the mobility information; based on the predicted handover frequency it can estimate the required bandwidth.

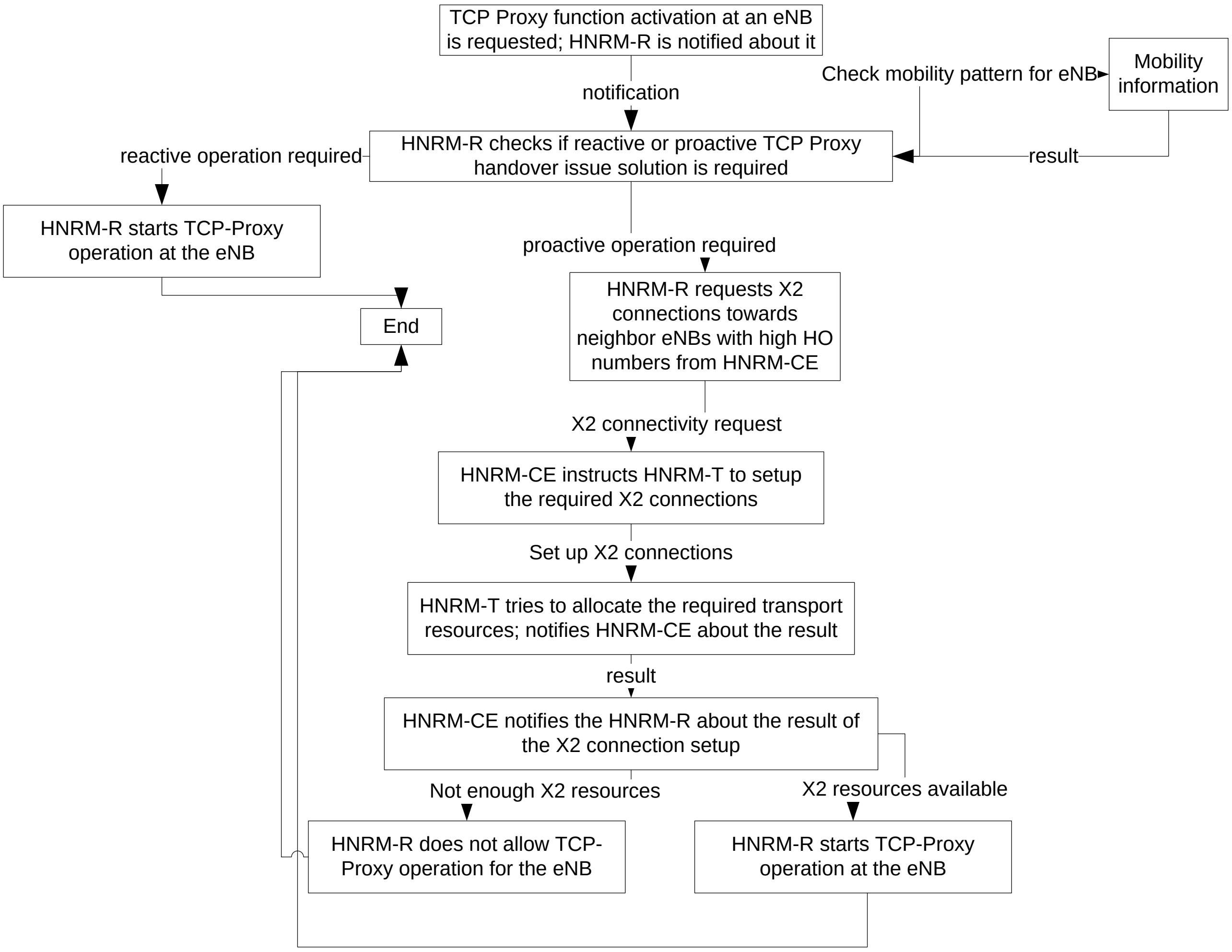

TCP Proxy function activation at an eNB is requested; HNRM-R is notified about it
notification
Check mobility pattern for eNB
Mobility information
result
HNRM-R checks if reactive or proactive TCP Proxy handover issue solution is required
reactive operation required
HNRM-R starts TCP-Proxy operation at the eNB
End
proactive operation required
HNRM-R requests X2 connections towards neighbor eNBs with high HO numbers from HNRM-CE
X2 connectivity request
HNRM-CE instructs HNRM-T to setup the required X2 connections
Set up X2 connections
HNRM-T tries to allocate the required transport resources; notifies HNRM-CE about the result
result
HNRM-CE notifies the HNRM-R about the result of the X2 connection setup
Not enough X2 resources
X2 resources available
HNRM-R does not allow TCP-Proxy operation for the eNB
HNRM-R starts TCP-Proxy operation at the eNB

Figure 32: Proactive operation

When the NM-CE receives the information about the requested X2 connections, it first checks if the X2 connection is already setup for a specific eNB pair and if the requested bandwidth is already reserved; if yes, then there is nothing to do about the request. If the X2 connection is not set up at all or the reserved capacity is not matching the requested one, then the NM-CE requests the setup of the corresponding X2 connection from the NM-T. It tries to allocate the requested transport resources and notifies the NM-CE about the result of the process. NM-CE notifies the NM-R about the X2 status; the NM-R can decide if not all the requested X2 resources could be granted if still wants to allow TCP proxy operation in the eNB. For example, if X2 resources are missing only towards eNB where the number of handover is that high, it may allow the operation; if X2 resources are missing only towards one eNB with frequent handovers then it might consider preventing TCP proxy operation due to the expected high negative user experience. These decisions are to be controlled bases on the operator’s policy and approach. If all the X2 resources are in place then the TCP proxy operation will be started by the NM-R in the selected eNB.

Even though, the proactive operation grants the X2 resources required for proper TCP Proxy operation in case of handovers, due to the dynamicity of the mobile networks situations can occur where the originally provisioned X2 resources are not enough for proper operation anymore. To detect and correct such situation, the QoE of the users shall be evaluated in order to detect TCP proxy related handover problems and the necessary steps have to be taken to mitigate the problems. Figure 33 shows how this can be implemented in the NM.

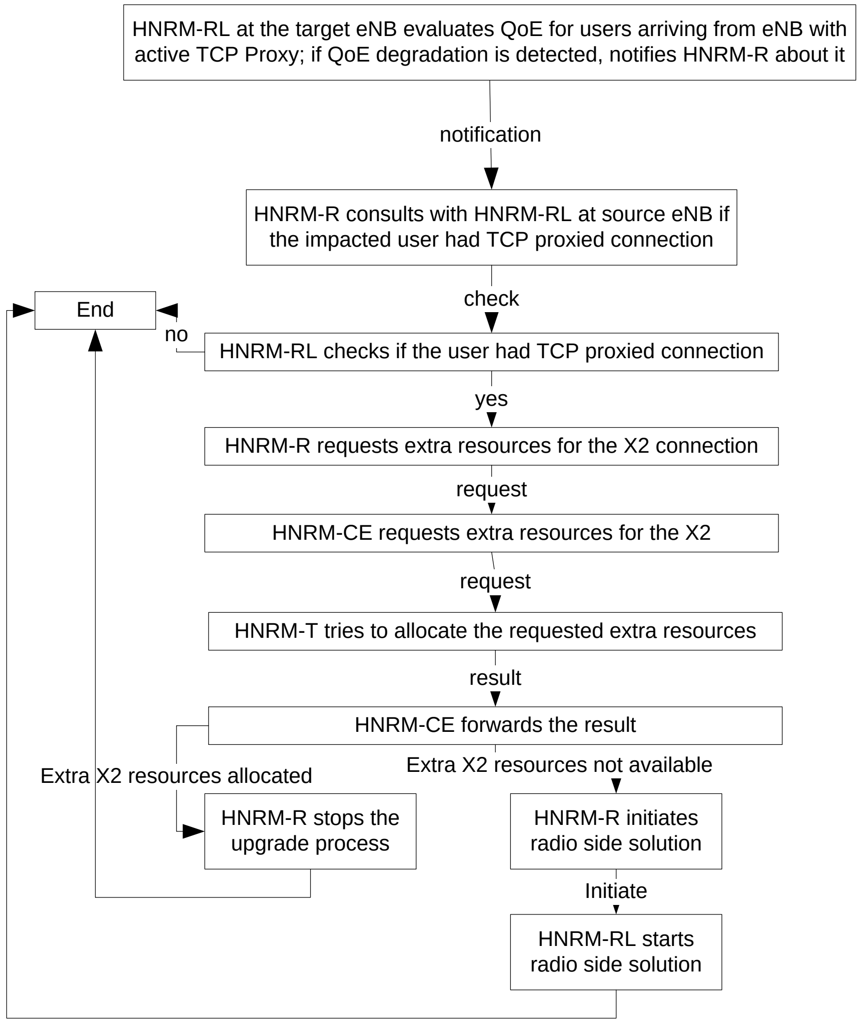

HNRM-RL at the target eNB evaluates QoE for users arriving from eNB with active TCP Proxy; if QoE degradation is detected, notifies HNRM-R about it
notification
HNRM-R consults with HNRM-RL at source eNB if the impacted user had TCP proxied connection
check
End
no
HNRM-RL checks if the user had TCP proxied connection
yes
HNRM-R requests extra resources for the X2 connection
request
HNRM-CE requests extra resources for the X2
request
HNRM-T tries to allocate the requested extra resources
result
HNRM-CE forwards the result
Extra X2 resources allocated
Extra X2 resources not available
HNRM-R stops the upgrade process
HNRM-R initiates radio side solution
Initiate
HNRM-RL starts radio side solution

Figure 33: X2 upgrade in case of QoE degradation

The NM-RL at the target eNB of the handover checks the QoE of the user right after the handover; this can be done either for all handovers where the TCP proxy is enabled at the source eNB (this is information is provided by the NM-R to all the NM-RLs running on the neighbour eNBs of the eNB where the TCP Proxy is running) or it can be done periodically selecting a few handovers for QoE evaluation. Based on the QoE evaluation the NM-RL can detect QoE degradations after handover; when such situation is detected the NM-RL notifies the NM-R about the QoE degradation and also provides information about the impacted users and the source eNB of the handover. Based on that the NM-R consults with the NM-RL at the source eNB to find out if the impacted users had active TCP Proxied connections during the handover. If not, it is concluded that the QoE degradation is not due to the TCP Proxy operation and the process stops. Otherwise it can be assumed that the QoE degradation is due to the TCP Proxy operation; this can be caused by change in the mobility pattern over time (this can follow trends on different time scales, e.g. daily trend). Increased handover frequency between the eNB pair increases the X2 bandwidth required for lossless X2 forwarding. Thus, the NM-R requests extra X2 resources from the NM-CE for the respective X2 connections in order to mitigate packet losses during X2 forwarding. The NM-CE requests the X2 resources from the NM-T; it will allocate the required resources if possible. The result of the X2 bandwidth extension process will be proxied by the NM-CE to the NM-R. If the required transport resources were allocated then the upgrade process stops. If they could not be granted due to missing transport capacity on the X2 path, the NM-R can initiate the radio side solutions described in section 4.4.1.2. The selected radio side solution(s) is started by the NM-RL. In case of frequent handovers, the forwarding of the TCP Proxy and applying the radio side solutions might introduce considerable processing load on the eNB, that is why it is preferred to solve the issue with X2 capacity extension in the first place. However, if capacity extension is not possible, then radio side solutions should be applied, but to avoid raising too much processing load, it can be used in a focused way, i.e. applying only for priority users, towards selected eNBs, etc.

4.4.2 Self-configuring, self-optimizing mobile backhaul with HNRM

4.4.2.1 The need for SON in Mobile Backhaul

Typically, MBHs are complex and heterogeneous systems (multiple technology layers from multiple vendors) that are spanning over large areas (whole countries), and that are collecting and aggregating the traffic of a whole mobile network. Accordingly, their CAPEX and OPEX have significant share in the total cost of ownership. Keeping these costs low mandates careful planning and reduced operation costs through automation. Nevertheless, the planning is typically a process with low accuracy as the resource need is calculated based on traffic forecasts (by nature is inaccurate) that at best are based on historical measurements and simplified network models (simplifications are needed to keep the numerical complexity and processing requirements at reasonably low level). During the planning, the parameters and configuration of each network element are calculated and these values are eventually downloaded to the network elements. Due to the inaccurate inputs and rough models these parameters will most likely not provide optimal system operation under traffic load. The common characteristic of all the alternative/complementary transport solutions is that in each case long-lasting transport tunnels are configured between the radio access nodes and that QoS schemes are applied at the packet level. These transport tunnels are configured either manually or via provisioning tools (that provide some level of automation) when the network is extended with new eNBs. During the provisioning process, the transport tunnels are configured on top of the existing ones one by one (as the network is extended) having their path calculated in a locally optimal way. The result is suboptimal on the system level and inefficient compared to the case when all the existing and new transport tunnels over a system are known in advance and the configuration is calculated by considering the whole network, the total demand, the granularity of the allocations vs. the granularity of the resources and applying MLO techniques. The gap is increasing with the size of the transport network and the amount of configured connections, making room for significant CAPEX savings. As discussed before, resource allocations and transport QoS parameters are statically configured using recommended values or in the best case based on the forecasted traffic. None of these are optimal as there is no globally suitable configuration whereas the traffic demand is changing dynamically. Due to the complexity of the system and the high number of parameters that must be configured in each case, reconfigurations and re-parameterizations are executed rarely. Efficient operation would require the self-adaptation of the transport parameters to the continuously changing traffic. When the transport connectivity is provided by third parties via transport services accessed through UNIs, according to the pre-agreed SLAs, the cost and the resource efficiency is a significant aspect upon the definition/calculation of the required transport services.

As the MBH has significant role in end-to-end performance its inefficient operation has significant negative impact on user experience that impacts the revenues the operators can realize (e.g., poor service leads to high churn). The Self Organizing Network (SON) concept was successfully deployed in LTE. SON reduces the risks of manual errors, enables optimal resource utilisation, lowers the energy usage and allows performance thus lowers OPEX. The standardised SON functions are addressing self-healing, self-configuration and self-optimisation of LTE's Radio Network Layer. Introducing SON to MBH is important to reduce OPEX and to increase the resource usage efficiency that reduces the CAPEX.

4.4.2.2 Issues addressed by the autonomous operation

In addition to the aspects described in the above section, the following pain points are addressed by the solution to be described in the next sections:

- Increased system complexity: large amount of network elements, complex topologies, multi-technology environment in the MBH that means that the system is difficult to dimension, commission and manage;
- Complex and dynamic traffic mix that can't be handled efficiently with static parameter sets, that is, there is no globally valid configuration that could be downloaded to each network element; adaptive and context specific parameterisation is needed instead;
- Inaccurate planning that is based on traffic forecast and simplified network models; even the most accurate traffic forecast becomes obsolete rather soon. In order to prevent that the planning (not only the configurations but the physical capacity and the resource allocation) becomes obsolete, over-dimensioning is needed;
- Not aligned transport and radio QoS. The former is aggregate centric whereas the latter is bearer centric. This causes that upon transport congestion, the end-to-end QoS targets are not met;
- Time consuming, failure prone node, network commissioning/operation/management;
- The need for rapid delivery of mobile services (coverage, connectivity, capacity, access to the applications).

These pain points are leading to high TCO, poor QoS/QoE, reduced system efficiency, waste of resources and energy, frequent degradations and difficult troubleshooting. As a potential improvement, a new solution is proposed that extends the SON concept (self-healing, self-configuration, self-optimisation) to the MBH. The solution increases automation, simplifies the planning and commissioning process, harmonizes transport and radio QoS, increases system efficiency, reduces TCO, provides valuable insight during trouble shooting, reduces energy consumption and keeps the resource allocations at a system level optimum.

4.4.2.3 SON for MBH

The scope of the SON for MBH framework is to increase the level of automation within the Mobile Backhaul (MBH) by applying true plug-and-play, self-configuration and self-optimization mechanisms. The operation of the framework is based on advanced KPI collection, anomaly and degradation detection mechanism and state of the art network management solutions such as Software Defined Networking (SDN). SON for MBH enables simplified planning and minimum touch commissioning and automated and adaptive network operation. The latter guarantees efficient network resource utilization, harmonization of the radio and transport QoS architectures and QoE driven operation and thus provides significantly reduced CAPEX and OPEX. The solution is versatile as it was designed to work in multivendor Radio Access (RA) and MBH environments.

The solution consists of SON for MBH Agents deployed at the eNB, S-GW/SAE-GW and a SON for MBH Manager located in the Core Network (Figure 34).

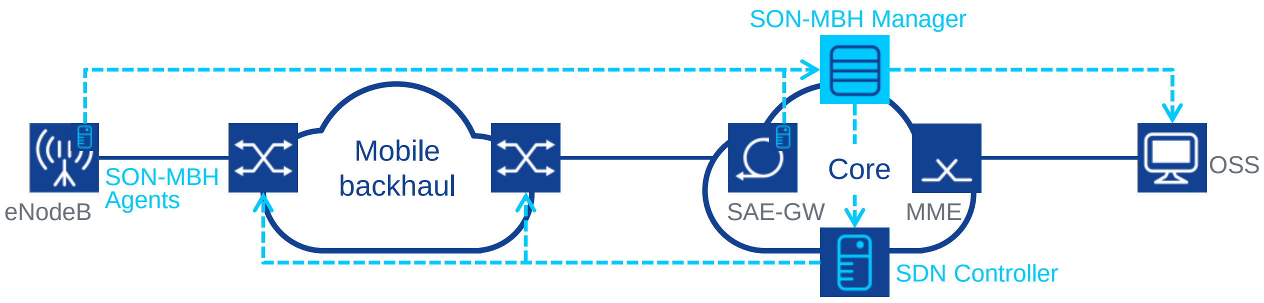


Figure 34: Building blocks of the SON for MBH

4.4.2.3.1 The role and functionality of the SON for MBH Agent

The eNB side SON for MBH Agent (SMA) is a software entity running on or attached to an eNB as a site device. Its role is to maintain a harmonized QoS over the RA system and a consistent end-to-end transport configuration. Additionally, the eNB side MSA is responsible to complete the commissioning of the eNB by creating and configuring the transport connectivity of the eNB. In similar manner, the MSA detects the ANR originated activation/configuration of an X2 interface and automatically establishes the corresponding transport connectivity.

In order to detect any anomaly and degradation on the transport layer, it monitors and profiles the whole traffic of the eNB. When an anomaly/degradation is detected, it first identifies the reason of the degradations, initially separating radio and transport side problems. In case the problems are rooted in the transport network, the MSA further localizes the problematic transport segment. Finally, if needed (e.g., in order to resolve the degradation or to maintain the QoS) it performs/triggers reconfigurations/optimizations. The reconfigurations can target the transport interface, the site router of the eNB (if such device exists) or the end-to-end transport service. MSA is a measurement point that extracts real time QoS and QoE KPIs and intercepts/detects relevant events from/within the user, control and management plane traffic of the eNB. Moreover, the MSA monitors the control plane traffic/messages of the transport network in order to detect any relevant change in the status of the network/link/path, etc. The MSAs communicate with each other and the SON for MBH Manager through in-band (header enrichment) or out of band (e.g., JSON/IPFIX) interfaces. Depending on the implementation (residing agent in a network element, standalone box, etc.) the MSA uses internal interfaces for eNB side transport configuration or legacy management interfaces such as SNMP, CLI, etc.

The S-GW/SAE-GW side MSA is a software entity running on or attached to an S-GW/SAE-GW, that acts as a counterpart of the eNB side MSAs, with similar role with these. Through cooperative measurements with the eNB side MSAs, it provides an efficient framework for anomaly/degradation detection and localisation.

4.4.2.3.2 The role and functionality of the SON for MBH Manager

The SON for MBH Manager (referred to as Manager for short) is a software entity that is either attached to an existing OSS tool such as NetAct in form of a content pack or is running on a standalone node. The Manager acts as a system level optimization and configuration entity that is responsible of resolving the degradations that are related to the end-to-end transport service or are affecting multiple network elements, e.g., congestion on a transport link shared by the traffic of multiple eNBs. Additionally, the Manager is responsible of trend analysis and prediction based on the relevant KPIs collected by the MSAs. This allows preventive operation, e.g., to trigger reconfiguration before the negative tendencies would cause hard failures. The KPIs collected by the MSAs are upstreamed to the Manager through out of band interfaces such as JSON or IPFIX depending on the use case in a raw or processed/aggregated format. Additionally, the MSAs trigger the Manager whenever a detected anomaly can’t be resolved through local configuration or it is due to a problem with the transport service (e.g., not proper resource allocation) or it is clearly within the MBH, i.e., it might affect other eNBs as well. The Manager collects these triggers, consolidates them to filter out false positives and finally correlates them to identify incidents reported by distinct MSAs that are caused by the same failure/problem. The Manager can operate in real time, implementing a closed control loop, e.g., as an entity that resolves single or transient incidents on the MBH and/or through a long control loop, e.g., as an entity that resolves persistent degradations or prevents failures caused by negative trends. Additionally, it is able to proactively configure the system to follow the typical traffic profile of the eNBs and the shifting load between network domains (caused by the mobility routing of the users). In addition of resolving transient or pathological failures the Manager is responsible of maintaining the system efficiency and keeping the system operation and resource usage/allocations at an optimal working point. Accordingly, in addition of being triggered by the MSAs it can trigger actions by itself whenever it detects not efficient resource usage/allocation or system operation by analyzing the collected KPIs. This mandates that the Manager is integrated with the existing OSS and CEM tools and has access to their KPI database. Depending on the environment (the existence of SDN controller, PCE or other transport provisioning tool) the Manager either connects directly to some or all the transport nodes through the existing/standard management and configuration interfaces or is acting as a user of the SDN controller, PCE or other transport provisioning tool. In the rest of this IPR, the operation of the Manager is explained as an application running as a client of an

SDN controller, however similar operation is possible in case of a PCE or any other provisioning/transport management tool. Moreover, if the reach of an SDN controller is limited to a certain domain, or there are several SDN controllers managing distinct network domains, the Manager acts as integrator, that is, triggers reconfiguration or optimization by separately addressing the individual SDN controllers or even directly configuring the network elements out of the reach of the SDN controller(s) in order to achieve a coherent end-to-end system configuration/status. The integration of the Manager to the OSS ecosystem enables the reporting of the actual network status, distribution of the special KPIs collected by the SON for MBH infrastructure or issuing alarms in case a problems/failures that can't be solved by the Manager (e.g., that require capacity extension).

4.4.2.4 The most relevant use cases/novel features of the SON for MBH framework

4.4.2.4.1 Plug and play eNB commissioning

This use case simplifies the transport planning to a great extent and increases the level of automation of the eNB commissioning process. The current eNB commissioning requires that transport services are pre-planned and pre-configured before the provisioning process. In contrast, the SON for MBH framework eliminates the need of pre-planning and pre-configuring the transport connectivity of each and every eNB. The eNB side SMA detects the start of the commissioning process and automatically triggers the transport configuration through the Manager. By the time the first user plane connection is established the transport connectivity is available. As a second step, the SON for MBH framework optimizes the resource allocation of the eNB to the traffic demand its serves.

4.4.2.4.2 Automated transport configuration for the X2 interface as complementary mechanism to the ANR

In a similar manner to the use case described above, the SON for MBH framework is able to detect the newly activated/established X2 interfaces and configure the transport connectivity in an optimal manner (reduced number of hops, efficient resource allocation).

4.4.2.4.3 eNB, SAE-GW transport interface and transport router self-configuration and parameter optimization

This is a use case that not only simplifies the planning and configuration process but also enables harmonized radio and transport QoS and guarantees coherent end-to-end transport configuration. Through the SMAs, the SON for MBH enables the dynamic and adaptive configuration of the transport parameters that serve best a given actual traffic mix. Accordingly, the SMAs are monitoring the traffic in order to detect if the target QoS is not met (for example due to the not harmonized radio and transport configuration) or if the configured resources are not sufficient for proper QoS. Whenever such cases are detected they are automatically reconfiguring the relevant transport parameters as a self-configuring process or are triggering the Manager for action (in case the problem can't be solved through local reconfiguration). This allows the deployment of network elements with default configuration as the solution itself takes care that the best parameters for a given eNB are found.

4.4.2.4.4 Transport service self-optimisation

The SON for MBH is capable of detecting if a transport service is not configured with the optimal resource allocation and can trigger a reconfiguration. Accordingly, the SMAs trigger the Manager in case the detected degradation is due to poor transport service configuration. As a result, the Manager selects one of the following actions: (1) increase of the bandwidth allocation in case of enough free resources on the existing path; (2) in case of not enough transport resources it identifies the underutilized transport services that are sharing the same resources with the one having not enough bandwidth allocation, and downsizing them to make room of extending the allocation of the problematic transport service; (3) rerouting of transport services (and at the same time maintaining even load and optimal system resource utilisation) in case of not enough resources on the path of the congested transport service and no possibility to free up enough bandwidth by downsizing other services; (4) triggering the operator with an alarm message coupled with recommended configuration in case none of the above actions are possible.

4.4.2.4.5 Traffic trend analysis and preventive optimisation

The Manager continuously monitors the traffic in order to detect traffic trends and to identify the potential future resource limitations/narrow points within the system. As a result it updates the system configuration to prevent these incidents. In order to leverage the gain of the geographical diversity and the daily commuting routine of the users, whenever the topology allows it, the Manager can free up resources allocated to serve the suburban cells and increase the resource allocation of the downtown areas, etc. during the business hours and revert the configuration after the business hours are over. Moreover, the Manager is capable of delaying the reconfiguration in order to not prevent planned network extensions or to operate based on allocation and retention priorities, cost functions, etc.

4.4.2.4.6 Maintaining optimal system level configuration and resource allocation

At network deployment when the system is extended upon the commissioning of eNBs, the system state can drift from the optimal status as transport resources are provisioned one-by-one to the new eNBs. This is because the transport services of a newly provisioned eNB are created by considering the actual status, i.e., the already configured services and the resource need of the new transport services being established. This approach provides a local optimum but as the number of newly configured eNBs increases leads to suboptimal configuration at system level. The Manager continuously monitors the existing allocations (resource allocations vs. available capacity, the paths of the transport tunnels vs. the topology, etc.) and in case enough gain can be achieved by a system level optimisation it calculates the optimal path for each service (by considering the system level optimum), creates a step-wise plan for reconfiguration and finally triggers the reconfiguration(s) according to this plan.

5. Conclusions

Network virtualization and cloud computing are two technologies that are available since longer time and have already gained their momentum in the IT area; in parallel, SDN is also gaining its momentum in computer networking. As these technologies are proven to be feasible to be implemented and to provide performance and cost gains, the mobile network operators and vendors also recognize their possible benefits in mobile networks. The trend of virtualization, cloudification of mobile networks and the introduction of SDN principles already started; virtualization is planned to be used in all areas of the mobile network, i.e. in radio, backhaul and core networks. Similarly, SDN like solutions are investigated and proposed for all of those network areas.

This document investigated how SDN is impacting mobile backhaul solutions. Section 2 provided first an overview of the architecture and building blocks of SDN MBH networks, then looked at the MBH network elements. It also introduced the main functions of SDN controllers and the usual measurement, data analytics and orchestration functions to be deployed in SDN MBH.

After the overview of SDN MBH, three concepts were introduced that aim at utilizing the SDN control for improved network operation and solving the issues related to integrating legacy networks under SDN control. The concept for integrating wireless mesh networks (WMNs) into SDN control is aiming at the previously mentioned goal. WMNs play an important role in mobile networks as they often provide the last mile access network, however integrating them under SDN control is not a straightforward task. The reason for that is that the legacy WMN management functions (like load balancing) requires very low latency control loop, whereas delegating those functions to a central SDN controller would result in too high latency possibly ruining the operation of those functions. The main idea of the concept is to hide the WMN internal structure from the SDN controller by showing it as a single switch towards the controller.

Section 3 introduced the concept of integrated RAN and mobile backhaul controller. Here, the goal is to develop an advanced RAN controller (RANC) architecture which can monitor and act according to various system state information such as resource requirements, mobility and QoS. Moreover, the concept also proposed to integrate that operation with the mobile backhaul controller to meet end-to-end performance requirements in a hierarchical architecture.

The concept of harmonized network and resource management (HNRM) was introduced in Section 4.2. This concept looked at cross domain and cross layer optimization possibilities of mobile networks: it proposed to harmonize the operation of the resource, mobility and network management tasks in mobile networks. The concept operates on three levels: mobile network, mobile network domain and network device level; on each level there is an HNRM entity which is carrying out the harmonization/optimization tasks.

6. References

[BFD] Bidirectional Forwarding Detection (BFD). http://tools.ietf.org/html/rfc5880

[BSCW+13] D. Bojic, E. Sasaki N. Cvijetic, T. Wang, J. Kuno, J. Lessmann, S. Schmid, H. Ishii, and S. Nakamura, Advanced wireless and optical technologies for small-cell mobile backhaul with dynamic software-defined management IEEE Communications Magazine, September 2013.

[DPDK] DPDK: Data Plane Development Kit. http://dpdk.org/

[ETH OAM] 802.1ag. IEEE Standard for Local and Metropolitan Area Networks Virtual Bridged Local Area Networks Amendment 5: Connectivity Fault Management

[FC13] F. Capozzi, G. Piro, L. Grieco, G. Boggia, and P. Camarda, "Downlink packet scheduling in lte cellular networks: Key design issues and a survey," Communications Surveys Tutorials, IEEE, vol. 15, pp. 678–700, Second 2013.

[FK97] F. Kelly, "Charging and rate control for elastic traffic," European Transactions on Telecommunications, vol. 8, pp. 33– 37, 1997.

[IVS] Project Floodlight, Indigo. http://www.projectfloodlight.org/indigo/

[KS07] K. D. Singh and D. Ros, "Normalized rate guarantee scheduler for high speed downlink packet access," in Global Telecommunications Conference, 2007. GLOBECOM'07. IEEE, pp. 576–580, IEEE, 2007.

[MEVICOD32] P. Wainio and T. Taipale, Wireless Mesh access backhaul for small cell base stations, in Jose Costa-Requena, editor, Innovative Solutions for Mobile Backhaul, CELTIC / CP7-011 MEVICO, 2012. http://www.mevico.org/D32.pdf.

[MRFRW13] C. Monsanto, J. Reich, N. Foster, J. Rexford, and D. Walker, Composing Software-Defined Networks, in Proceedings of NSDI '13: 10th USENIX Symposium on Networked Systems Design and Implementation.

[NFV WP] Network Functions Virtualisation. https://portal.etsi.org/nfv/nfv_white_paper.pdf

[NETCONF] NETCONF Configuration Protocol. https://tools.ietf.org/html/rfc4741

[NGMN-BHR] Small Cell Backhaul requirements, NGMN Alliance, 2013.

[NW DEBUG] Where is the Debugger for my Software-Defined Network? http://www.scs.stanford.edu/~dm/home/papers/handigol:ndb-hotsdn.pdf

[ODP] OpenDataPlane project. http://www.opendataplane.org/

[OF] OpenFlow Switch Specification. https://www.opennetworking.org/images/stories/downloads/sdn-resources/onf-specifications/openflow/openflow-spec-v1.4.0.pdf

[ON16] O. Narmanlıoglu, E. Zeydan, "Network Virtualization for Mobile Operators Using QoS-Aware Schedulers in Software-Defined Based LTE Networks", EUCNC 2016, Athens, Greece

[OVS] Open vSwitch: An open virtual switch. http://openvswitch.org/

[PH02] P. Hosein, "Qos control for WCDMA high speed packet data," Mobile and Wireless Communications Network, 2002. 4th International Workshop on, pp. 169–173, 2002.

[PSZ14] P. Szilágyi, Z. Vincze and V. Csaba, "Handover Friendly TCP Proxy Integrated in the LTE eNodeB", 2014 IEEE 25th International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC'14), Sep 2-5, 2014, Washington DC, USA

[REST] Representational state transfer. http://en.wikipedia.org/wiki/Representational_state_transfer

[RESTCONF] RESTCONF Protocol. draft-ietf-netconf-restconf-03. https://tools.ietf.org/html/draft-ietf-netconf-restconf-03

[RJ13] R. Jain and S. Paul, "Network virtualization and software defined networking for cloud computing: a survey," Communications Magazine, IEEE, vol. 51, pp. 24–31, November 2013.

[SGYA+09] R. Sherwood, G. Gibb, K. Yap, G. Appenzeller, M. Casado, N. McKe-own, and G. Parulkar, Flowvisor: A network virtualization layer, OpenFlow Switch Consortium, Tech. Rep, 2009.

[Taipale12] T. Taipale, Feasibility of wireless mesh for LTE-Advanced small cell access backhaul, master's thesis, Aalto University School of Electrical Engineering, 2012. http://lib.tkk.fi/Dipl/2012/urn100686.pdf.

[TF13] T. Flach, N. Dukkipati, A. Terzis, B. Raghavan, N. Cardwell, Y. Cheng, A. Jain, S. Hao, E. Katz-Bassett and R. Govindan, "Reducing Web Latency: The Virtue of Gentle Aggression," SIGCOMM Comput. Commun. Rev., vol. 43, no. 4, pp. 159-170, 2013.

[Y3011] Recommendation ITU-T Y.3011 Framework of network virtualization for future networks, 2012.

[YANG] YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF). https://tools.ietf.org/html/rfc6020