Objective FP7-ICT-2009-5-257448/D-4.1

Future Networks

Project 257448

“SAIL – Scalable and Adaptable Internet Solutions”

D-4.1:
(D-C.1) Architectural Concepts of Connectivity Services

Date of preparation: **11-07-31**
Start date of Project: **10-08-01**
Project Coordinator: **Thomas Edwall**
Ericsson AB

Revision: **1.0**
Duration: **13-01-31**

Document Properties:

Document Number:	FP7-ICT-2009-5-257448-SAIL/D-4.1(D-C.1)	
Document Title:	**D-C.1 Architectural Concepts of Connectivity Services**	
Document responsible:	Lucian Suciu (FT)	
Editor(s):	N.A.	
Author(s):	Ramón Agüero (UC), Pedro A. Aranda (TID), Philippe Bertin (FT), Roksana Boreli (NICTA), Luisa Caeiro (IST-TUL), Reuven Cohen (Technion), Rami Cohen (Technion), Luis M. Correia (IST-TUL), Fariborz Derakhshan (ALUD), Lúcio Studer Ferreira (IST-TUL), Marta Garcia (UC), Guy Grebla (Technion), Heidrun Grob-Lipski (ALUD), Sofiane Hassayoun (IT), Marco Marchisio (TI), Ronald Marx (FHG), Olivier Mehani (NICTA), Ibrahim Menem (TID),	Avi Miron (Technion), Susana Perez (Tecnalia), Danny Raz (Technion), Horst Rößler (ALUD), David Ros (IT), Simone Ruffino (TI), Peter Schefczik (ALUD), Peter Schoo (FHG), Michael Soellner (ALUD), Golam Sarwar (NICTA), Lucian Suciu (FT), Andreas Timm-Giel (UHB), Sascha Todt (FHG), Asanga Udugama (UHB), Iñigo Urteaga (Tecnalia), Yasir Zaki (UHB), Liang Zhao (UHB)
Target Dissemination Level:	PU	
Status of the Document:	Final	
Version	1.0	

Production Properties:

Reviewed by:	Marcus Brunner (NEC), Luis M. Correia (IST-TUL), Stephen Farrell (TCD), Benoit Tremblay (EAB)

Disclaimer:

This document has been produced in the context of the SAIL Project. The research leading to these results has received funding from the European Community's Seventh Framework Programme ([FP7/2010-2013] [FP7/2010-2013]) under grant agreement n°257448

All information in this document is provided "as is" and no guarantee or warranty is given that the information is fit for any particular purpose. The user thereof uses the information at its sole risk and liability.

For the avoidance of all doubts, the European Commission has no liability in respect of this document, which is merely representing the authors view.

Abstract:

The deliverable D.C.1 presents the first year results from the research work carried-out on the Open Connectivity Services (OConS) in the SAIL project, thus tackling aspects such as: i) the initial architectural framework with its functional entities and interfaces, ii) the proposed connectivity services and their management mechanisms, and iii) the mapping of these services and mechanisms on the architectural framework according to several use-cases.

Keywords:

Future Internet, SAIL, Openness, Advanced Connectivity Services, Decision Making, Component-based Architecture, Multi-Path/Multi-Point/Multi-Protocol, Dynamic Distributed Mobility, Data-Centre Interconnect, Resource Management, Wireless Challenged Networks.

Executive Summary

This document is a public deliverable of the Scalable and Adaptive Internet Solutions (SAIL) EU-FP7 project and it presents the Open Connectivity Services (OConS) architectural framework, the proposed connectivity services and their management mechanisms, as well as the mapping of the components on the proposed architecture according to several use-cases.

In our research work we were mainly motivated by the fact that traditional approaches to the networking are starting to show their limits when it comes to the requirements imposed by new and upcoming applications and services. However, it is widely recommended not to dismiss everything, but rather to build on what is working well, only replacing or ameliorating the unsatisfactory mechanisms; hence, the OConS aims at addressing the challenges which characterise the forthcoming communication environments, while providing proper migration strategies. In this sense, among the main OConS objectives it is worth highlighting: to optimise the multi-path/multi-layer/multi-domain transport and routing, to efficiently exploit networking resources' heterogeneity, to maintain the Quality of Experience (QoE) in highly mobile situations, to support challenged networks through self-* mechanisms, and to effectively and autonomously control the data-centre (inter)connectivity. Accordingly, one of the key cornerstones of the OConS approach is its holistic approach, as opposed to the various disparate proposals from the related work. OConS provides thus a framework to ease the integration of different techniques, protocols, and algorithms, spanning the access networks, the core part, and the data-centres interconnection. The way to achieve this is through an open environment, flexible enough to accommodate the currently available procedures and to suit the needs for the forthcoming ones; therefore, the openness is another distinctive feature and requirement of the OConS approach, being prepared to adapt to the continuous evolution of the technological environment and of the end-users demands.

This document starts by recalling the principles of the current networking architectures, explaining the reasons why they are not always sufficient to cope with the needs of future applications and services, and it proposes a set of requirements and design guidelines to be followed within OConS. Then, it defines the OConS architectural framework with its three functional entities as the elementary building blocks for all the OConS mechanisms and procedures, presenting their internal interfaces as well as the external API towards other networking functionalities (e.g., NetInf and CloNe), and it introduces the various pieces of information needed by the OConS mechanisms. Furthermore, it details several advanced connectivity services provided by OConS, such as: Multi-Path mechanisms that allow the same flow to use multiple simultaneous paths in a fair and efficient way, mechanisms to support Multi-homed nodes for effective handovers and for delivering a given flow to multiple points, Multi-Protocol mechanisms for dynamically selecting different transport protocols and configuring the parameters for a given flow, mechanisms that assist Information-centric networks to benefit from the established multiple paths at the transport and network layers, integration of network-coding and cross-layer techniques to improve the performance of Multi-P* mechanisms, and end-to-end network control for supporting the WAN interconnectivity. Likewise, it depicts a set of mechanisms to manage and control these connectivity services in an efficient and scalable way, and thus it specifically tackles: the dynamic and distributed mobility management, the security aspects in relation with mobility, and the resource management mechanisms (e.g., cognitive radio through spectrum sensing, channel and radio resource allocation, wireless mesh and DTN management, policy routing, and overlaying for data-centre interconnection). Finally, it provides several use-cases examples from the project-wide Flash Crowd scenario, thus applying the OConS framework on realistic cases, showing how the OConS mechanisms are used and what components are required to implement a given use-case. A brief reminder of the prototyping and experimentation activities carried out so far is also provided, together with a self-assessment and future work which will be carried out based on the OConS framework.

Table of Contents

List of Figures

List of Tables

List of Abbreviations, Acronyms, and Definitions

ACS	Ambient Control Space
AN	Access Network
AP	Access Point
API	Application Programming Interface
ARP	Address Resolution Protocol
AS	Autonomous System
BGB	Border Gateway Protocol
BGP	Border Gateway Protocol
BS	Base Station
CDMA	Code Division Multiple Access
CE	Customer Edge
CM	Cluster Manager
CMT	Concurrent Multi-path Transfer
CN	Core Network
CQI	Channel Quality Information
CW	Congestion Window
DC	Data-Centre
DCC	Domain Control Client
DCU	Domain Control Unit
DDC	Distributed Data-Centre
DDMM	Dynamic and Distributed Mobility Management
DE	Decision Making Entity
DTN	Delay Tolerant Network
EE	Executing and Enforcement Entity
eNB	eNodeB
EPC	Evolved Packet Core
EPS	Evolved Packet System
FCP	First Common Parent
FE	Functional Entity
GLL	Generic Link Layer
GMPLS	Generalised Multiprotocol Label Switching
GP	Generic Path
ICN	Information Centric Networking
IE	Information Management Entity
IEC	International Electrotechnical Commission

IP	Internet Protocol
KPI	Key Performance Indicator
LMP	Link Management Protocol
LSP	Label-Switched Path
MAC	Medium Access Control
MAP	Mesh Access Point
MH	Mobile Host
MIH	Media Independent Handover
MM	Mobility Management
MMS	Microsoft Media Server, Protocol
MPLS-TE	MPLS support for Traffic Engineering
MPP	Mesh Point Portal
MRA	Multi-Radio Access
MRRM	Multi-Radio Resource Management
MS	Mobile Station
MT	Mobile Terminal
NA	Network Address
NNI	Network to Network Interface
OConS	Open Connectivity Services
OFDMA	Orthogonal Frequency Division Multiple Access
OMC	Operations and Maintenance Centre
OS	Operating System
OTV	Overlay Transport Virtualisation
OWAMP	One Way Active Measuring Protocol
PCC	Path Computation Client
PCE	Path Computation Element
PDN GW	Packet Data Network Gateway
PE	Provider Edge
QoE	Quality of Experience
QoS	Quality of Service
REST	Representational State Transfer
RM-ODP	Reference Model of Open Distributed Processing
RNC	Radio Network Controller
RSVP	Resource Reservation Protocol
RTMP	Real Time Messaging Protocol
RTSP	Real Time Streaming Protocol
RTT	Round Trip Time

SAPI	Service Access Point Identifier
SCTP	Stream Control Transmission Protocol
SLA	Service Level Agreement
SMF	Security and Mobility Framework
SNR	Signal to Noise Ratio
TCP	Transmission Control Protocol
TDM	Time-Division Multiplexing
TOR	Top of Rack
UE	User Equipment
UMRRM	Unified Mesh Radio Resource Management
UNI	User Network Interface
VLAN	Virtual Local Area Network
VPLS	Virtual Private LAN Services
VPN	Virtual Private Network
WAN	Wide Area Network
WAP	Wireless Access Point
WCMN	Wireless Coded Mesh Network
WLAN	Wireless Local Area Network
WMN	Wireless Mesh Network

1 Introduction

The Open Connectivity Services (OConS) together with the Network of Information (NetInf) in [SAIL-D.B.1] and the Cloud Networking (CloNe) in [SAIL-D.D.1] constitute the three facets of the SAIL (Scalable and Adaptive Internet soLutions) project, where these approaches are combined in different ways (see [SAIL-D.A.2]) to address in a comprehensive manner the Future Internet challenges.

This document presents the first year results from the research work carried out on the Open Connectivity Services (OConS), tackling aspects such as: i) the initial architectural framework, ii) the specific connectivity services and their management mechanisms, and iii) the mapping of these components on the proposed architecture according to several use-cases.

Capitalising on the scenario introduced in [SAIL-D.A.1], we have placed ourselves in a Flash Crowd context where multi-domain heterogeneous technologies coexist (e.g., mesh, self-organised, ad-hoc, 3G/4G, Wi-Fi), where convenient edge-to-edge interoperation and connectivity across the core networks is sought-after (e.g., autonomous data-centre cloud interconnection, intensive data traffic), and where the requirements for communications are dynamically changing (e.g., real-time services, downloading content, uploading on-spot generated data, and so on). Figure 1.1 captures this challenging scenario, showing also the key mechanisms and services which are offered by OConS.

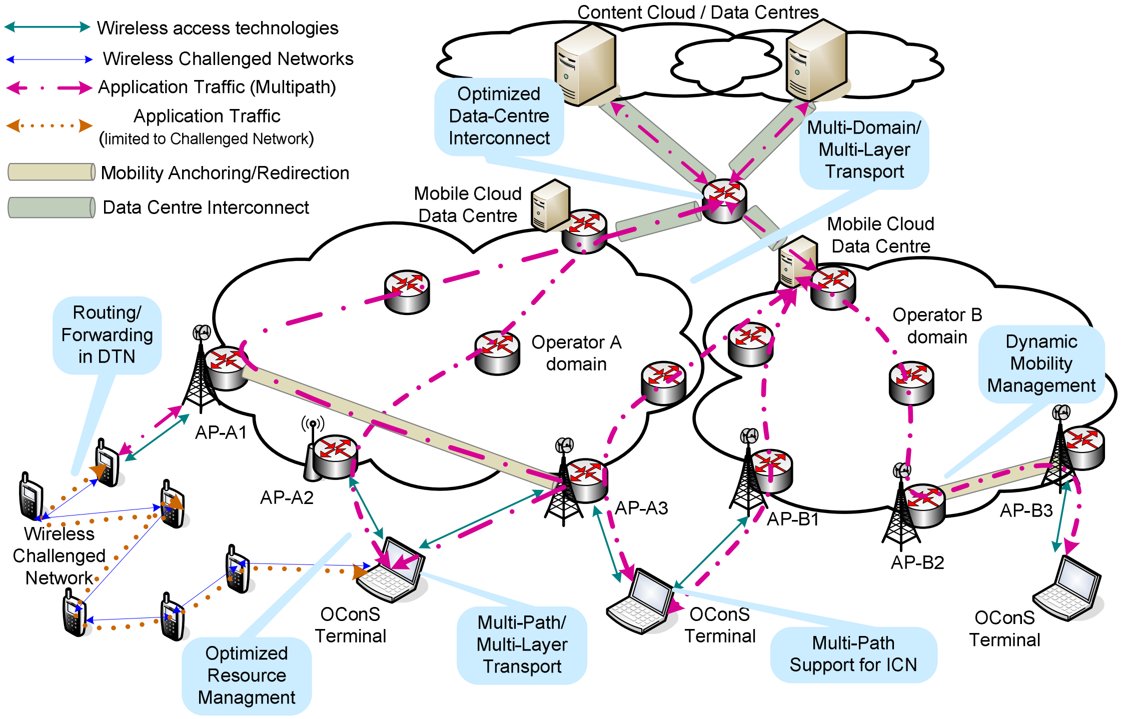

Wireless access technologies
Wireless Challenged Networks
Application Traffic (Multipath)
Application Traffic (limited to Challenged Network)
Mobility Anchoring/Redirection
Data Centre Interconnect
Content Cloud / Data Centres
Optimized Data-Centre Interconnect
Multi-Domain/ Multi-Layer Transport
Mobile Cloud Data Centre
Mobile Cloud Data Centre
Operator A domain
Operator B domain
Routing/ Forwarding in DTN
Dynamic Mobility Management
AP-A1
AP-A2
AP-A3
AP-B1
AP-B2
AP-B3
Wireless Challenged Network
OConS Terminal
OConS Terminal
OConS Terminal
Optimized Resource Managment
Multi-Path/ Multi-Layer Transport
Multi-Path Support for ICN

Figure 1.1: OConS scenario with key mechanisms and services

After surveying the relevant related work, we thus commenced by identifying the challenges arisen from our scenario, such as: dealing with spontaneous topologies through self-* mechanisms, assuring sustainability in Delay Tolerant Networks (DTNs), optimising the multi-path/multi-layer/multi-domain transport and routing, maintaining the Quality of Experience (QoE) in highly mobile situations, efficiently exploiting networking resources' heterogeneity, or effective control and management of connectivity services on User Network Interface (UNI) / Network Node Interface (NNI).

We have subsequently deduced an applicable set of requirements for the OConS, and have proposed the design guidelines to be followed, both the generic ones (such as openness, flexibility, modularity, distributable), but also specific ones pertaining to transport, routing, security, mobility, or resource management.

Considering the aforementioned SAIL project-wide scenario, the stated requirements and the identified design guidelines, we have further developed the OConS architectural framework in Chapter 4. Accordingly, our framework relies on three clearly identified functional entities: the Information Management Entity (IE), the Decision Making Entity (DE), and the Execution and Enforcement Entity (EE). The IEs gather various pieces of information (e.g., measurements, event, etc.), aggregate and process it, and then provide it to the DEs, either upon request or through preconfigured notifications. The DEs use the collected information, the rules, policies and preferences to make appropriate decisions (e.g., resource management, mobility, routing, multi-p transport, and so on), either using a centralised model or a distributed one. Once the decisions are made, they are transmitted to the suitable EEs where they are executed and enforced. Likewise, we have identified the need of an orchestration among the OConS mechanisms and services, and we have shown the flexibility of our approach by depicting both centralised/hierarchical and decentralised/p2p models. Furthermore, we have identified (and started to specify) the open interfaces between the OConS functional entities, as well as the external interfaces, i.e., Application Programming Interface (API), toward the users of the OConS such as the applications, the NetInf, or the CloNe.

The next three chapters present the studied OConS procedures, mechanisms and schemes. Chapter 5 discusses the advanced connectivity services, Chapter 6 elaborates on the traffic steering services, including mobility and security considerations, and Chapter 7 collects the contributions regarding resource management and various enhancements. Accordingly, these three chapters map their innovations into the proposed architectural framework presented in Chapter 4; to facilitate this mapping and to have a consistent approach, we have used a common representation methodology based on the three functional entities abovementioned and their interfaces.

Likewise, the algorithms and mechanisms presented in Chapters 5, 6 and 7 can be generally classified into one of the following two types:

- Those intended to be deployed (and run in real time) on the future networks; they are thus expressed in terms of the architecture entities, components and interfaces, complying with the proposed OConS architectural framework.
- Those for benchmarking, to provide us with a better understanding of dimensioning, provisioning or configuration of the network nodes and elements (i.e., they provide metadata that can be then used for improving the real time algorithms); they are likely to run on a simulated or emulated environment, or even running on an environment that has a different architecture than the one proposed here.

The main focus of Chapter 5 is to present the advanced connectivity services. We introduce these advanced connectivity services because of the need to better cope with the evolution of services and applications, such as content-centric and cloud networking paradigms. In order to achieve this, and also to increase the flexibility of future networks, we have proposed different connectivity services supporting Multi-P mechanisms (such as Multi-Path, Multi-Protocol, or Multi-Layer algorithms), covering the necessary phases for information collection, decision-making and execution. Furthermore, our work includes the integration of Network Coding and Cross Layer techniques to improve the performance of these mechanisms at the transport and network layers.

The focus in Chapter 6 and Chapter 7 is put on several decision-making mechanisms to manage the connectivity in a comprehensive manner (e.g., mobility, resource management, routing/forwarding), on how we can exploit the information collected from the terminals and the network entities, on how to make appropriate decisions for these different aspects of network connectivity, and on how to enforce them back on the network devices and terminals. OConS also provides the means to activate on-the-fly the appropriate mechanisms/protocols and, thus, to react to changes in the networking conditions. Accordingly, after receiving the information from IEs, the DEs decide on the actions to implement in the network and terminals to fulfil the goals of the involved actors (typically users and operators). These goals can be, for example, to guarantee certain QoS for an application flow of a specific user, or to resolve congestion on a specific radio network node. The decision is taken dynamically and in real time, using not only network information (e.g. the conditions of the multiple networks nodes), but also the behaviour of the users and in general the different needs of users and devices.

Chapter 8 provides several examples of use-cases, thus showing how we are applying our framework on real cases, how the proposed OConS technologies and mechanisms are used, and what elementary components are required to implement a given use-case.

Chapter 9 concludes this report with a brief summary of our main findings, providing a self-assessment part, and also presenting the planned future work.

Furthermore, the four common Themes across the SAIL project (i.e., Inter-provider, Security, Management, and Prototyping, see [SAIL-D.A.2]) are widely covered throughout this document. Accordingly, some of the Inter-provider issues related to the OConS cross-domains interactions are discussed in Sections 5.2.1 to 5.2.3, the Security theme is addressed both throughout the guidelines and within Section 6.2, the Management theme spans somehow on several sections but it is more relevant from within Sections 7.1.3 and 7.2.1, and, finally, although the experimentation and prototyping activities are at early stages we have also made progress on these in coordination with the Prototyping theme as presented in Section 9.2.

2 Related Work

Due to the large number of networking aspects and techniques which are covered within the OConS framework, the related work is rather broad. We highlight in this chapter some of the most relevant activities in the related areas which have streamlined the design of the OConS architectural framework, as well as the functionalities and mechanisms related to the ones to be provided by OConS. Thus, it is not meant to be extensive and the reader may refer to the references which are provided to get a more detailed picture on each of these areas.

One of the key cornerstones of the OConS approach is its holistic approach; in this sense, as opposed to the various proposals and works which are cited hereinafter, the OConS provides a common wrapper so as to ease the process of integrating different techniques, protocols and algorithms, ranging for the access to the core parts of the network and to the interconnection of data-centres, thus facilitating the interoperation among them.

2.1 Heterogeneous Access Networks

In this case, there are mainly two lines of work to be highlighted: access selection in heterogeneous scenarios and the related activities of the relevant standard bodies.

The first one corresponds to the work carried out (mostly) by different research initiatives which aimed at proposing novel architectures to deal with access selection in heterogeneous scenarios. In this realm, one of the most relevant proposals was the EU Ambient Networks Project [Niebert04], [Niebert07]. The Ambient Networks designed a networking architecture, aiming at leveraging the cooperation between different networks, embracing mobility, context-awareness, security, and other control functions. For that, a novel control plane, the so-called Ambient Control Space (ACS) was introduced [Schieder07], establishing all the required interfaces to enable the interconnection between peer Functional Entities (FEs) [Kappler07]. One of the cornerstones of the ACS was the Multi-Radio Access (MRA) architecture [Sachs06], [Johnsson06], [Sachs07], which dealt with the management of the available resources. In this sense, it received the requests from the applications and services (to establish a flow), and considering the particular requirements of such flows, the corresponding policies (user, operator, etc), as well as the situation of the available networks, selects the most appropriate access interface to be used. The MRA had two main entities, namely the Multi-Radio Resource Management (MRRM) and the Generic Link Layer (GLL), which are briefly discussed below, but its operation was heavily related with other functional entities (mostly related with mobility tasks).

- The GLL facilitates a transparent and dynamic management of the set of wireless interfaces which a terminal might be using at any time. In this sense, it offers information about the available resources, so that other entities may use them, especially the MRRM when executing its access selection algorithms. For that, the GLL abstracts the information from the corresponding RATs (including, amongst others, link quality, load information, etc.), thus making possible the fair comparison between them.
- The MRRM is the main control entity of the multi-access Ambient Networks architecture. It performs the joint management of the radio resources within heterogeneous network environments, by selecting the optimum access each time. Using the services provided by the GLL, the MRRM monitors the available networks, and collects information about status of the existing links (in terms of their quality and the availability of resources). Based on this information (and the particular requirements of the current services/applications), the MRRM executes the access selection algorithms, which might require a handover.

The second line of work which is worth highlighting here, concerns the efforts taken by the relevant standardisation bodies. For instance, the Evolved Packet System (EPS), see [TR23.401], [TR23.402], is part of the ongoing 3GPP architecture standardisation, both for the new radio access part – generally referred to as Long Term Evolution (LTE) – and into the

Internet Protocol (IP) based Evolved Packet Core (EPC). A flat architecture for the access is used: Radio Network Controller (RNC) and node-B (NB) are now collapsed into one single entity, the evolved node-B (eNB). The EPC supports both the existing 3GPP accesses (i.e., 2G/3G) as well as the interworking between 3GPP and non-3GPP accesses (e.g. Wi-Fi). To ease the migration and to support different deployment scenarios, EPS proposes a split between the control plane (eNB, MME) and the user plane (eNB, SGW/PGW). Within the EPS domain, one or more EPS bearers (equivalent with a connection or a path) are used to provide different QoS levels. Mobility management is implemented with GTP tunnels, but also with IETF technologies such as MIP and PMIP for interworking scenarios.

In addition, the IEEE 802.21 Media Independent Handover (MIH) standard [IEEE802.21] defines media access independent mechanisms aiming at enabling seamless handovers between IEEE 802 (802.11, 802.16, or 802.3) systems and non-IEEE 802 (e.g. 3GPP, 3GPP2) cellular systems. Both horizontal (i.e. within a same access) and vertical (i.e. between different access technologies) handovers are addressed by the IEEE 802.21, supporting both terminal-initiated and network-initiated handovers. Three types of Media Independent Services are provided: Event Service, Command Service and Information Service. The goal of IEEE 802.21 is to bring intelligence from the link layers and make it available to the MIH user, which may be any logical entity requiring MIH services and interacting with them (such as a Layer 3 or higher mobility protocol, a handover decision module, applications, and so on). The OConS goes beyond the current scope of the MIHF framework, since it will also consider policies/rules and requirements coming from other parts of the network, rather than from the subjacent link layer technologies only.

In general, the OConS approach considers some issues which were not tackled by previous initiatives, like the use of virtualised resources. Additionally, the decision process will be distributed amongst various entities (as opposed to the traditional centralised approaches) and, what it is more relevant, it will consider elements which go beyond the access networks, taking advantage of the global view fostered by the OConS architecture.

2.2 Core Networking Techniques

In this group there are various initiatives to be highlighted. Going upwards in the protocol stack, the Generalised Multiprotocol Label Switching (GMPLS) is intended to bridge the gap between the lower layer (e.g. optical) transport infrastructure and the IP layer. GMPLS is also designed to enable multivendor interoperability and multilayer functionality. The document [RFC3945] describes the GMPLS architecture. GMPLS extends the functionality of Multiprotocol Label Switching (MPLS) to include a wider range of label-switched path (LSP) options for a variety of network devices. Traditional MPLS is designed to carry Layer 3 IP traffic by establishing IP-based paths and associating these paths with arbitrarily assigned labels. These labels can either be configured explicitly by a network administrator or dynamically assigned by a protocol such as the Label Distribution Protocol or Resource Reservation Protocol (RSVP). In contrast, GMPLS supports various types of Layer 1 through Layer 3 traffic. GMPLS labels and LSPs can be processed at four levels: Fibre-Switched Capable, Lambda-Switched Capable, Time-Division Multiplexing Capable (TDM), and Packet-Switched Capable (PSC). Thus, the GMPLS labelling is more flexible than MPLS, as it can be used to represent a TDM time slot, a Dense Wavelength Division Multiplexing wavelength (also known as a lambda), or a physical port number. To enable multilayer LSPs, GMPLS uses a number of techniques:

- Separation of the control plane from the data channel through the Link Management Protocol (LMP) [RFC4204], to manage both control and data channels between GMPLS peers.
- RSVP-TE extensions for GMPLS [RFC3473], [RFC4208] to request path setup for non packet LSPs (wavelengths, time slots, and fibres).

- OSPF extensions for GMPLS [RFC4203], [RFC5392] to also route packets to virtual peer interfaces defined in an LMP configuration.

In fact, GMPLS provides a set of control and management functionalities, not being directly involved in data exchange.

Likewise, the automatic switched optical network architecture [ITU-G8080] describes control plane functions on layer networks [ITU-G805], with interoperable procedures for requesting and establishing dynamic connection services across heterogeneous technologies and domains, thus requiring the specification of several reference points, such as E-NNI, I-NNI, and UNI. Domains are established by operators' policies and have a range of membership criteria; i.e., a domain represents a collection of entities grouped for a particular purpose. A control domain is a type of transport domain, where the criterion for membership is the scope of a control plane component responsible for the transport resources within the transport domain. The architectural model of layer networks is described in [ITU-G800], which is a unified view of both connectionless and connection-oriented networks. It is a superset of [ITU-G809] and [ITU-G805]. In all three Recommendations, the relationship between layer networks is described in client/server terms. While [ITU-G805] presents only a view of resources, [ITU-G809] and especially [ITU-G800] describe forwarding as a necessary function to layers. When two layer networks are in a client/server relationship, this constitutes a multi-layer network. By definition, client/server relationships are recursive so that more than two layers can comprise a multi-layer network. Control plane entities are described in [ITU-G8080], incorporating changes to control plane components from [ITU-G800] packet layers.

In addition, the IETF TRILL standard (or IEEE 802.1aq Shortest Path Bridging), currently under study, provides a method of interconnecting links that combines the advantages of bridging and routing [Perlman04]. It has the potential to unify two of the most relevant protocols to be used for the interconnection of data centres: Virtual Private LAN Services (VPLS) and Overlay Transport Virtualisation (OTV) (see next section for a more thorough description of these solutions). TRILL keeps Ethernet's dynamic data-plane learning mechanisms intact. However, flooding is now controlled by use of distribution trees and hop counting. The net effect of flooding is significantly reduced due to the fact that topology changes do not flush the MAC address tables. TRILL networks are easier to troubleshoot, as every RBridge associates the "flat" MAC address with the "location" in the network defined via the remote bridge name. Lastly, the problem of address table growth is somewhat resolved, due to the fact that MAC addresses need not to be known on every switch in the domain, but only on the switches that actually have connection to the end equipment [OTV].

Although the capacity of transport networks is continually increasing, the stringent requirements coming from the new applications and services demand solutions for guaranteeing appropriate quality of service levels. The Path Computation Element (PCE) architecture [RFC4655] has been introduced to provide effective Traffic Engineering solutions, i.e., to cope effectively with complex constraint-based path computations. The detailed list of RFCs and drafts can be found at IETF PCE Working Group. The main motivations that drove the introduction of the PCE architecture included the need to perform CPU-intensive path computations and to deal with several scenarios where the node responsible for path computation has limited visibility of the network topology and resources (e.g., multi-domain and multi-layer networks). The architecture relies on the PCE (i.e. an entity, component, application, or network node) that is capable of computing a network path or route based on a network graph and applying computational constraints, and the Path Computation Client (PCC) defined as any client application requesting a path computation to be performed by a PCE. Communication between PCC and PCE is guaranteed by the PCE Communication Protocol (PCEP). PCE-based routing architectures for multi-domain networks can be classified [Chamania09] into two major groups: Peer-to-peer and Hierarchical. For example, [RFC5441] proposes the Backward Recursive Path Computation (BRPC) protocol to compute optimal inter-domain paths in a multi-domain network, while [RFC5151] describes procedures and protocol extensions for the use of RSVP-TE signalling in MPLS/GMPLS-TE packet networks to support establishment and maintenance of LSPs that cross domain boundaries.

The BroadBand Forum (BBF) has also investigated the Next Generation Transport and Services using the so called unified MPLS approach, to examine how the MPLS technology can be applied in Broadband Multi-Service Architectures. They also describe MPLS extensions needed to support the MPLS transport profile (MPLS-TP) and they further looked at how specific MPLS architectural elements and mechanisms can be used to enable a particular service or application. Unified MPLS can be used from the core through the aggregation to the access network, providing a flexible and scalable network architecture thereby interconnecting different network domains.

Finally, we could highlight the efforts on the multipath transport protocols realm. This has become an extensive and diversified research area, with various proposals ranging from modifying the currently prevalent TCP protocol [Han06] to proposing generic transport for the Future Internet [Ford08], and they can be applied to different layers or entities, such as routing and transport protocols or applications (e.g. in a peer to peer overlay) or anywhere in-between. The MPTCP WG is standardising TCP extensions for multipath support, requiring substantial modifications of the standard TCP protocol which does not have multihoming support. A suite of drafts and RFCs deal with the MPTCP activities, with the main proposals addressing: the architecture [RFC6182], the management of multiple IP addresses [Ford11], the congestion [Raiciu11] and the threats [RFC6181]. The SCTP-related multipath extensions, broadly grouped under the term Concurrent Multipath Transfer (CMT), provide means to manage concurrent paths [Iyengar06]. There are a number of related IETF documents which contribute to this area. [Dreibholz11] specifies extensions to the SCTP sockets API for configuring the CMT-SCTP and resource pooling SCTP extensions. Load Sharing for SCTP [Becke10] proposes changes to standard SCTP congestion control to handle multipath.

As has been seen, there exist a large number of different proposals; OConS aims at providing the grounds so that we can instantiate the most appropriate one, depending on the particular needs at any particular time.

2.3 Data-Centre Interconnection Technologies

When it comes to data-centre interconnection technologies, see [Labovitz10], most Internet inter-domain traffic is exchanged directly between large content providers, content delivery networks (CDNs), and consumer networks. Data-centres often employ Ethernet as transport technology, and they use server virtualisation in order to run several virtual machines on one physical host. If a large number of these nodes are attached to a flat data-centre network, the broadcast traffic caused by the Address Resolution Protocol (ARP) can result in scalability issues [Dunbar10]. Although the scalability of address resolution in Ethernet networks can be improved by partitioning the Ethernet network, e. g., into smaller Virtual Local Area Networks (VLANs), mechanisms like redundancy, load sharing, or virtual machine mobility require that a large subset of nodes is in the same VLAN. Accordingly, many different solutions to tunnel Ethernet frames over a Wide Area Network (WAN) have been proposed, e.g., see [Knight04]. Two prominent protocol examples are VPLS [RFC4762] and OTV [Grover10], [Cisco-OTV]. VPLS uses MPLS tunnelling and data plane learning along with flooding of unknown Ethernet frames to connect customer networks. OTV in contrast uses IP tunnelling and a separate overlay control plane to connect Ethernet customer networks over an IP core network. Both technologies thus represent two different principles, addressing the problem of realisation of Ethernet address resolution [Klein11].

We are exploring OpenFlow and FlowVisor (see next section) as possible alternatives to implement data centre interconnectivity (as described in the next section). We are defining interfaces and extending the functionality they provide in order to implement inter-domain communication. This will provide a proof of concept for these technologies as OConS instantiations for data centre interconnection.

2.4 OpenFlow and ForCES

The OpenFlow novel concept deserves some particular considerations. The discussion above leads to a clear conclusion regarding the large number of networking concepts which are recently popping up. Starting from the observation that these newly conceived networking concepts can barely be deployed and tested, the OpenFlow [OpenFlow] framework has recently taken roots. Thus, the idea of OpenFlow is to make commercially-deployed networks programmable by manipulation of the entries of the flow table, e.g., in an Ethernet switch via an open interface implemented by the OpenFlow protocol. This way, it becomes possible to control the network traffic more easily. The OpenFlow protocol is implemented by the switch vendor itself and integrated into the firmware, therefore the internal functionality of the switch do not need to be systematically exposed. The OpenFlow architecture consists of OpenFlow-enabled Ethernet switches, the OpenFlow protocol which is SSL-encrypted and one or more OpenFlow controllers running the controller software. If an OpenFlow switch receives a packet for which no matching entry in the flow table of the switch is found, then, the incoming packet is not flooded but it is sent to the OpenFlow controller using the OpenFlow protocol. There the controller software handles this packet according to its coded logic and processes it, e.g. sends it back to the switch in order to output it to a certain port or establishes a new rule to be inserted into one or several OpenFlow switches. There are several controllers widely available, e.g. the reference one from Stanford University or the NOX controller from [Nicira]. Also there exists the FlowVisor, see [OpenFlow], which is a special purpose OpenFlow switching controller that acts as a transparent proxy between OpenFlow switches and multiple OpenFlow controllers. FlowVisor creates slices of network resources (bandwidth, CPUs, FIB-tables, topology) and delegate control of each slice to a different controller.

A quite similar approach has been also proposed in the framework of the ForCES IETF working group, which has specified the Forwarding and Control Element Separation (ForCES) protocol. The ForCES protocol is used to standardise the information exchange between Control Elements and Forwarding Elements placed within a ForCES Network Element, as well as between the Control Elements themselves. The ForCES requirements activity [RFC3654] originates at least back to 2003. Although the ForCES protocol [RFC5810] was finalised in March 2010, it is not widely in use by vendors.

In order to achieve the desired OConS functionalities, OConS focuses on the development and adaption of both mature and brand new technologies. We are exploring OpenFlow as an alternative to implement data centre interconnection use case as a Multi-domain environment. As of today, OpenFlow is missing an inter-domain interface. We are developing an inter-domain interface for OpenFlow. This includes prototyping interfaces for control and data planes, defining types and formatting for data exchange and studying the integration with other technologies.

We are also working on the mapping between the slicing concept provided by the FlowVisor project and the connectivity resources required by the OConS use cases involving inter-domain communications. We are expanding FlowVisor's single-domain slicing functionalities to provide the controlled inter-domain resources connectivity needed in the inter-domain use case. At the same time, we keep our solution updated with the rapidly developing OpenFlow eco-system.

2.5 EU-IST 4WARD – Generic Path

To better cope with Future Internet requirements, the 4WARD project (see [4WARD-D5.1] and [4WARD-D5.2]) developed a clean-slate architectural framework based on the Generic Path (GP) concept. The main objective of the Generic Path (GP) model was the support of various communications needs in highly mobile and dynamic networking conditions, while adapting the end-to-end transport and QoS procedures to the capabilities of the underlying networks. In addition, it also benefits from paths diversity over multiple routes as well as the inclusion of advanced techniques such as network coding.

Considering that classical layering approaches, like the ones from Internet or the ISO/OSI model, do not reflect the reality of today's networks, the GP applies an object oriented approach where path composition between end-points can be built recursively on any given set of (sub-)GP classes and instantiated objects.

Only a minimum set of functional blocks are specified; their re-usability through object class inheritance, attributes and instantiations leading to a recursive network architecture. These basic blocks are: the *Entity*, an object capable of producing, processing or consuming information; the *End-point,* an identifiable data entry/exit points for a GP service; and the *Mediation point* along a service path where configuration operations related to the services provided are executed (e.g. paths duplication or forwarding, data trans-coding, and so on). Likewise, the GPs are organised in control domains, named *Compartments*.

The 4WARD project also provided more insights on how implementing routing protocols, coding and cooperation frameworks, mobility mechanisms and the actual sharing of physical resources (see, e.g., the evaluation results from [4WARD-D5.3]. More specifically, different approaches supporting mobility and multi-homing (either network or end-to-end based), were compared; a framework that helps in detecting and reacting to network situations by means of cooperative coding was proposed; an ontology to characterise available resources and services was designed; finally, various aspects from Mesh Networks were also outlined, showing how new functionality can be brought into existing networks much easier with the GP concept and a recursive network architecture.

Despite the interesting properties of the GP concept from 4WARD, its adaptation to SAIL is questionable. Since GP implemented a clean-slate approach, migration strategies were not considered. Consequently, the GP concept as a whole cannot be used as it is; instead, we might consider adapting some elements into the SAIL solutions.

2.6 US-FIA Related Projects

In the following, we briefly summarise some of the most relevant aspects of those US-FIA projects which share some of the objectives and goals which are being pursued by the OConS framework.

The NEBULA network architecture [NEBULA] is a clean-slate Future Internet approach mainly motivated by the cloud computing paradigm. They are dealing with challenges coming from the technology evolution but also assuring economic and policy/regulatory viability. NEBULA architecture is built upon three pillars: (1) NEBULA Virtual and Extensible Networking Techniques (NVENT) which is a distributed control plane that provides access to the network services and abstractions as required by an application (such as policy routing and multipath routing); (2) NEBULA Data Plane to establish and enforce the policy-compliant paths according to the received policies from the NVENT, and proving that an administrative domain has authorised a given path and also that a certain packet has followed that path; and (3) NEBULA Core that redundantly interconnects enterprise data-centres containing replicated data with ultra-high availability next-generation core routers. Technically speaking, NEBULA requires thus new interfaces between control and data planes to specify the appropriate policies for each application and service, prior to packet flows/path establishment. Likewise, enforcing these policy decisions in the data-plane requires new per-packet authentication techniques.

ARCHSTONE (Advanced Resource Computation for Hybrid Service and TOpology NEtworks) project [ARCHSTONE] develops technologies to enable better resource computation and provisioning across multi-layer networking architectures. Accordingly, its main goal is the dynamical and flexible creation of "slices" of networking resources across multiple network layers, thus generating virtual network topologies. They have proposed a "Multi-layer/Multi-dimensional Topology Computation Element" (MX-TCE) which serves as an advanced path computation element, extending the concept of path computation to multi-layer, multi-dimensional scenarios. Through a well-defined Network Service Interface, they intend to provide several “atomic” Network Services (such as Connection Service, Topology Service, Monitoring Service, Measurement Service, or Resource Computation Service), but also “composite” Network Services which are built from several atomic network services (by using appropriate policies, schedulers and relation operators/workflows among them). Finally, the project has strong liaisons with the Open Grid Forum Network Services Interface WG, as well as being an active supporter of the Inter-Domain Controller Protocol.

MobilityFirst project, see [MobilityFirst], considers the Mobility as the key driver for the Future Internet (together with the Robustness and the Trustworthiness), and thus they mention as design goals the following: more efficient Host/Network Mobility and Content Addressability, scalable Location Service for both mobile hosts and content (using both flat and hierarchical names), enabling Path Diversity for the end-to-end Routing (through multi-homing and multi-path), as well as assuring better Security and Privacy; they also stress the importance of having easily evolvable network services**.** To achieve its goals, MobilityFirst has proposed several architectural components, such as: a) Decentralised Naming/Location Service to dynamically map in real-time a self-certifying global identifier of users, devices, or content) to the network addresses (NA); b) Mobility Service is provided either by updating the Location service with the new NA, or using a home-agent-like function in the old NA domain to redirect the traffic to the new NA; c) Disruption-tolerant Routing by using a generalised DTN (i.e., storage-aware) mechanism; d) Segmented Transport Service with path diversity (besides end-to-end transport); e) Intentional Data Receipt to enable a receiving host to specify its receipt policies towards the network; f) Resource queries and allocation mechanisms, as well as mechanisms to provide feedback on network conditions; g) Support of Context-Aware Pervasive (and Mobile) Services together with more users/terminals involvement than today; and h) Inclusion of the Computing and Storage resources through a virtualised and programmable computing layer. Finally, they have also investigated (and apparently backed) the possibility of having a separate control/management within their framework.

3 Requirements and Guidelines for OConS

3.1 Requirements

Requirements are the characteristics (e.g. behaviour and performances) expected from the system under design/development, usually independent of a given solution and expressing well quantified/measurable/testable criteria. In this chapter we are thus elaborating on the general requirements for Open Connectivity Services, but also on those specifically related with routing, transport, security, mobility and resource management.

It is widely recognised now that traditional networking approaches are starting to show their limits, and in addition, patches and evolutions of currently available architectures are deemed insufficient. Therefore, novel architectures have been proposed in the recent years, some of them even being clean-slate approaches. However, a clean-slate approach may also come with disadvantages, such as the need for valid migration strategies to ensure a relatively quick rollout and deployment, and the risk of improving some aspect of the network while creating unforeseen new problems. It is therefore widely recommended not to dismiss everything, but rather to build on what is working well, only replacing or ameliorating the unsatisfactory mechanisms or protocols. OConS aims thus at addressing the challenges which characterise the upcoming communication environments, while providing a sound migration strategy.

3.1.1 General Requirements

The OConS framework aims at tackling some of the most relevant challenges which are posed by new communication paradigms, also brought about by the so-called Future Internet.

The way forward is therefore to foster an open environment, flexible enough to accommodate most of the currently available procedures and to suit the needs for the forthcoming ones. This openness is the most distinctive feature (and requirement) of the OConS approach, which will need to tackle some additional aspects, which are briefly introduced below.

The architecture should be able to adapt to the rapid evolution of the communication technologies and related processes. This flexibility shall also span to the dynamic creation of mechanisms and services on an autonomous manner (self-healing, self-configuration, etc.); if possible, this creation should be based on the activation and de-activation of the already existing modules.

Last, but not least, and from a general perspective, it is of outer relevance to highlight the need of a distributed/collaborative architecture. Centralised approaches, albeit reducing the inherent system complexity, might lead to scalability and robustness issues and, therefore, considering the rapid growth of nodes might become unacceptable. A direct consequence of this distributed approach is that the system should provide the means to discover its features and available services.

These generic requirements, together with the well-known ones such as resilience, scalability and manageability, can be applied to any OConS functionality we could think of. Furthermore, in the rest of this section we also discuss the specific challenges of the technical areas OConS particularly deals with, namely routing, transport, mobility, resource management and security. The identified requirements serve as the basis to propose the barebones of the OConS architecture, as described in Chapter 4.

3.1.2 Requirements for Routing

From a routing perspective, mechanisms must address general expected requirements linked to other globally desired features such as: (i) the suitability of strategies even under conditions of mobility, (ii) the consideration of security as a primary concern within the design phase of the strategies, and (iii) the concept of multi-path as the norm rather than the exception when deciding the routing strategy.

Routing in OConS is supposed to tackle the general needs assumed by routing in heterogeneous environments, but we also aim at dealing with some more specific requirements, as follows:

- Provide end-to-end routing across heterogeneous physical technologies such as optical, wireless, or copper based networks.
- Support multi-domain routing:
  - Routing shall be implemented using a multi-domain paradigm. Domains constrain routing information and provide an aggregated view of the infrastructure they represent. Due to their different nature, domains can be divided in following categories:
    1. Administrative domain, i.e., separating entities that operate the network (such as Network Operators or departments thereof);
    2. Policy domains, thus grouping different areas of a network depending on their functionality (e.g., access, core, or based on Service Level Agreement (SLA) / Quality of Service (QoS) requirements, etc.);
    3. Trust domains, where different security models and policies apply.
  - Regarding the orchestration of multiple domains, these must exchange comparable tokens of information and to do so, an initial handshake phase is needed to establish the nature of the information tokens exchanged. Domains must exchange information in a secure way.
- Provide novel techniques for multi-path routing, shifting the actual expectations to the combination of several simultaneous paths available for a split transmission or service. In addition, algorithms are required to see when and to what extent the multi-path routing needs to be deployed for a given flow.
- Innovative topologies and deployments demand new routing strategies that take advantage of specific features on those types of network and can derive routing approaches that are able to self-adapt to changing conditions. Self-learning techniques applied to DTN and challenged networks may provide valuable information for the routing, not previously considered (useless) in a traditional topology. Behavioural patterns of mobile nodes, for instance, are a good example if we consider human mobility.
- The global end-to-end routing will also demand support for effective communications among entities on different layers, and thus cross-layer signalling, since this is expected to be one key strategy for developing innovative routing and forwarding approaches.
- Distributed environments, not based on traditional infrastructure, but formed by nodes of the same nature with similar roles and responsibilities, are more familiar with hop-by-hop views than with end-to-end requirements, so both need to be considered as a requirement for the sake of heterogeneity and completeness.

3.1.3 Requirements for Transport

The requirements for transport encompass support for a wide range of flexible solutions to enable efficient and optimised services. These requirements may be organised into the following broad areas.

Support for multiple paths, within the novel requirements of edge-to-edge, where the transport services are delivered between the network edges. Such delineators may be defined as any set of end points (locators), which may be associated by a multi-homed device (which simultaneously supports multiple network interfaces), by a network cloud i.e. a number of end points which may include a single administrative domain. More broadly, the edges may also be defined by a set of multi-domain end points including a number of multiple administrative domains, policy domains, trust domains, etc. To support legacy transport, please note that the edge can also be assimilated with a single end-point.

We also need optimised multi-path transport to support applications with heterogeneous content, by enabling customisable transport parameters within selected paths. These include congestion control type, reliability and in-order delivery options to best suit the particular content types. Multi-path transport requires also fair and efficient congestion control algorithms that use all (or some) of the available paths, to increase the multipath flow throughput without hurting concurrent, legacy flows.

In DTN and other Wireless Challenged Networks the concept of traditional transport does not hold any more. Communication is deployed in a hop-by-hop basis in such a way that each transmission has neighbouring nodes as source and destination, and there are no flow or congestion control mechanisms, nor end-to-end path establishment. Otherwise, provided that buffer and memory are normally scarce resources, there is a need for optimising their availability, so that transmission algorithms are able to manage their use in an efficient way.

For the data-centre interconnect case, the two prominent solutions for tunnelling the Ethernet frames over WANs were discussed in Section 2.3, namely the Virtual Private LAN Services (VPLS) [RFC4762] and the Overlay Transport Virtualisation (OTV) [Cisco-OTV]. The VPLS uses MPLS tunnelling and data plane learning along with flooding of unknown Ethernet frames to connect customer networks, so we need to better control and reduce the amount of flooding required. The OTV in contrast uses IP tunnelling and a separate overlay control plane to connect Ethernet customer networks over an IP core network, and although it reduces the flooding in the core, its convergence process can be too slower in case of topology changes.

3.1.4 Requirements for Security

The OConS mechanisms should ensure that it cannot be misused such that the system integrity is endangered. Requirements regarding security are identified as security objectives describing protection targets according to some security policy [RFC4949].

To make this happen, security services will be used to describe the security objectives of the main OConS mechanisms addressed, namely support for advanced mobility management, transport functionality and network resource management. The security objectives identified for mobility management specifically are:

- Legitimate use of the advanced mobility management.
- Misuse prevention and thus the availability of mobility management capabilities.
- Accountability of having used mobility management functionality.

The accountability of having used such mobility management functionality is already a debatable security objective, as it may challenge privacy concerns. The data handled may be used to derive information about communication behaviour profiles, if this data is technically not protected. On the transport side, necessary security services have to:

- Ensure the availability of functions and elements enabling the transport capabilities.
- Accountability is highly desirable, although the extent to which privacy concerns are enforced may set some limitations.

Concerning the network resource management, last not least, related security services then should ensure

- Legitimate use of resources to prevent misuse, and
- Integrity of infrastructural services needed to manage such resources to assist availability.

The SAIL project does deliberately not follow a clean slate approach. For this reason OConS mechanisms build partially on existing functionality and infrastructure services (routing, name resolutions, etc.) that are outside the research scope. OConS mechanisms may hence inherit also security properties or even shortcomings because of the project scope.

3.1.5 Requirements for Mobility

On one hand, the consumers (i.e., end-users) have already a multitude of devices to communicate through a range of different heterogeneous networks, each one with specific connectivity services (e.g. different mobility approaches). We need thus to inherently support the multi-access (i.e., L1/L2 technologies) and the multi-homing (i.e., several L3 addresses). Moreover, several business models may exist in parallel for the same user, and thus the support of the mobility in a multi-domain scenario is also a prerequisite (e.g., different administrative domains). Then the consumers want their applications' flows to conveniently and transparently switch from one network to another, and therefore they require service continuity or even seamless handover for certain flows (e.g. voice). Likewise, they also need to be always reachable and be provided with consistent and personalised services, i.e. awareness of their location and network capabilities as well as speeding-up the adaptation of higher layers (L4 and above) to acceptable levels.

On the other hand, our framework needs to offer connectivity services that are profitable for operators. Accordingly, the necessary means for flexible deployment and operation need to be in place; e.g., providing the mobility-as-a-service only when needed, minimising the protocol encapsulation/tunnelling, and minimising the mobility context necessary within the network nodes. In addition, as we have mentioned above, the users will likely need per-flow approaches for mobility decision and execution, which in turn imply specific procedures such as per-flow mobility anchor selection and activation depending on a given communication context (type of application, user preferences, terminal capabilities, radio environment, etc.) and on mobility patterns. Finally, to cope more easily with the introduction of new connectivity services and the gradual expansion in network capacity, we reckon that the support for decentralised approaches for mobility (i.e., both decision and execution) is also required.

3.1.6 Requirements for Resource Management

Within the heterogeneity of deployed networks a key requirement is the seamless integration of resources control and management, from the edge. To achieve it, the context of the application shall be used, to fulfil the requirements of advanced networked applications (e.g., content distribution, cloud computing). An important aspect is to exploit diversity (e.g., random variations in channel quality or structural differences in channel properties like different delay/data-rate trade-offs) existing over different communication technologies between two end-points, aiming at a dynamic and seamless switching between technologies as the flow's required data rate changes.

The availability of end-to-end abstraction of network resources and features is also an important requirement, supported and not limited by the heterogeneity of technologies. The support of virtualised resources shall also be considered. On the other side, the exploitation of the particular network resources and features of each technology and their combination on an end-to-end perspective shall be addressed.

The cooperative planning, operation, control and management of connectivity services and technologies is another requirement, e.g., by establishing the appropriate interactions among them. This enables better network efficiency, resilience, scalability and future evolution. It shall leverage advanced features of link technologies, making use of network diversity.

The support of self-organised and distributed resource management is an important requirement, creating and sustaining the connectivity in wireless challenged networks. The concept of self-organisation embraces self-configuration of newly added nodes in a plug-and-play fashion, self-optimisation of resources, and self-healing in the event of failures. Energy and spectrum efficiency shall be addressed. The resource management of such wireless networks shall be supported by cognitive radio and spectrum sensing, and mechanisms shall be energy efficient in the management of resources. On the other side, management of resources should be dynamic and adaptive to changes in the network.

3.2 Design Guidelines

Current design paradigms for networking architectures are fading away; they are becoming unfitted and unable to cope with the requirements which characterise nowadays applications and services. These design principles (mostly affecting the IP-based core networks) are enumerated below (see [FIArch11] and references therein for a comprehensive discussion):

- Network of collaborating networks (e.g., inter-net-working via gateways)
- Connectionless (i.e., best-effort) IP-datagram forwarding and maximum sharing of the routing information (i.e., routing tables in each router)
- End-to-end transport principle (accordingly, most of the complexity is kept within end-nodes, e.g., TCP, SCTP, HTTP)
- Modularisation (i.e. layering) with loose-coupling (i.e., weak cross-layers interactions)
- Locality principle (i.e., local causes result in local effects)
- Simplicity principle (e.g. cost-effectiveness)
- End-points are identified by node locators/addresses, i.e., not by their names
- Security was retrofitted with more or less success, i.e., not aiming at Security-by-design

As for the mobile access part (i.e., 3GPP), we can mention the following design guidelines:

- Usually having a business relation with a single mobile operator for each end-device
- Mostly centralised and quite static decision models for mobility and QoS policies
- Access network selection/handover decisions are User Equipment (UE) driven, but strictly network controlled
- Connectivity services not always adapted to the networking context and the applications

3.2.1 Challenging the current design guidelines and principles

Most of the current solutions for managing the connectivity services (such as data-transport, routing, mobility, QoS) deal with rather concrete aspects of the whole problem. For example, they are either focusing on the establishment/maintenance of an end-to-end flow (but sometimes still related to specific IP realms), while others are concentrating on the particular issues which affect the core or the access part connectivity (i.e., the mobile-fixed dichotomy).

Accordingly, some of the guidelines and principles which have shaped the current solutions should be, at the very least, revisited in order to see whether they are appropriate to deal with the challenges and requirements presented earlier.

Thus, in our view, the first architectural design guideline to be followed by the OConS architecture is a holistic approach to the networking. Likewise, the openness, which intrinsically characterises the OConS approach, calls for much more comprehensive approaches to address the overall connectivity issues.

Then, from the point of view of the internetworking perspective, some of the traditional design principles, which have been more or less successfully used so far in the Internet, are no longer valid. E.g., the IP datagram forwarding (and its routing) is based on a best-effort approach, which makes it unsuitable when it comes to resource management, multi-path, quality of service, mobility, and so on. Traditional approaches, like DiffServ/IntServ, did not completely solve the problem and the situation has become more challenging with the appearance of the optical networks and the increase of wireless access alternatives.

Another cornerstone of the traditional Internet architecture is (or better, it was) the end-to-end principle. Nowadays, the communications do not necessarily need to involve always a source and a destination pair, e.g. see NetInf scenarios with some illustrative examples. Therefore, this end-to-end principle needs also to be revisited, because appropriate handling of these new communication paradigms (information-centric, content-based, x-cast) require more active participations of the intermediate elements (e.g. routers, caches, edge-devices), which is not yet commonly used in today's approaches.

Furthermore, although the TCP/IP Internet-model has already broken the strict layering approach (which was backed by the ISO in the 80s), it still uses a layered approach; thus, several steps towards extensive and consistent cross-layer solutions need to be taken.

Another significant difference to the original approach comes from the heterogeneity which currently characterises the networking solutions; the increasing relevance of the so-called Internet of Things (e.g., sensor networks, M2M) brings into play a wider range of devices which, despite having completely different requirements and characteristics, need to be interconnected somehow. Therefore, the future connectivity solutions may need to be aware of the different needs of the interconnected hosts and devices, in order to offer appropriate and tailored resources and capabilities.

Additionally, experience tells us that security solutions have been usually retrofitted into already running/in-production architectures and solutions. This patchwork approach has not been always efficient and, despite the fact that security is not the focus of the WPC, we must provide, from early design phases, appropriate consideration for the security aspects.

On the other hand, considering the (increasingly ubiquitous) wireless accesses, there are also additional guidelines which need to be considered or reconsidered in order to build connectivity solutions which are able to satisfy the previously described requirements. To start with, the progressive embracement of new access technologies necessitates a seamless integration of those into the whole networking landscape (i.e., we need migration paths), so that appropriate connectivity services can be provided regardless of the evolving access technologies.

Then, when deciding on an interface and a network to be used, the end-users do have clear options of their preferred access technology and network operator; yet, currently there is a limited involvement of the end-user in the decision process about the network to connect to. Thus, OConS should allow different strategies for network selection and handover steering, in which the decision process might be shared by several actors, such as end-users, operators, and service providers. This is likely to become even more relevant for the multi-path solutions, where different applicative services might use different paths (and probably also different accesses) depending on the particular characteristics, user policies, network conditions, etc. Correspondingly, greater cooperation between several entities should be strengthened even between different technologies, domains or operators. The relevant standardisation activities have already identified the need for such cooperation and have specified solutions which partially cover the requirements (e.g. see IEEE 802.21), but those are limited to particular use-cases (i.e. handover between heterogeneous networks in this mentioned example).

Furthermore, more and more wireless techniques continue to flourish, along with their corresponding benefits and challenges. E.g., the Wireless Mesh Networking comes with a multitude of wireless connected devices and with the establishment of multi-hop (and sometimes spontaneous) topologies which can be used to extend the connectivity beyond the legacy single-hop access networks; nonetheless, resource management within such topologies brings much more difficulties, such as requiring on-the-fly cooperation strategies between different nodes to establish relays and to select gateways.

Finally, one could argue that the mentioned requirements and functionalities can be achieved with currently available networking technologies; however, they lack flexibility and commonly require complex configurations and continuous oversight.

3.2.2 Proposing new design guidelines to be followed by OConS

After recalling the principles of current architectures and communication technologies, and explaining the reasons why they are not always sufficient to cope with the above stated requirements, we now propose a set of design guidelines which should be taken into consideration by the OConS architecture.

One of the cornerstones of the OConS framework should be its technology independence, i.e. to minimise the impacts induced by technology constraints. This spans over both the access part (wireless and fixed) as well as the core network (e.g., switching, routing, interconnection between data-centres, and so on). The multi-P paradigm has been coined within OConS so as to reflect this intrinsic characteristic.

The management of the connectivity services should be autonomous, able to dynamically adapt to various conditions as well as to balance between various decisions points, thus deviating from the more traditional centralised approach. This automaticity requires, among other things, procedures to discover and negotiate the corresponding services and functionalities.

As has been briefly mentioned above, the architecture should be able to adapt to the rapid evolution of the communication technologies. This implies that components of the architecture need to offer common services and functionalities, which can be used independently of the particularities of the subjacent technologies or the application/services using OConS. A straightforward consequence is thus the choice of a modular architecture, designed and built following an object-oriented approach, which can instantiate the various entities according to the particular needs and which can therefore be re-used in difference contexts. By implementing well-defined interfaces, this modular design also allows the independent modification and enhancement of each module, while hiding the complexity of the embedded mechanisms (and their evolution) to the users.

Besides, an appropriate (including tight) interoperation between layers is also foreseen as a key aspect of OConS, so as to dynamically couple the corresponding connectivity services across several layers (e.g. cross-layer mobility management, cross-layer GMPLS instance, etc.). This cross-pollination would also bring context awareness into the OConS, which thus will be able to tailor its services to energy, cost, QoS, cloud constraints, etc.

Finally, and whenever possible, our framework should facilitate the recursive/reflexive use of the different methods, mechanisms and services proposed (i.e., the polymorphism, where an OConS mechanism can call itself with the same or with a different input/policy).

3.2.2.1 Design Guidelines on Openness

The “Openness” motto has various implications and consequences to the design of the OConS architecture. It affects all types of connectivity services, and as such (using an illustrative example) we should go beyond the current OpenFlow and not limiting ourselves to the policing/steering of the forwarding mechanisms for a given flow.

Furthermore, it also implies accessibility to the available connectivity services to any authorised user, breaking some of the existing frontiers between different domains, as opposed to a closed system in which a protective strategy would be to hide the knowledge about them. This has a clear impact on security (e.g., privacy and access control), and in OConS we investigate the means to appropriately deal with it.

On the other hand, this openness also leads to the definition of publicly available interfaces, with standardised functions (primitives), behaviour (sequence of primitives) and formats (encoding of information elements). The need to offer flexible and extensible functionality, that is able to adapt to upcoming connectivity services, has also to be addressed, e.g., with a clearly defined migration strategies.

3.2.2.2 Design guidelines on Routing

One principle which could be adopted here, is to split data forwarding (which usually happens on a distributed way) from routing control and policy (mostly a centralised process), with two main facets: (1) both mere data forwarding and routing protocols should be executed in a possible distributed manner; (2) there should be a clean split between the routing decisions to a (set of) destination(s) when multiple paths are available without involving policies, and the policies themselves.

Besides, since global routeability might not be available for all services and applications, the OConS should consider the limitations imposed by the selected addressing and naming schemes, such as the size of routing tables, the scalability of Border Gateway Protocol (BGP) routing mechanisms, or number of VPNs. Therefore, the definition of domains might become necessary, as well as the specification of the approaches to deal with the inter-domain cases.

3.2.2.3 Design guidelines on Transport

As opposed to most of the current communication models, OConS deals not only with the traditional end-to-end paradigms, but also with hop-to-hop (like in DTNs) or edge-to-edge (e.g., VPLS/OTV/TRILL) approaches.

OConS will ensure a set of minimum requirements, like the support of multiple points of attachment, broadcast and multicast communications. Besides, OConS should facilitate the establishment of connectivity to a set of destination and potential sources (for instance, when gathering content for various caches).

Likewise, multiple types of congestion control (e.g. window based, rate based, delay based) may be supported for an application depending on specific flow requirements, as well as different options for reliability on specific paths and/or a specific reordering level.

For the establishment and the management of the connectivity, OConS should not be limited to the control/management of single packets, but also to their different logical aggregation levels, such as: flows, sessions, bearers, paths, etc.

We are not targeting thus a connection-oriented approach; instead, we are advocating a connection-emulated approach, enriched with several connectivity services (such as Multi-Path/Multi-Point, comprehensive resource/QoS management, dynamic/distributed Mobility support, or autonomous data-centre-interconnect), while still making use of the advantages of the packetised networking (e.g., IP, MPLS, and Ethernet).

3.2.2.4 Design guidelines on Security

The OConS framework will follow the overall SAIL security principles, aiming at authentication and authorisation as well as confidentiality, integrity and availability. The implementation of security services shall use suitable cryptography technologies following a security by design, as opposed to security by obscurity. Besides, the goal for selecting implementation technology shall be to first use existing, well-proven standards, and only develop new solutions if this cannot be avoided.

An overall SAIL security framework is developed in the Security Theme. This effort covers apart of OConS also NetInf and CloNe results and is developed as a means to identify suitable security services and to streamline their design and implementation. Likewise, suitable security management procedures shall be developed to maintain the original levels.

Of the traditional requirements of any security solution, authorisation/authentication and privacy are of outer relevance for OConS. Regarding the first two, it is worth saying that connectivity services should be only provided when all involved entities (previously authenticated) have agreed to do so.

On the other hand, privacy (tightly linked with security) goes beyond traditional requirements, to ensure not only protection of users' data, but to enable user control of the level of this protection. Besides, we consider the broader case, which targets protection of data belonging to operators, service providers or any entity related to either the use or the provision of OConS services. Hence, the exposed information shall be adapted and filtered to other entities depending on the particular policies, but still assuring the correctness of that information. There will be neither one nor a unique solution, e.g. if these entities reside in different legal frameworks and privacy thus becomes another flavour, or if the shared information gets correlated with other data and thus is breaching privacy concerns. In this sense, the privacy related work is intended to highlight potential issues relating to privacy loss within the OConS architecture, rather than propose specific solutions to ensure it.

3.2.2.5 Design guidelines for Mobility

The ultimate goal here is to ensure transparent and seamless mobility to the user of OConS. This may imply the possibility to instantiate on-the-fly various mobility solutions only when needed; one illustrative example of this would be the possibility of establishing on-demand tunnels instead of re-routing. On the other hand, mobility decision entities should be dynamically distributed or chosen, as opposed to the centralised approaches.

Mobility support might be confined to a given domain or considered at a global scope, thus leading to various types of resolution/mapping mechanisms (e.g., global, localised, “service-specialised”, and so on). On the other hand, mobility services do not have to be restricted to end-terminals, but they could be extended to content-IDs (e.g., NetInf) or processes/virtual-hosts (e.g., CloNe).

3.2.2.6 Design guidelines for Resource Management

As we have discussed before, the increasing complexity and variety of resources brings about the necessity of providing the means of managing them in a proper way. Most of the currently available procedures are based on centralised approaches, and they do not benefit from the cooperation with other resource managers. In OConS we assume that resource management mechanisms should be autonomous (that is, able to operate of a self-* way), while supporting a distributed operation, and being able to share the decision processes with other peer-entities. OConS also facilitates the interoperation between different entities belonging to different administrative domains (e.g. operators).

In order to achieve such cooperation, a modular approach is desirable, so that mechanisms can be combined on-the-fly, as they are needed for any particular situation. In addition, a comprehensive framework is also deemed necessary, since the management of a particular set of resources may have some consequences over others.

Last, but not least, it is worth highlighting that in the OConS framework we assume that the networking/communication resources can be virtualised and thus, we consider them as yet another type of item to be managed.

3.2.2.7 Design guidelines for Migration

Migration in this context deals with phased introduction and inter-operation of subsystems of generation N+1 (N+1.G) with functionally comparable subsystems of generation N (N.G). Thus, a migration path is the process describing how to update a system of generation N to a complete system of generation N+1, without compromising its legacy functionality. To support migration in OConS, the following assumptions will be made:

- N+1.G subsystem shall implement, at least, the functionalities of a N.G subsystem, as far as they are observable from outside (i.e., the backwards compatibility)
- N+1.G subsystem shall be able to detect whether a subsystem operates on N.G or N+1.G
- N+1.G subsystem shall inter-operate with a N.G subsystem via N.G-compliant interfaces
- N+1.G subsystem shall behave as a N.G subsystem towards other N.G subsystems
- N+1.G subsystem shall exploit the new functionality when inter-operating with another N+1.G subsystem.

In order to allow N+1.G subsystems to communicate with other N+1.G subsystems via intermediate N.G subsystems, the N.G subsystems need to implement some extension mechanisms, so as to tolerate N+1.G features on the N.G subsystem. This may imply some security concerns which are left for further discussion.

4 Architectural Framework

As seen from the requirements presented in Section 3.1, the OConS architectural framework needs to cope with various challenges. By following the architectural guidelines stated in Section 3.2 (i.e., openness, modularisation, on-the-fly connectivity services, "path" concept for emulating the connections, self- and distributed control, seamless mobility), we have first defined the functional entities which are the elementary building blocks for OConS. We have then continued by presenting the internal interfaces between these different functional entities and the external API towards other networking functionalities (e.g., NetInf [SAIL-D.B.1] and CloNe [SAIL-D.D.1]), and we have finally discussed the various types of information needed by the OConS mechanisms.

4.1 Architecture Overview

The great variety of mechanisms, procedures, and protocols which are deemed necessary to cope with the discussed requirements, together with the openness which we aim for our OConS architecture, lead us to the definition of a flexible framework. From a bird's eye, most of the actions which are envisaged within the scope of the OConS could be characterised in three basic phases: (1) collecting the needed information; (2) taking the suitable decisions on the basis of such information; (3) and enforcing the decisions by instantiating the appropriate mechanisms. Likewise, these three phases can be invoked iteratively or following different patterns, e.g., depending on the collected information and the decision processes outcome.

This stepwise approach of dealing with the connectivity procedures is depicted in Figure 4.1 in which we also highlight the main scope of the two tracks which streamline the design and development of the OConS vision: collecting the information and taking decisions accordingly pertain to the management of connectivity functionalities, while the execution and enforcement can be considered as providing the advanced connectivity services.

As will be seen later, the OConS architecture has been conceived so as to mimic these three phases, with the main goal of having the degree of flexibility required to integrate mechanisms and procedures which have at the first view quite many differences among them.

After defining the barebones of the OConS architectural framework and the interfaces between its various functional entities, we will also exemplify how these can be deployed within real networking elements (e.g., end-terminals, access elements, core networks, etc.), the way they interoperate between each other following different control models, as well as the services that can be provided to the external parties (most notably towards the NetInf and the CloNe frameworks).

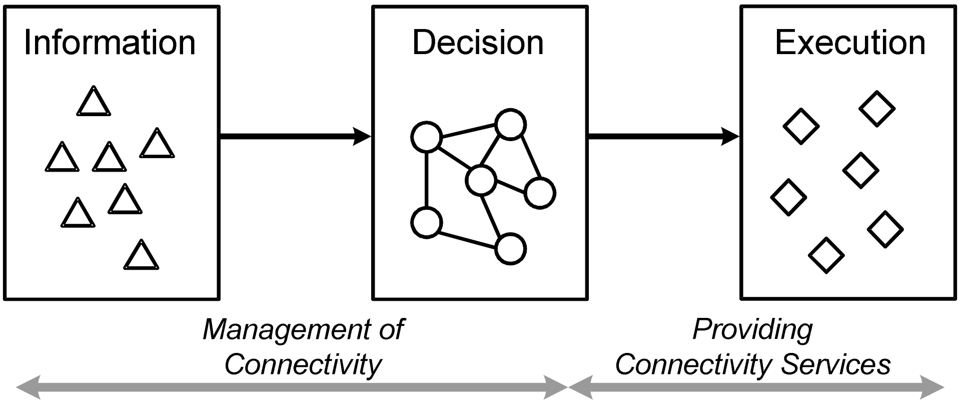


Figure 4.1: OConS three main phases for handling the connectivity

4.2 Generic Functional Entities and Communication among Entities

After carefully taking into consideration the requirements and following our design guidelines, we propose a component-based OConS architecture with three different functional entities, independent of and abstracted from any layer or protocol, as follows:

- **Information Management Entity (IE):** They collect useful information, e.g., related to QoS, signal strengths, traffic, etc., and provide it to the Decision Making entities. The information gathered can be processed (e.g., abstracted, aggregated, filtered, and so on) by an IE before being transmitted to the DE requesting it or being subscribed to it. The IEs can be hosted on different devices in the network such as routers, access points, base-stations or on end-user terminals; they can also be hosted on a dedicated device, i.e. a specific monitoring device.
- **Decision Making Entity (DE):** A DE uses the information from the IEs to make a decision, e.g., to initiate a handover, to change a service treatment, to grant or deny user access to a service, and so on. Likewise, a decision can be taken in one centralised location within the network (which is typically the case in 3GPP systems), but it can also be made by a distributed decision mechanism. The information taken into account for making the decisions are, e.g.: the measured QoS in the networks and terminals, the QoE information collected from users and applications, but it can also include various policies and preferences (e.g., on SLAs, security/trust, costs, etc.).
- **Execution and Enforcement Entity (EE):** Once the decisions made, they need to be executed and enforced, so the OConS also provides this functionality. Moreover, the execution and enforcement cannot generally be performed by the network element which made that decision; e.g., for granting access and performing handovers several elements have to be involved in the execution phase, such as access nodes, routers, databases, servers, end-user terminals or servers.

Likewise, the assignment of these functional entities to concrete devices or nodes can be dynamically done, depending on the communication context, i.e., a device can host one, two or all of these functional entities. Accordingly, the possible node instantiations for these entities are: a) IE & DE & EE; b) IE & DE; c) IE & EE; d) DE & EE; e) IE alone; f) DE alone; g) EE alone. Therefore, we smooth the possible grouping (i.e., layering) of these entities into major communication functions known from the OSI model, such as: link-transmission, routing, or end-to-end transport, but also into other novel models that fit better with a given networking context.

As two illustrative examples we may have: 1) a Router that measures the traffic, it decides on the route and changes its routing table accordingly; 2) an Access Point that measures the radio signal strengths and it sends them to a controlling node in the Core Network where a decision is taken, then this control node sends a handover command to be executed at the mobile terminal for the uplink and to the Mobile IP Home Agent for the downlink.

Limiting the framework to only three functional entities, but allowing each of them to be placed onto one or distributed over several entities, enable us to support different configurations, topologies and scenarios. Because some of the functionalities can be realised on different layers (i.e., layer-independence), the OConS approach facilitates the endorsement of new layering models, as well as the support of legacy approaches.

4.2.1 Inter-Entities Communication Concepts

The OConS guideline on Openness requires that the interfaces between the components of the architectural framework shall be “open interfaces”, i.e., they need to be specified in terms of messages, behaviour, formatting/encoding and functions.

Thus, for ensuring the generic communication among entities, the following control interfaces (or reference-points) were identified:

- (O_{DD}) : DE <-> DE
- (O_{DI}) : DE <-> IE
- (O_{DE}) : DE <-> EE
- (O_{II}) : IE <-> IE
- (O_{EE}) : EE <-> EE
- (O_{EXT}) : OConS <-> External

The only missing case is the direct IE-EE communication, which has not been foreseen within our architectural framework. Furthermore, as working assumptions, we have considered that the interfaces are bi-directional and that the requesting direction starts in general from a DE.

In addition, to allow for further functional split models like traditional layering or other novel approaches, we have proposed the external interface (O_{EXT}) thus providing services to external "neighbouring" functional entities (e.g., the "upper layers" such as NetInf or CloNe); therefore, the (O_{EXT}) serves the role of an OConS "Application" Programming Interface (i.e., OConS-API).

Figure 4.2 depicts the generic functional entities instantiated (as an example) on four network nodes together with the identified interfaces.

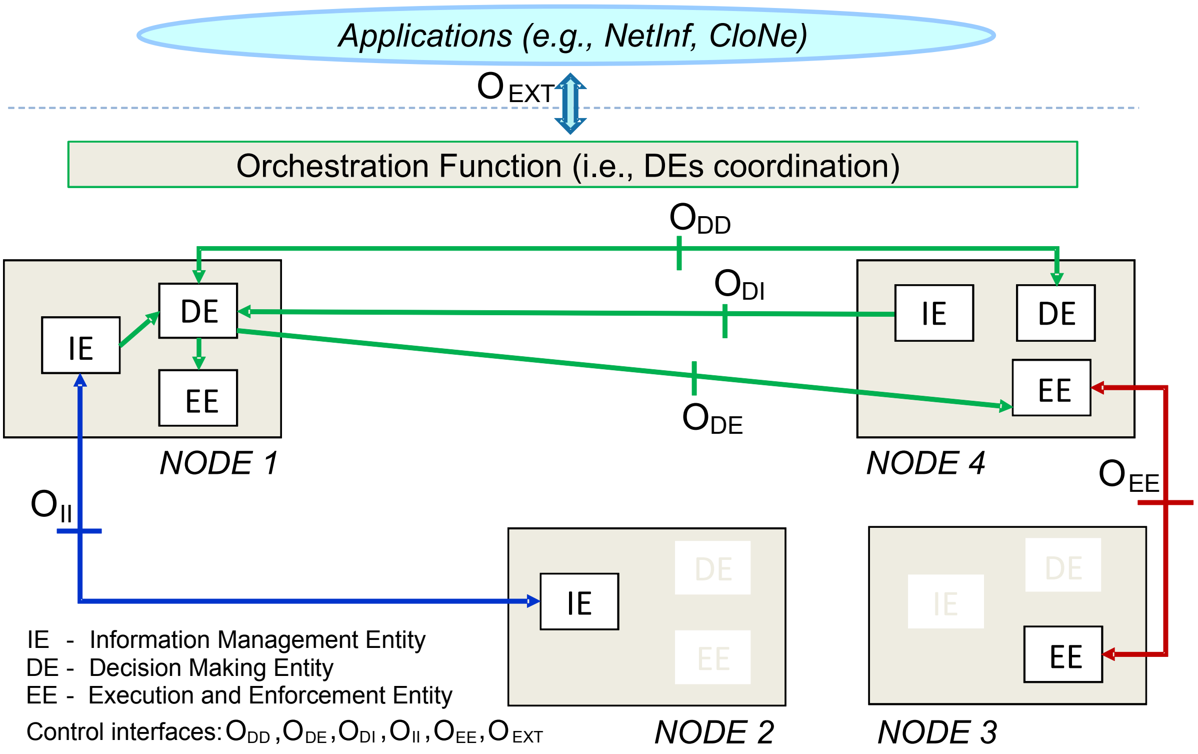

Applications (e.g., NetInf, CloNe)
O EXT
Orchestration Function (i.e., DEs coordination)
ODD
ODI
ODE
OEE
OII
IE
DE
EE
NODE 1
NODE 4
NODE 2
NODE 3
IE - Information Management Entity
DE - Decision Making Entity
EE - Execution and Enforcement Entity
Control interfaces: ODD ,ODE ,ODI ,OII ,OEE ,O EXT

Figure 4.2: OConS functional entities and their interfaces

Accordingly, one of main features of the OConS architectural framework is the capability to distribute the functional entities across several nodes using its "open interfaces" as presented in Figure 4.2. We have also proposed an Orchestration function to coordinate the several DEs that might be involved when dealing with an OConS service request for a given "application"; for the proper design and specification of this functionality, we also intend to capitalise on the feedback provided by the prototyping and experimentation activities later on; therefore, the full definition for this Orchestration function will be provided in a subsequent deliverable.

Moreover, the distribution of these functional entities poses specific requirements on the communication between them and, thus, different approaches can coexist:

1) ***Intra-node communication***: due to several factors, and whenever possible, it is beneficial to confine the intelligence and the corresponding software complexity inside the DE and

keep the rest of the entities as simple as possible; the additional benefit of this approach is that there is no need to define several different interfaces; the other reason for this strategy is to avoid temporal, as well as semantic, inconsistencies resulting from either diverse simultaneous or non-consistent inputs received from different entities; thus, in this case, we have only defined two bidirectional internal interfaces (O_{DI}) and (O_{DE}) (not marked textually in Figure 4.2. for simplicity).

2) ***Inter-node communication***: as the nodes/devices containing the OConS entities are usually in different environments (i.e., technologies, layers, or geographically), the information exchanged between them must be coordinated, e.g., using a unified Inter-Node Communication (INC) function. Some functionalities of the INC comprise the compatibility identification of the communicating nodes (i.e., version matching), the adaptation of the inter-node messages if necessary (i.e., translation), and possible the generic security/AAA functions. For the placement of the INC within a node, we have proposed three different cases:

   a) Dedicated INC entity: a separate INC entity carries out all external communications of all the entities inside that node/device; the INC in case must have also the capability to identify the destination of the incoming messages to direct them to appropriate entity.

   b) INC integrated in DE: the external node communication is only allowed from/to the DE; in this case the INC function has to proxy the messages through the (O_{DD}) interface.

   c) INC integrated in each Entity: in this case all OConS entities within a node/device have the INC functionality to communicate with a remote entity in another node/device.

4.2.2 OConS framework flexibility exemplified with generic mechanisms

We present here a couple of examples of generic mechanisms showing the flexibility of our framework. First, we depict in Figure 4.3 a decentralised/peer-to-peer control model where each node can make certain decisions (e.g., about routing, mobility, resource allocation) autonomously based on local or neighbouring information and by exchanging its decisions with the adjacent nodes, and where each node locally executes the decisions once they are made.

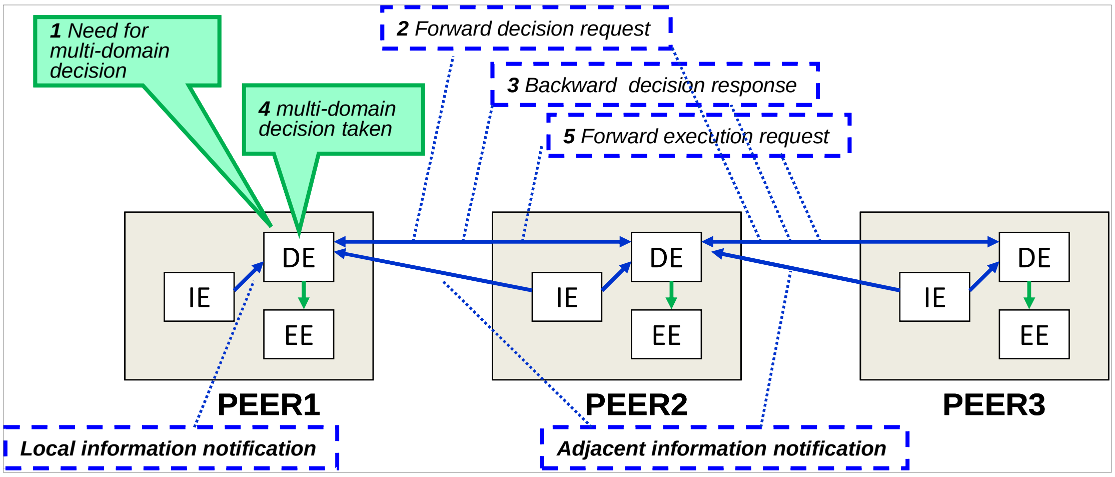

1 Need for multi-domain decision
2 Forward decision request
3 Backward decision response
4 multi-domain decision taken
5 Forward execution request
IE
DE
EE
IE
DE
EE
IE
DE
EE
PEER1
PEER2
PEER3
Local information notification
Adjacent information notification

Figure 4.3: OConS control model: Decentralised/peer-to-peer interoperation

Then, in Figure 4.4 we show an example for the multi-domain control, where a hierarchically-centralised approach is used within these domains based on a parent node (i.e., a master controller) which coordinates the decisions for several child nodes and which needs first to concentrate all the necessary information before making the appropriate decisions and then sending the execution orders towards all its child nodes.

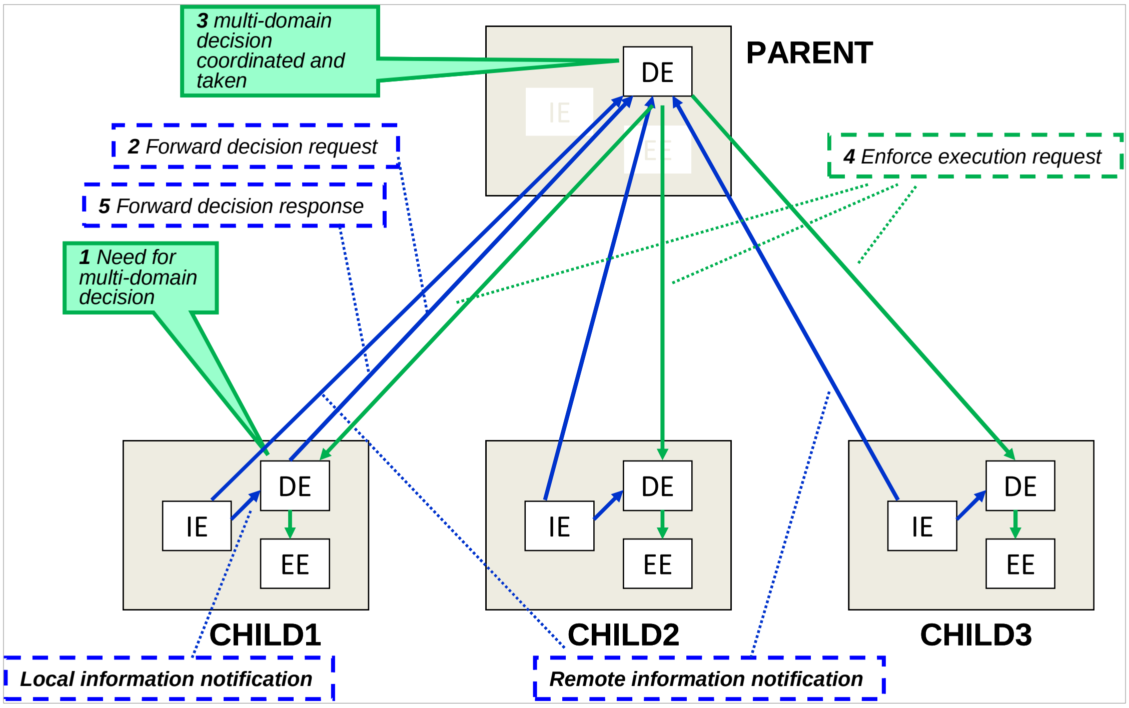

3 multi-domain decision coordinated and taken
PARENT
DE
IE
EE
2 Forward decision request
5 Forward decision response
4 Enforce execution request
1 Need for multi-domain decision
IE
DE
EE
CHILD1
IE
DE
EE
CHILD2
IE
DE
EE
CHILD3
Local information notification
Remote information notification

Figure 4.4: OConS control model: Hierarchically-centralised interoperation

4.3 Discussion on the Interfaces and the API

Regarding the interfaces' specification between the OConS functional entities, we have identified the need to separate the control from the data transfer; most of the OConS procedures apply to management and control operations, and, thus, the number of interfaces to be specified therein is rather large; likewise, some of the foreseen messages (e.g., dealing with request/response exchanges, discovery procedures, etc.) are common to all the interfaces, while others are envisaged for a particular procedure. On the other hand, when it comes to the data transfer, we have restricted it to the transfer of data between EEs.

We also provide some initial set of messages to be exchanged with external entities/users (through the OConS-API) willing to make use of the services provided by OConS; special attention is to be paid for the interoperation with NetInf and CloNe.

We have also considered the use of RESTful [REST] interfaces towards the users of OConS. The OConS interfaces operate on individual resources, quite similar with the CRUD approach, i.e., Create (POST), Retrieve (GET), Update (GET and PUT), and Delete (DELETE); for example, in OConS we may need to create path, create link, create port, retrieve path, update path, delete path, and so on. In such a slim RESTful API, every resource is thus uniquely identified (e.g., by its URI) and the resources are managed through these CRUD operations.

4.3.1 Internal Interfaces for Control/Management

In order to better depict the interfaces between the different OConS entities, we make the two following assumptions: (1) all OConS entities (IEs, DEs and EEs) have names (to be defined at a later stage) which can be resolved into the corresponding addresses/locators, depending on the underlying technology (for instance, DNS for IP domains, NRS for NetInf, etc.), and (2) OConS relies on the available L1/2/3 technologies so as to perform the required bootstrapping mechanisms.

The following messages are common to all the interfaces between OConS entities:

- ***Discover OConS Entities request***: it is part of the bootstrapping mechanisms and can be used to find peer or other OConS entities.
- ***Discover OConS Entities response***: this message will be used so as to complete the discovery mechanism upon the reception of the corresponding request.
- ***OConS Capability request***: this message will be used so as to find the capabilities implemented by any OConS entity.
- ***OConS Capability response***: this message is sent upon the reception of the corresponding request and indicates the capabilities implemented by the OConS entity, as well as the status of the corresponding action.
- ***OConS Capability notification***: in this case we assume that an OConS entity might want to, proactively, inform about its capability (without previously receiving a request).

The interface between DE and IE (O_{DI}) comprises the following messages:

- ***Configure IE request***: this message is used so as to configure the operation of the IE and to subscribe to various pieces of information. It includes the parameters to be notified about, thresholds, model to be used to retrieve the information, etc. This message is also used to configure an IE to send certain information to other IE(s).
- ***Configure IE response***: when an IE receives a request from a DE, it replies back with a response, in which it indicates the status of such configuration.
- ***Information request***: although the IE can provide the information proactively, there might be cases in which a DE needs to retrieve a certain piece of information immediately, thus we need this message for such purposes.
- ***Information notification***: the IE uses this message to send the corresponding information to the DE, either after the reception of an “Information request” or according to the specified configuration (e.g., periodical, threshold crossing, etc.).
- ***Notification response***: this message is sent by the DE as an acknowledgment of a received “Notification”.

The interface between peer DEs (O_{DD}) comprises the following message:

- ***Decisions Exchange***: for time being we only envisage one message between peer DEs, used to interchange either rules or partial decisions between them (the latter will be used so as to implement distributed decision procedures). This message might be unicast, anycast, multicast or broadcast, depending on the particular situation.

The interface between DE and EE (O_{DE}) comprises the following messages:

- ***Execution request***: this message is sent from a DE to an EE so as to enforce a previously taken decision, providing (if needed) the parameters with the corresponding configuration (e.g., the name of a remote EE).
- ***Execution response***: the EE sends this message to inform the DE of the status of the previously requested execution.

The interface between peer IEs (O_{II}) comprises the following message:

- **Information Exchange**: in some cases, the information which is to be delivered to the DE might come from various sources, which shall be coordinated between them; this message is intended to enable such cooperation between the IEs.

The interface between peer EEs (O_{EE}) comprises the following message:

- ***Execution Exchange***: once the DE has enforced a particular decision, there might be particular issues and actions which lie under the responsibility of the particular EE.

4.3.2 Internal Interface for Data Transmission

In our framework, we assume that the only interface in which we have data transfer (i.e., from the upper services/applications) is between the EEs; we call it data transfer interface (O_{DATA}) (not shown in Figure 4.2 for simplicity) with the following two messages:

- ***Send/Receive*** (actual data): it is used to send and receive actual data between EEs.

4.3.3 External Interface for Control/Management (OConS-control API)

This interface will provide the API between OConS and its users, i.e., (O_{EXT}) from Figure 4.2. The definition of the definitive messages will also depend on the particular needs and requirements of some of these users (most notably NetInf and CloNe).

In the following we describe a set of some messages which are likely to be needed.

- ***Register to OConS domain***: it is used by an entity (e.g., node, end-point, application, process, etc.) to register to the OConS current domain. It is worth saying that despite being part of an external interface, it can also be called recursively, i.e. by another OConS entity. On the other hand, this registration might embed some security (e.g., authentication) mechanisms.

- ***Discover OConS capabilities***: this message can be used by the calling entity so as to gather the functionalities which are offered by the corresponding OConS domain.

- ***Remove from OConS domain***: it is used to delete a previously registered entity from the OConS domain. Security concerns should also be considered herewith (e.g., if an entity might be removed or not by other entities).

- ***Update OConS path***: the basic functionality of OConS is to provide connectivity services to its users. The main abstraction for offering such services is the bearer (or the path). Thus, this message can be used to adapt the operation of a bearer/path while in use, or to create a new one if it did not exist. An entity should be able to update/create a bearer/path only if it was previously registered. Furthermore, this message can be sent recursively, i.e. by another OConS entity.

- ***Delete OConS path***: it is used when the previously employed connectivity services are not longer needed.

4.3.4 External Interface for Data Transmission (OConS-data API)

A large number of current services and applications use the Berkeley socket interface to transmit and receive data. Although OConS (through a migration strategy) would coexist with this legacy approach, an OConS new socket type is to be developed, so as to allow the OConS aware services and applications to send and receive data their data.

- ***Send/Receive data***: it shall include the ID of the bearer/path to be used to transfer the actual data.

Note that even if a bearer/path ID is required to send/receive data, we are not targeting a connection-oriented approach; we are advocating instead a connection-emulated approach.

4.4 Description of the information needed

The proposed OConS architecture provides the means to activate the appropriate mechanisms/ protocols and, most important, react to changes in the network conditions. Within this proposed OConS architectural framework, the decision mechanisms are based on information collected by the IEs. The decisions are then propagated from the DEs to the EEs through the OConS interfaces.

We present here the structure of the different types of information. We will formalise in a subsequent deliverable the detailed description of this information (e.g., in an XML Schema), which will be mostly used for the prototyping and experimentation activities.

Therefore, the information needed by decision processes is structured as follows:

- ***Resources***: split into two types, network resources and end-terminal/devices resources. They are described through their characteristics and capabilities, some varying over time. Network resources are essentially nodes and links, being specified by attributes that are specific from their technology and composition. Examples of such attributes are load, bandwidth, price, mobility events history, operator, neighbours Action Points (APs) or Base Stations (BSs), etc. Identified terminals attributes are, e.g., available interfaces, screen, battery, mobility events history, measurements, neighbours terminals/APs/BSs, and so on.
- ***Context***: it refers to relevant situation constraints, e.g., location/time, scenario characteristics, mobility patterns, user behaviours, type of payload/service. The decision processes can be context-aware to adapt optimally the connectivity service.
- ***Requirements***: can be from an application or a user perspective, depending on the source of the request for connectivity service. Some examples of required information from the application viewpoint are: application identification, minimum QoS, SDP, minimum signal level, minimum throughput required. From the user point of view, the requirements are, e.g., maximum price, minimum QoE expected.
- ***Policies and preferences***: are identified from user, operator and service perspectives. The user's preferences can be, e.g., preferred interface for an application, low cost operator or minimum power consumption. For the operator's perspective, examples of policies and preferences are the preferred access for a user and an application, the minimisation of operation cost or energy consumption. Finally, service's policies and preferences can be the preferred interface, or the list of preferred radio access technologies.

In Chapter 5, 6 and 7, several decision mechanisms are presented and they are mapped onto the OConS architectural framework. For each mechanism, the needed information collected by the IEs is clearly identified and hints on the possible protocols to be re-used/enriched have been provided.

Nevertheless, the complete definition of these mechanisms and the full specification of their interfaces are currently under development; thus, the homogeneous adoption of the proposed interfaces by all our connectivity services will be provided in the subsequent deliverable.

5 Advanced Connectivity Services

The continuous evolution of services and applications, and the emergence of novel paradigms related to content-centric networks and cloud networking, necessitate both performance improvements and increased flexibility of networking mechanisms, protocols and algorithms.

In order to achieve these goals, new and advanced connectivity services are required. In this chapter we introduce different connectivity services which are integrated within the OConS framework, the techniques to achieve them and the means to improve their performance. By supporting these advanced services, the networks can more easily evolve than nowadays; moreover, the migration strategies that come with them provide the required backwards compatibility. Such new connectivity services will be mainly based on the following novel Multi-P mechanisms:

(a) Multi-Path: the mechanisms that allow the same flow to use multiple simultaneous paths in a fair and efficient way.

(b) Multi-home: the mechanisms to support multi-home nodes; besides commercial incentives for multi-homing, effective handovers are required to deliver a given flow to multiple points.

(c) Multi-Protocol: the mechanisms to be executed on the same flow, dynamically selecting different transport protocols and configuring the parameters for a given communication task.

(d) Mechanisms that assist information-centric networks to benefit from the established multiple paths at the transport and network layers.

Our work also includes the integration of Network Coding (NC) and Cross Layer techniques to improve the performance of Multi-P mechanisms at the transport and network layers. Furthermore we have looked at network control functions and the End-to-End support for WAN interconnectivity as well as control functions for the Multi-P mechanisms introduced above.

At this stage of our work, we are designing the major Multi-P functionalities; the API to expose such functionality is left for future work.

Likewise, smooth migration from existing solutions is taken as a major objective and thus considered in the OConS design. The extensions required for each protocol together with the migration path will be elaborated in future deliverables.

5.1 Enhanced Encoding for Multi-P

5.1.1 Network Coding for M-to-N Routing in DTNs

It is well-known that in challenged networks undergoing scarce end-to-end connectivity (e.g., Delay Tolerant Networks), naive broadcasting mechanisms such as flooding can be complexity-efficient, yet bandwidth-inefficient data dissemination procedures, mainly due to the high number of unnecessary retransmissions and the subsequent wasted bandwidth at intermediate relay nodes. In this line, epidemic and/or probabilistic routing protocols such as PRoPHET account for the history of encounters and transitivity between constituent nodes in a challenged network environment.

The main goal is to show how Network Coding (NC), through its underlying mechanisms (i.e. packet selection, combination with adaptive probabilistic forwarding approaches) can benefit from network-wide social metrics. In this context, we will implement and validate specially-tailored NC approaches for data dissemination (M-to-N) over real DTN connectivity traces registered in a field trial at our facilities. The availability of certain information such as pair wise inter and intra-contact time statistics facilitated by the OConS architecture favour an optimised selection of packets which should be network-encoded at a given intermediate node, with an emphasis on challenged networks with extremely short transmission windows.

Advanced tools for synthesising and analysing such statistics have been applied and cross-checked with the behaviour of overlay NC when applied to a certain scenario. We have focused our work on addressing NC benefit from social graphs. First stage of this analysis consisted on establishing a balance between the priority of connectivity statistics and the degree of innovation one node encounters (i.e. the number of innovative packets spread by every node during its encounters). As a consequence we are in the phase of simulating environments where nodes encode packets according to our established balance of priorities in a distributed fashion and using incrementally computed metrics. We will derive an adaptive probabilistic version of NC where the decision engine will not only affect the selection of the packets to be network-coded, but will also establish the probability of transmitting to a certain node (technique thus coined as joint adaptive network-coding and probabilistic forwarding).

In the context of challenged networks, further efforts must be particularly conducted towards a more realistic modelling of the dynamicity and mobility of constituent nodes. A generic MATLAB framework is being built for computing discrete-time-slotted performance metrics when applying naive and socially-inspired NC to DTNs. This simulation framework will permit to evaluate the performance of the designed NC approaches over both synthetic and practical scenarios based on long-term registered connectivity traces at the TECNALIA premises.

Figure 5.1 presents the mapping of the proposed selection and encoding (NC) process onto the OConS architecture. Our decision engine relies on the information exchanged among DTN Nodes which estimate inter-contact and contact duration times from historical encounters (previous behaviour) with their neighbours (see Section 7.2.2 for a more detailed description). This data generate network-wide social information like the social innovation rate of nodes (i.e. the ratio of new pieces of information – new packets - each node creates and sends) that is actually very useful for the encoding decision. The DE is the entity in charge of making a decision on which packets should be jointly network-coded, as well as to which neighbouring node the encoded packet should be forwarded to. This twofold decision is tightly coupled to both the social learning process (see Section 7.2.2) and specific data of the node (e.g. buffer status, number of stored innovative packets).

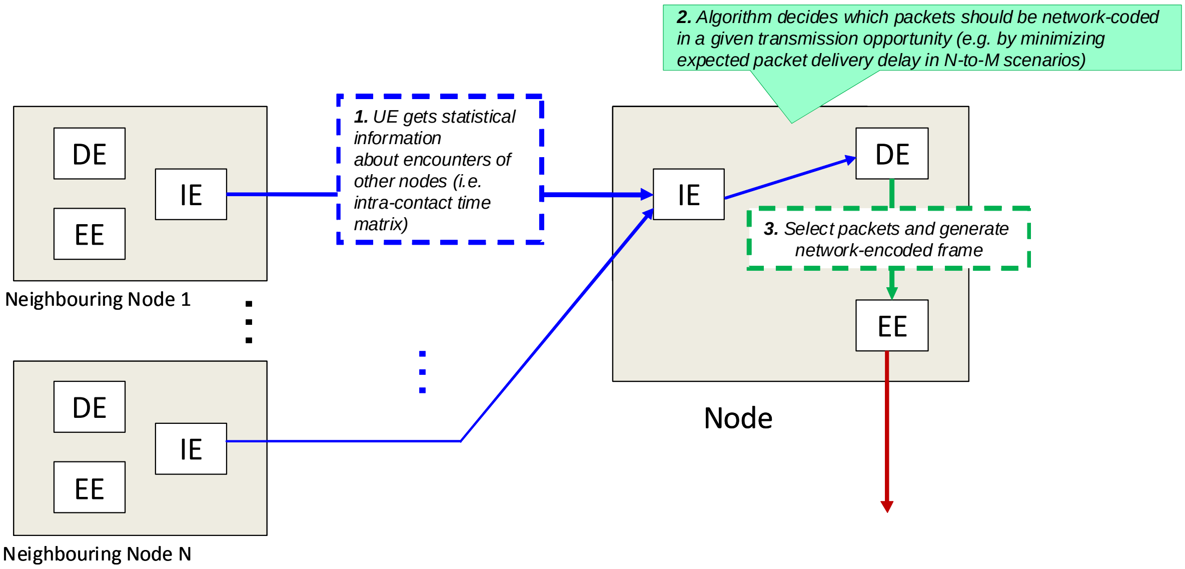

2. Algorithm decides which packets should be network-coded in a given transmission opportunity (e.g. by minimizing expected packet delivery delay in N-to-M scenarios)
DE
IE
EE
Neighbouring Node 1
1. UE gets statistical information about encounters of other nodes (i.e. intra-contact time matrix)
IE
DE
3. Select packets and generate network-encoded frame
EE
Node
DE
IE
EE
Neighbouring Node N

Figure 5.1: Selection and Encoding (NC) process mapping onto the OConS architecture

As a result, the DE produces a list of packet indexes to be coded into a single frame, as well as the probability of sending this encoded frame to a certain number (balance between effectiveness and overhead) of neighbours.

Another line of research is to investigate tools especially suitable for synthesising the connectivity information in a graphical format for its exploitation at the NC processing layer.

This is focused on implementation, although conceptually the benefits of social metrics on NC can be guessed from previous developments, graphical tools such as the so-called social graphs permit to embed mobility and node encounters into a single parametrically-defined graph structure, which can be useful for formatting the social information generated at every node into a unified and visually-analysable layout.

The information needed by the decision process is summarised in Table 5-1.

Table 5-1: Information needed for NC for M-to-N Routing in DTNs

Resources	Node resources	• Buffer status • Size of NC field affordable by the node logic (usually 8 bits, but can be set arbitrarily). • Reserved memory for computing the degree of innovation gained at every reception event.
	Radio channel resources	• Available data-rates.
	Network	• Number of neighbours (both instantaneous and historical). • Link loss probability • Origin node and time-stamping of a given received innovative packet.
	Context	• Popularity index (it may be useful in certain types of communications, i.e. if NC is considered) • Inter-contact time and contact duration estimations (to characterise interactions among nodes)
Policy	Strategy goals	• In case of severely faulty links, minimise the number of retransmissions. • Minimise energy consumption. • Maximise throughput. • Speed up and increase the packet delivery rate and ratio, respectively. • Minimise the number of redundant transmission events for a 100% delivery ratio.

5.1.2 Network Coding and Transport (TCP) over wireless

In theory, the use of NC in Wireless Mesh Networks (WMNs) should yield significant gains, in terms of increased throughput. However, in practice the characteristics of TCP---in particular, TCP's congestion control mechanisms---may lead to lower-than-expected gains. Indeed, several issues may arise when the traffic carried by a WMN using NC is composed mostly of TCP flows.

5.1.2.1 Motivation

One issue is related to transmission losses. Actually, a high loss rate, due to random losses, can prevent the destination from getting not only coded packets, but also non-coded packets required for decoding. Consider a destination node that is the end-point of a TCP connection. Assume that, in order to correctly perform the decoding process, such node requires the reception of packets over two different wireless links.

As a consequence, the TCP flow will be exposed to a higher loss rate (compared to the store-and-forward scheme), since in effect it uses as last hop two wireless links instead of one. This will lead to more reductions of the TCP congestion window over time and, possibly, to the under-utilisation of link capacities. Such increased random losses may eventually counteract (or even cancel out) the throughput gains offered by NC.

This opens a first line of research to be tackled, since it is well known that some of the most widespread channel models lack of the required accuracy. We will use more complex models, able to reflect the bursty behaviour, which characterises real propagation environments, to see their effect on the performance exhibited by Wireless Coded Mesh Networks (WCMNs).

A second issue is a potential increase of packet loss synchronisation between TCP flows. In general, drop synchronisation is in principle difficult to happen in traditional TCP operation. However, in the context of WMNs with NC, the loss of a single coded packet will often be equivalent to *several flows experiencing simultaneous packet drops*. Therefore, we may expect loss synchronisation to be much more salient in coded WMNs.

Despite the substantial volume of research devoted to studying NC and wireless mesh networks, few *works* have focused on the issues arising from the interaction of TCP with NC, particularly in the context of multi-hop wireless networks.

Nevertheless, the poor performance of TCP over a WCMN has been mentioned in the first paper that dealt with NC and WMNs [Katti06]. In fact, in some experiments presented in this paper, Katti et al. found that NC fails to improve TCP performance, showing a throughput gain of just 2-3%. The authors explain this poor performance by the high loss rate and the frequent appearance of hidden terminal problems. They propose a protocol for opportunistically coding, COPE. They showed that COPE can provide a goodput gain up to 38% when all hidden terminals are eliminated and a sufficient traffic load is provided. This is relatively small compared to the throughput gain assessed when using UDP traffic, which varies between 300% and 400%.

In [Huang08], Huang et al. presented a testbed implementation of a NC protocol, in order to study the performance of TCP over a WCMN. This protocol is similar to COPE, in the sense that it looks for coding opportunities and uses XOR operations to code packets. However, there is no opportunistic listening; each node performs coding based only on what it has already sent or received, and not on what neighbours have received from other nodes.

The implementation of [Huang08] in mesh routers uses a “NC timer”, associated with each outgoing packet. This timer is used to delay sending the packet so as to wait for a coding opportunity. Huang et al. showed that the performance of TCP is sensitive to the value of the timer. A high value increases coding opportunities, leading to a lower loss rate (since coding packets decreases contention for the wireless medium) but it also increases the Round Trip Time (RTT) experienced by TCP flows. A small value of the timer leads to few coding opportunities but also to a lower RTT. They found that, with an appropriate empirical choice of the NC timer, the throughput gain offered by NC may vary from 20% to 70%, depending on experimental conditions.

One aspect that has not been tackled so far is the use of “randomly” deployed WMN; in this sense, most of the existing works assume simple network deployments (in some cases the butterfly network). It becomes really relevant to assess the impact of having randomly distributed nodes, which reflects better the WMNs considered in the framework of the SAIL project (DTNs, etc). Under these circumstances, the impact of hidden terminals, interference, etc might be much higher.

The main focus is therefore twofold: on the one hand, the flexibility of OConS is challenged from the point of view of the NC functionality and, besides, extensive analysis of TCP behaviour when NC is employed will be carried out (supported by the NC instantiation of the OConS framework).

As an illustrative example, one way of coping with the first issue identified earlier is by using a cross layer approach. In this sense, when a receiver cannot decode a packet, not because it has not received the coded packet, but because it has not correctly overheard another one, TCP will reduce its sending rate. As a consequence, TCP becomes affected by loss and errors that have not occurred in its path. A potential mechanism to overcome this situation implies sending back a new type of acknowledgments that asks the sender for retransmission without halving its congestion window (i.e. this requires a slight modification on the TCP implementation). When the TCP receiver receives a coded packet and could not retrieve it, it must at least be capable, using information from the coding header, to identify the TCP-connection that the original packet belongs to.

Regarding the second problem (the correlation between window reduction and the inability to properly decode packets at the destination), this first needs to be confirmed (by means of proper simulation analysis and, even better, real experimentation). Afterwards, the analysis will be used so as to propose appropriate solutions, taking advantage from the framework which is offered by the OConS architecture.

5.1.2.2 NC and the OConS Architecture

Traditional NC operation can be easily mapped on the OConS architecture as shown in Figure 5.2. NC can be said to operate at the network layer, and needs some extra-overhead, which is usually implemented between the network and the link layers.

IE: In order to be able to perform the corresponding tasks, the encoding node needs information about what packets have been already received at the destination, to be able to select those packets with which the original packet can be coded. The decoding node must keep track of a certain number of overheard packets that may be used for future decoding operations. When working on isolation, NC needs to gather this information with some own means; however, if integrated in the framework of the OConS architecture, NC might be able to benefit from the functionalities provided by the IE of the network.

DE: An encoding node must carefully choose the set of packets to be coded together to maximise their chance to be successfully decoded at the destination.

EE: It is the entity in charge of actually coding and decoding the TCP segments. Furthermore, in case of a decoding failure, the decoding node must identify the TCP-connection and notify the TCP receiver which will generate the appropriate acknowledgment.

In addition, NC operation needs to be able to interoperate the information with the rest of the OConS architecture. When establishing a network flow, the NC-related entity should be able to communicate to the requesting entity the possible performance figures which might be achieved, so that the corresponding connectivity services could be instantiated or not.

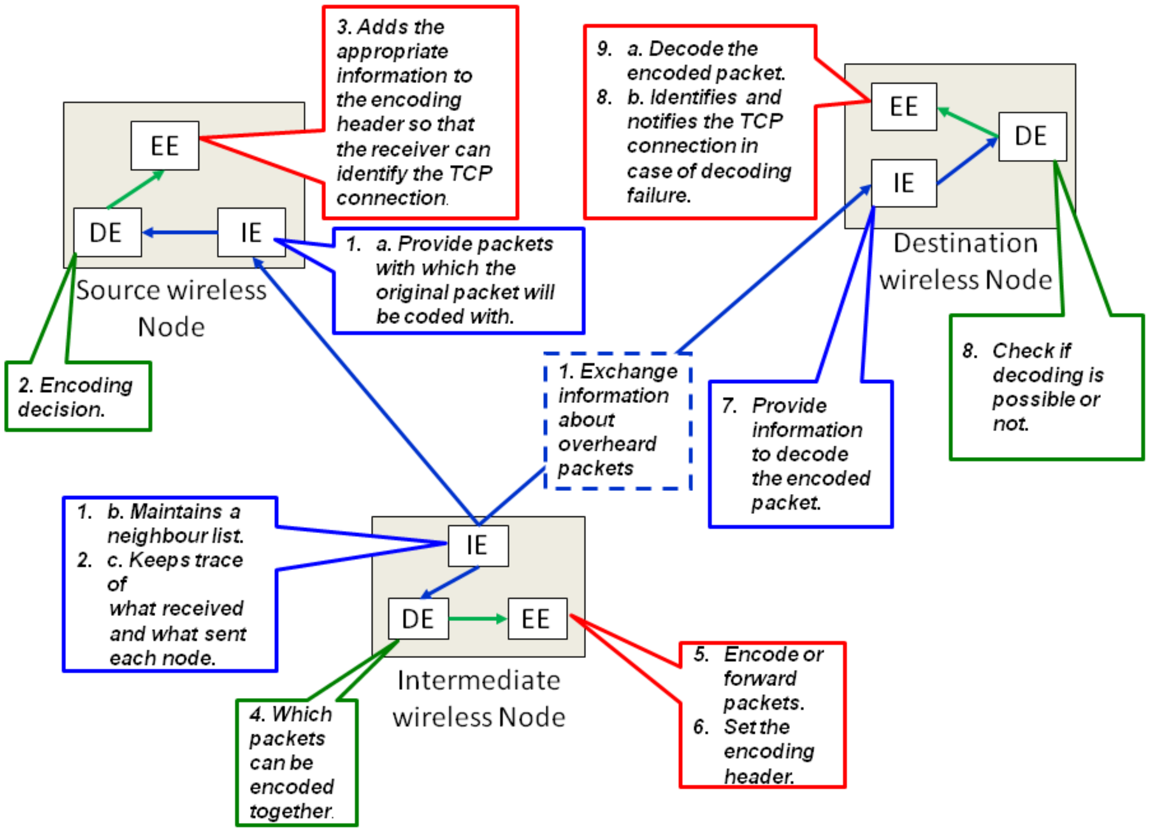

3. Adds the appropriate information to the encoding header so that the receiver can identify the TCP connection.
9. a. Decode the encoded packet.
8. b. Identifies and notifies the TCP connection in case of decoding failure.
EE
DE
IE
Source wireless Node
EE
DE
IE
Destination wireless Node
1. a. Provide packets with which the original packet will be coded with.
2. Encoding decision.
1. Exchange information about overheard packets
7. Provide information to decode the encoded packet.
8. Check if decoding is possible or not.
1. b. Maintains a neighbour list.
2. c. Keeps trace of what received and what sent each node.
IE
DE
EE
Intermediate wireless Node
5. Encode or forward packets.
6. Set the encoding header.
4. Which packets can be encoded together.

Figure 5.2: NC operations mapped onto the OConS architecture

Table 5-2 summarises the information which is deemed necessary by the DEs of the described mechanism.

Table 5-2: Information needed for NC (applied to TCP flows)

Resources	Node resources	• Buffer per destination containing the packets to be sent. • List of the identifiers of packets received by each neighbour. • Appropriate (memory and CPU) resources to encode and decode packets. • NC-timer.
	Network	• Neighbour list. • Transmission error rates per wireless link.
	Context	• Enabled promiscuous mode.
Policy	Strategy	• A choice between looking for the best coding opportunity and encode at the first opportunity. • Encode only the packets having a NC-header added by the source.

5.2 Network Control Functions for Multi-P

5.2.1 WAN Interconnectivity of Distributed Data-Centres for Virtual Networks

5.2.1.1 Context and Requirements

Many of the current and emerging Internet applications are hosted in large data-centres or even cloud networks. Data-centre operators use several geographically dispersed locations for performance, load sharing and resilience reasons. This however requires synchronisation of the different data-centres, and an efficient connection between the different locations is necessary. The relevant background has been briefly discussed in Section 2.3.

Typical data-centre (DC) architectures today consist of either two- or three-level trees of switches or routers building a private internet domain upon an Ethernet-based LAN. A distributed data-centre (DDC) consist of locally dispersed data-centre 'islands' which are connected and bridged over WAN technologies to a form a single private IP domain. The connecting WAN technologies can be L3 IPv4/IPv6- based, MPLS-based, or OTN (Optical Transport Network) based. Data-centre gateways or DC Controllers provide external reachability of the data-centre cloud, specific network address translations (NAT) into the internal private addressing scheme, and manage the data distribution via internal switching and routing strategies.

The main focus here is the management and control of the DC interconnections over heterogeneous WAN technologies, with multi-path optimisation capabilities over multiple layers (involving multiple protocols) and multiple administrative domains.

We summarise below the high-level requirements of such a scenario, as follows:

- Standard autonomous operation modes of the underlying L2 technologies (spanning tree, MAC-learning, ARP) shall coexist for the L2 based DC islands (migration aspect)
- Improvements of WAN interconnect between these DC islands on top of the basic mechanisms for bridging the L2-based DC islands in terms of
  - Scalability in numbers of DC sites and involved nodes (Ethernet scalability)
  - Higher efficiency of WAN Interconnect mechanisms (performance, energy, availability) than by L3-Overlay
- Focus on operation according to reference model below DC-|UNI|-|NNI|-|UNI|-DC, later on operation across multiple core network domains and multiple technologies (e.g. OTN).

5.2.1.2 Reference Model and Network Architecture

Augmenting the end-user and access oriented OConS use-cases, we introduce a network architecture reference model which focuses on edge-to-edge communication of (large) data-centres across the core network. That is to say that the OConS user in this case is a complete network (a single-sited or distributed DC , e.g. a data-centre cloud or service cloud) and thus creates a different demand for connectivity between such (business-, provider-) users and the OConS provider (in the role of a network operator or operator of a sub-network/domain).

The focus is on the autonomous behaviour between the (edge) domains (represented by service or data-centres) which may not necessarily be triggered directly by end user actions but may be more dependent on the type of service, its connectivity requirements and the aggregation of the traffic of users demanding this service. Traffic examples are transmission of workflows between several service providers, e.g. video creation, production and distribution by service providers like media companies, agencies, broadcasters. Other cloud service (co-) operation requiring reliable and cost-effective connectivity across network domains includes the migration of data bases and servers, processes for backup, load balancing, energy efficiency, etc. Large distributed data-centres with variable processing, storage and networking resources create a global challenging interconnectivity and transport demand to be served by cooperating network operators as OConS providers, providing their resources (e.g. optical transport networks) to build a provider-to-provider service.

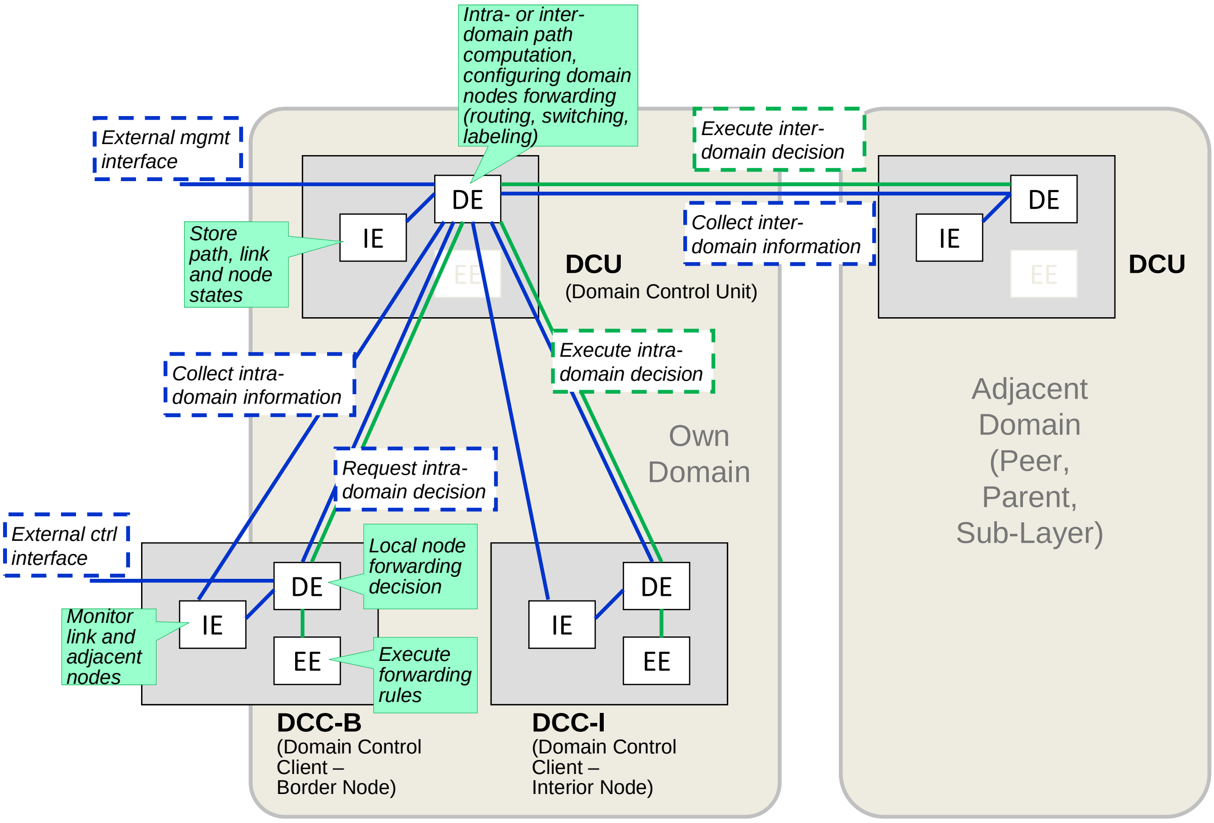

Intra- or inter-domain path computation, configuring domain nodes forwarding (routing, switching, labeling)
External mgmt interface
Execute inter-domain decision
DE
IE
EE
DE
IE
EE
Store path, link and node states
Collect inter-domain information
DCU
(Domain Control Unit)
DCU
Collect intra-domain information
Execute intra-domain decision
Own Domain
Adjacent Domain (Peer, Parent, Sub-Layer)
Request intra-domain decision
External ctrl interface
Local node forwarding decision
DE
IE
EE
DE
IE
EE
Monitor link and adjacent nodes
Execute forwarding rules
DCC-B
(Domain Control Client – Border Node)
DCC-I
(Domain Control Client – Interior Node)

Figure 5.3 Reference Model of Intra- and Inter-Domain Control Architecture

Therefore we follow a model that is based on a centralised control server entity per domain (as a Decision Making Entity; here called Domain Control Unit DCU) which is connected to each of the client switching or routing entities called Domain Control Clients (DCC) within the domain (both for monitoring traffic and collecting information by an Information Management Entity; and manipulating forwarding and routing tables in the switch/routing entity, mapped onto the Execution and Enforcement Entity). According to their connectivity, the DCCs can be categorised into an 'interior node' role or a 'border node' role with external links (see Figure 5.3).

This model is grounded in the approaches for the Path Computation Element (PCE) of the IETF (see Section 2.2) and in the advanced capabilities of the OpenFlow switch concepts (see Section 2.4), which lead us to develop the advanced innovative architectural approach combining elements of both for an intra- and inter-domain connectivity control concept.

The proposed network architecture model consists of (see Figure 5.4 for a high-level network view):

- Data-centre domains (DCD1, DCD2) at the edge of the network, playing the role of client networks
- Multiple Core network domains (CND3, CND4) operated by the same or different providers, even operating at different (layer) technologies.
- Domain Control Units (DCU1-4) as control server nodes which manage the connectivity and flows of their own domain in each of the involved domains (e.g. compute paths,

enforce path setting by table manipulation in client nodes) -- acting as OConS DEs) and cooperate with adjacent network domain DCUs.

- Edge and core switches/routers as nodes with DCC client functionality (e.g. mainly link and traffic monitoring, neighbourhood and resource discovery as well as table-based routing, switching and forwarding) -- acting as OConS IEs and EEs. These roles can be separated into (domain-) interior and (domain-) border nodes with external link connectivity.
- Control Interfaces between
  - DCU servers and DCC clients for domain-internal monitoring traffic flows, triggering of route and path computation and controlling flow forwarding tables.
  - Data-centre-based DCU and Network-based DCU (UNI client or user-network interface) for cross-domain control of multilayer flows, paths and routes, including support of edge-to-edge optimum connectivity computations.
  - Network-based DCUs (E-NNI network-network interface) for cross-domain control of multilayer flows, paths and routes, including support of domain-overarching connectivity computations (multipath capabilities).

This architectural model supports the following ‘Connectivity Services’ between the data-centre domains and the core network domains, which can be divided into the DC-driven connectivity services and the network-driven connectivity services:

- Discovery, advertisement and monitoring of available nodes and links, their topology and vicinity relationships within a domain, and with some possible abstractions also across domain boundaries.
- Address resolution and translation: An address resolution request received (at the gateway or forwarded to the DCU) from the interior will be either:
  - Directly responded by a local proxy/cache function, or
  - Forwarded/broadcast to all virtual connected DC domains and their positive responses returned into the cloud
- Supporting inter-process communication between the DC domains with mapping of virtual to physical addresses/identifiers between the DC and the core network domains: when the destination location and address of the target process is already known, direct forwarding of any unicast message can be provided by the gateway or DCU (acting as a virtual bridge for the overlay). Otherwise a multi-domain or multi-layer aware address and path discovery mechanism has to provide the required address and forwarding information (including required mappings).
- Identification, monitoring and aggregation of internal and external flows: the traffic between the sub-clouds and to the exterior WAN may be monitored by the local cloud control entity, the flows can be identified, and possibly aggregated if appropriate. For those aggregated flows a dedicated path can be set up or modified via the UNI, thus optimising the required connectivity between sub-clouds. This typically happens to ‘elephant flows’ which exceed a certain bandwidth demand and need to ensure a certain latency, caused by migration of a whole server process, or on-going live synchronisation between content servers.
- Flow exchange at network level: The advantage of using flow control enabled switches as clients, e.g. OpenFlow switches, is the integration of the three OConS tiers functional approach: measurements (flow counters), decision making by (rules) and enforcement (action). A centralised intra-domain controller is able to take a higher level decision and make measurements, while the inter-domain controller provides the policy management interface to the provider. This controller makes it possible to control critical provider owned data, deploy AAA and guarantee a basic level of flow control at the same time. Another advantage is the layer/protocol agnostic of flow control.

A more detailed discussion of the network-related DC interconnect mechanisms is given in the following section and, in particular for the use case of 'path setup' over the UNI in the context of specific OConS-based message flows, contained in Section 8.4.

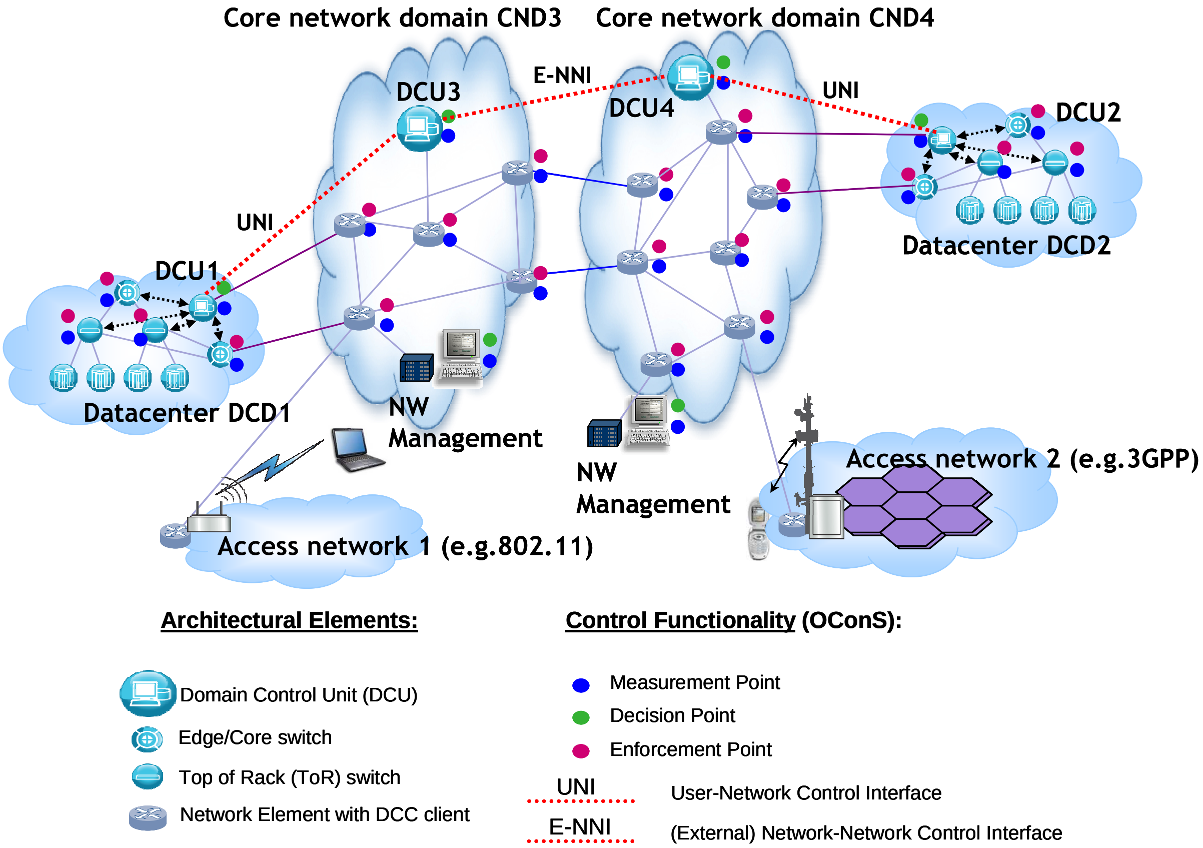


Figure 5.4: Functional Mapping for Interconnectivity of Distributed Data-centres

5.2.1.3 Multi-Domain Capable Interconnect Mechanisms

Note that in the DDC scenario of Figure 5.4 the DCUs in the data-centres (at the edge of the network) and the DCUs in the core network domains operate functionally at different layers. The main function of the data-centre DCU is to build a Layer2 overlay network over the core network domains, whereas the main function of the core network domains is managing the path and flow-based connectivity, optimised over multiple domains or layers in the WAN.

For the latter problem of finding and setting up edge-to-edge paths across multiple domains, both fully distributed peer-to-peer approaches as well as hierarchically-centralised architectural approaches are considered, as outlined in Section 4.2.2. Both concepts have close relationships to the concepts developed for the PCE as discussed in IETF. However, the DCU architecture presented here also involves new ideas for the coordinated execution and enforcement by the DCC clients – i.e. switches and routers -- by direct manipulations of forwarding and routing tables as supported e.g. by the OpenFlow switches or the FORCES protocols of IETF. Related multi-domain approaches are studied also in the STRONGEST project [STR-D3.2], where multi-domain PCE-based architectures are discussed in the context of GMPLS.

In this first analysis, we describe the centralised hierarchical solution based on an additional 'umbrella DCU' (U-DCU), a parent entity, which coordinates the path computation across its underlying domains.

The umbrella DCU is responsible for the interdomain path computation between the domain border nodes and the global multi-domain edge-to-edge optimisation while the underlying local DCUs in each domain perform the intradomain path computation and the local setting of paths. The umbrella DCU continually collects the external connectivity information of the covered domains (incl. their border nodes), and thus is able to compute optimum paths from source to destination in different domains without the necessity for predefining the sequence of domains to be crossed. The hierarchical umbrella DCU approach allows also for the balancing of load among the contributing domains.

The umbrella DCU announces the set of domains for which it is responsible and the domain DCU register to it. For initiating an interdomain path computation, the source domain client (router or switch) first sends a path computation request to its local DCU containing the destination to be connected.

If the local (source) DCU cannot find the requested destination in its domain, it forwards the request to the umbrella DCU. The umbrella DCU determines the destination domain and its DCU based on topology information collected and cached. Based on the knowledge of border nodes of registered domains and their inter-domain connectivity, the umbrella DCU determines viable paths from source to destination domains via possibly several intermediate domains. If the required intra-domain sub-paths and their metrics are not already available, the U-DCU will request them from the involved domains. With this sub-path information collected, the U-DCU is able to optimise the global edge-to-edge path. The U-DCU then provides the selected ingress border nodes with the information necessary for them to forward the incoming packets to intra-domain paths towards the appropriate egress border nodes.

After all, the U-DCU sends a resource confirmation message to the source DCU together with the next egress border node. Upon receiving this message, the local source DCU computes the best intradomain path and delivers the information to the DCC.

The umbrella DCU can further optimise the interdomain communication according to a set of parameters defined by the operator. For example it can balance the load among its underlying domains. To fulfil this task it must be provided by the local DCUs with load state information of the network. The umbrella DCU can also engage in policy-based path computation where it determines the appropriate paths with regard to a set of policies stored by the operator in a policy data base.

Currently, the mechanisms in a multi-domain scenario are considered, as explained above. In a further step, the same principles will be applied in a multi-layer architecture to study the multi-layer edge-to-edge path optimisation.

5.2.2 Control Functions for Multi-Layer Networking and Transport

For the MPLS (see Section 5.2.1) transport service, two kinds of interfaces are defined [RFC5921]. One is the User-Network Interface (UNI), the other the Network-Network Interface (NNI). The UNI exists between the Customer Edge (CE) node and the MPLS-TP Provider Edge (PE) node. The NNI is present between two MPLS-TP PE nodes, e.g. in different administrative domains. Both NNI and UNI handle packets between a service layer and an underlying client layer. Thus a packet from the control plane above is associated with a traffic flow of the transport service data plane for transporting it.

In the data-centre interconnect use-case (see Figure 5.4 and Section 8.4) the UNI resides between the data-centre (DC) DCU and the Core Net (CN) DCU. The NNI is foreseen between two CN DCUs, see Figure 5.4. We now elaborate on the interface functionality between a DCU-server/controller and the DCC client residing in a DCU controlled switch, e.g. in the Top of Rack (TOR) switch in Figure 8.6. The control interface from the DCC client IE towards the DCU server IE has the following functionalities: First the triggering of route and path computations in the DCU is conveyed. Then the DCC client is monitoring the traffic flows and conveying the counters to the DCU server. Next the DCC client attaches itself to the DCU server to allow the server to discover nodes and their properties. After that the DCC client also signals updates on node resources like port up/down or link up/down and thus the DCU server can update the network topology accordingly. While some of this functionality can be implemented using OpenFlow there are some additions needed, e.g. today the OpenFlow protocol does not include event generation by the OpenFlow switch, e.g. if a threshold overruns it cannot be signalled to the controller easily.

The control interface from the DCU server/controller DE towards the DCC client EE has the following functionalities: First the establishment of the forwarding entries in the DCC client, and thus the control of the involved forwarding tables in the node, is handled. Then the DCU server can take up/down a port on a client node. Although technologies like Wake on LAN, they are not reliable concerning persistence and repeatability. This is why operators today are hesitant when it comes to introducing such technologies, i.e. network operators do not want to take up/down a node, a link, e.g. an optical transport link. Moreover there is no possibility in current solutions like OpenFlow to convey a message to take up/down unused ports/links/nodes for decreased energy consumption.

Summed up the high level information needed to control the DCU network described above is listed in Table 5-3. A fully worked out example, including a message sequence description can be found in Section 8.4.

Table 5-3: Information needed for the DCU controlled data-centre interconnect network

Resources	Networks	• Current network topology map • Current bandwidth available on each link
Requirements	Data-centre Application	• Source and destination address for each data-centre interconnection • Bandwidth needed per flow
Policies/ Preferences	Strategy goals	• Minimise delay • Minimise energy consumption

Next we look at the Inter network provider issues, i.e. what information can be disclosed when transferring information between two DEs in neighbouring DCU servers of two different administrative domains. A data-centre based DCU can communicate with a network based DCU using the UNI user-network interface for cross-domain control of flows and edge-to-edge optimum connectivity computations.

As always there is the problem of the inter-provider confidentiality, i.e. a provider usually is very reluctant to expose internals of his network to other providers. If a path has to be established running through another provider's network the concept of so called loose paths can be applied. With that a provider only tells a list of border routers that a potential path can use when requesting a path computation. However such a loose per domain path computation does not necessarily guarantee the use of an optimal constrained path.

For the end to end path there are two architectural/service variants on where to terminate paths. Paths can end in the PE or in the CE. This choice defines the role and the function of the UNI as control interface between the core network at PE and the data-centre networks at the customer edge.

Not all path computations must be initiated by a DCU necessarily. There is also the possibility to let an Operations and Maintenance Centre (OMC), e.g. in the core network domain explicitly establish a path. For this an additional OMC-DCU communication is necessary. Last but not least another research problem to be tackled is to use one central DCU or a couple of distributed DCU in the target architecture. Our research will show the pros and cons herewith.

We think that the OConS architecture is capable to cope with the challenges listed above. And with a proper design of the control interfaces we have a good chance to make the design of a novel open connectivity architecture a reality and show selected features in a prototype implementation.

5.2.3 Advanced Multi-Layer, Multi-Domain Control Layer

5.2.3.1 Context and Motivation

IP off-loading to layers below, normally to the optical layer, is becoming popular. While promising results (regarding savings) are obtained within a domain, the edge nodes become the bottleneck, because they have to do the inter layer up and down conversion between the physical layer and the network layer (see Figure 5.5).

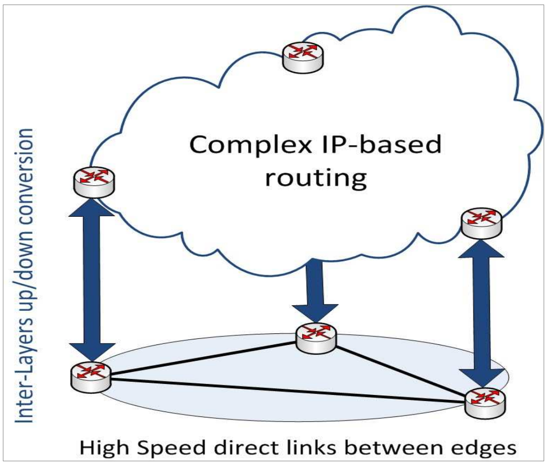

Complex IP-based routing
Inter-Layers up/down conversion
High Speed direct links between edges

Figure 5.5: Physical and network inter-layers up/down conversion at the edges

In the inter domain scenario, we have both computation and bandwidth bottleneck. Besides enduring all the inter domain traffic, the inter domain interface router should be powerful enough to perform the up/down layer communication and speed adaption to the inter domain link (and maybe even more functionalities in the future), see Figure 5.6.

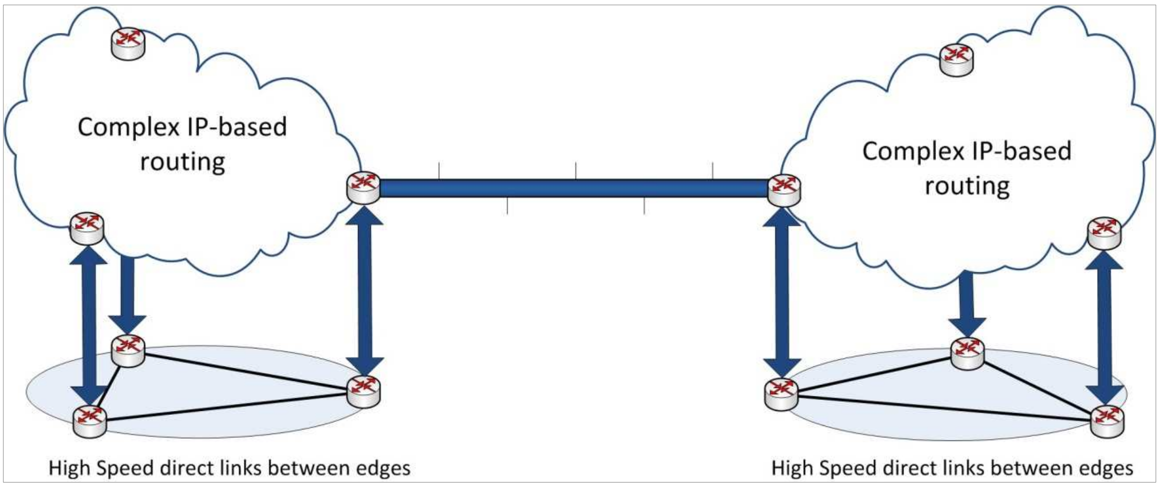

Complex IP-based routing
Complex IP-based routing
High Speed direct links between edges
High Speed direct links between edges

Figure 5.6: Inter-domain inter-layers up/down conversion

A first attempt to overcome the problem was the multi-domain MPLS interface (Inter-Autonomous System (AS) MPLS). It can be divided in two types: the MPLS/BGP L3VPN which is mature and widely applied, and the MPLS L2VPN which is still not standardised.

The Inter-AS MPLS VPN methods can be further divided in three categories:

1) Option A: known as the back-to-back method where the Border Router of an AS serves as the PE for its own AS and as the CE for the other AS (and vice-versa); between two AS Border Routers regular IP forwarding is used without any LSP.
2) Option B: known as the single-hop multi-protocol exterior-BGP method where a given AS Border Router can be itself the next hop (single-layer LSP between the Border Routers is used) or the next hop is the Border Router of the other AS (double-layer LSP between the PE and the Border Router of the other AS is thus needed).
3) Option C: known as the multi-hop multi-protocol exterior-BGP method where the LSP is built end-to-end by the PE routers; the ASes Border Routers only provide routing between these PE routers with double-layer LSP between them.

Each option has its advantages and disadvantages, to say a few; Option A is simple to implement and to maintain but does not scale very well and has QoS difficulties; Option B, scales better and supports enterprise QoS transparency but requires further coordination, agreements and trust models between ASes; Option C is more complicated and harder to maintain than B but offers better scalability. A new option, a combination of A and B, was also discussed by the IETF.

Inter-AS MPLS models are normally applied within internal interfaces in AS confederations or confederation-alike (two ASes belonging to the same organisation). Limitations of option A limit its use to scenarios with a small number of VPNs, while the need of trust relation and agreements for setup and maintenance in Options B and C, makes the implementation further harder in organisationally separated ASes. We need a solution that provisions the currently available technologies and supports inter-As communication over totally separated ASes and possibly supports further classes of interfaces and switching technologies (heterogeneous ASes). GMPLS would be a candidate technology.

5.2.3.2 Reference Scenario for Inter-AS VPN

The Inter-Autonomous System (Inter-AS) VPN generalisation using the (G)MPLS is shown in Figure 5.7.

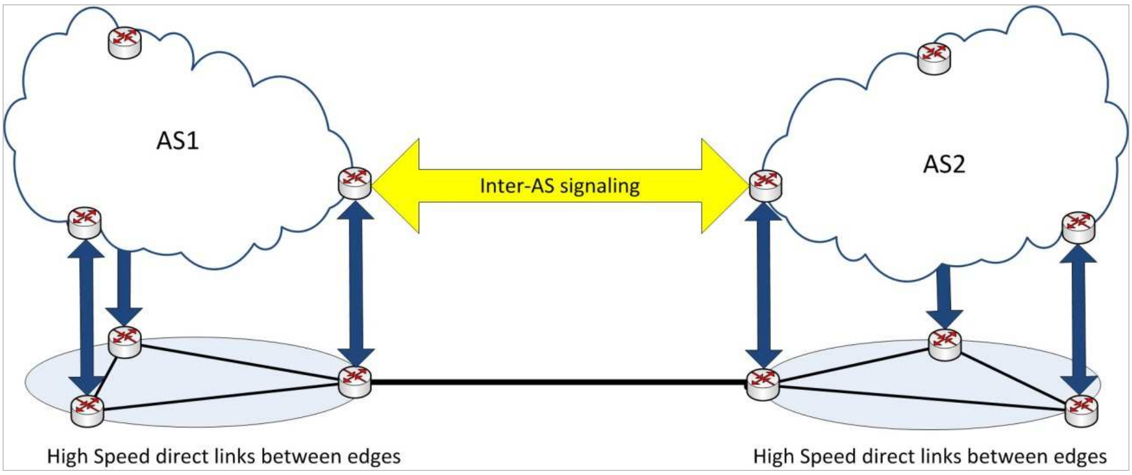

AS1
Inter-AS signaling
AS2
High Speed direct links between edges
High Speed direct links between edges

Figure 5.7: Inter-AS VPN generalisation using the (G)MPLS

As requirements coming from the previous scenario, we can say that the involved peers require to have compatible lower (i.e. physical) layers that may be provided by GMPLS and a common layer 3 understanding which includes handling of, at least, the following protocols and services: 1) IPv4, 2) IPv6, 3) IPv4-VPN, and 4) IPv6-VPN.

5.2.3.3 Required functionality and Algorithms

The minimum required functionality required to implement this scenario are the following:

- **Scenario wide address/name resolution mechanism:** Systems on both ASes need to be addressable with the limitations imposed by the service they are going to subscribe for. E.g. IPv4 end points must be addressable by other IPv4 endpoints. The same holds for IPv6 endpoints. Regarding endpoints of IPv4 or IPv6 VPNs, the addressability must be limited to the members of a given VPN.
- **Loop-free forwarding:** Systems on peering ASes must be able to communicate using a loop free path in the forwarding plane.

Additional requirements which need to be satisfied may include:

- **Traffic engineering:** It is a desirable feature that the paths between end-systems can be controlled following a set of commonly agreed rules between the peering ASes
- **Topological information to be kept local to the AS:** Peering ASes should be able to establish connectivity between end-systems without the need to disclose topological information.

As for the investigated algorithms, the main objective of this activity is to provide new functionalities or extend and optimise current implementations in the fields of:

- Edge-to-edge multilayer optimisation
- Generalised IP offloading algorithms

5.2.4 Multi-Path Extensions for Information-Centric Networks

Network use has evolved to be dominated by content distribution and retrieval, while networking technology still speaks only of connections between hosts. Accessing content and services requires mapping from the 'what' that users care about to the network's 'where'. Information Centric Networking (ICN) is a new paradigm in networking, which treats content as a primitive - decoupling location from identity, security and access, and retrieving content by name. There exist a number of ICN architectures that introduce new approaches to perform routing named content. They derive ideas heavily from the Internet Protocol achieving simultaneous scalability, security and performance.

Though multi-path capabilities are inherent by design in most of these ICN architectures, a number of formal strategies need to be incorporated for the content to be carried over the available multiple paths. There are a number of objectives for utilising the multiple paths in a content-based network:

- Bandwidth aggregation through the use of multiple paths to carry different content types that arrive from the same content producer (server).
- Bandwidth aggregation through the use of multiple paths to carry content of the same content type over multiple paths by splitting the content types.
- Improving content receipt reliability by replicating content of the same content type into multiple paths.
- Improve content receipt reliability by making content made available in new locations during mobility without having to re-request the content.

To provide these capabilities in a content-based network, mechanisms must be adopted that sets the stage in the ICN architecture as well as the underlying transport protocols used. These mechanisms being part of the OConS Architecture will interact with the ICN protocols and the transport protocols that carry the content in different transport networks.

The basic functionality of these mechanisms performs the following activities to achieve the previously mentioned objectives, as illustrated in Figure 5.8:

- Obtain information on current paths, content streams, mobility, user's preferences/policies.
- Determine the strategy to be adopted to distribute content requests and the receipts over multiple paths.
- Enforce the strategies identified in the ICN network by informing the enforcement points.

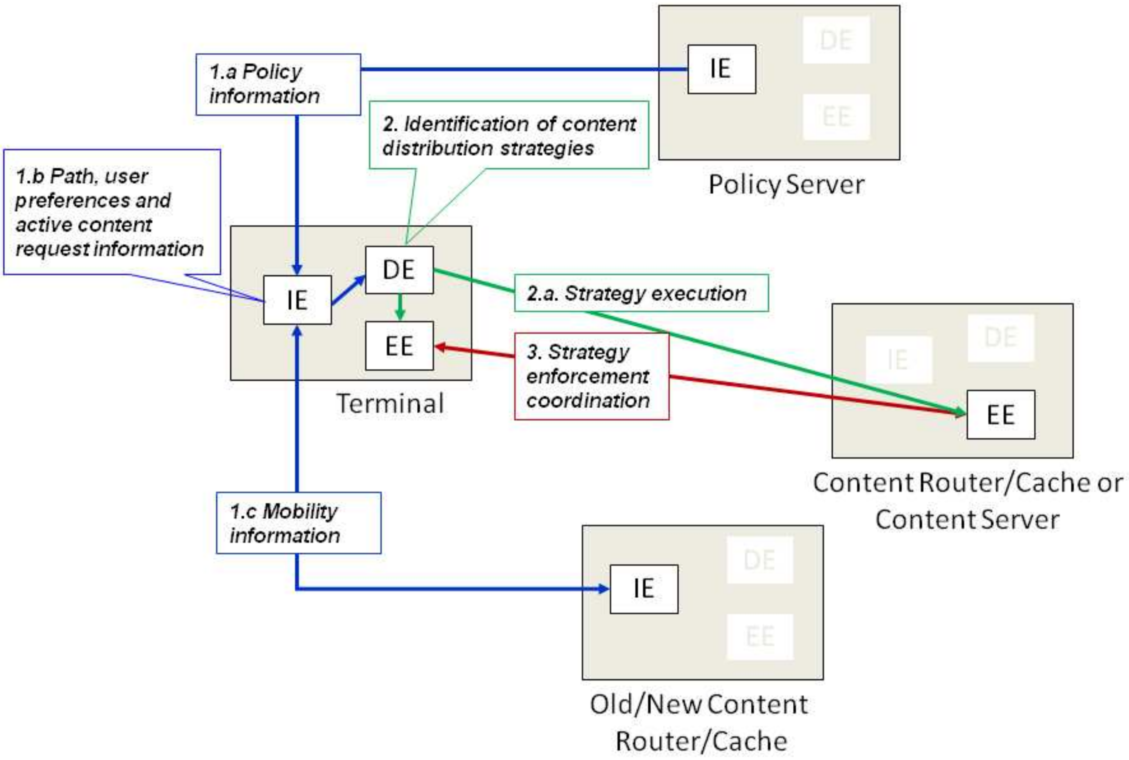

1.a Policy information
IE
DE
EE
Policy Server
2. Identification of content distribution strategies
1.b Path, user preferences and active content request information
IE
DE
EE
Terminal
2.a. Strategy execution
3. Strategy enforcement coordination
IE
DE
EE
Content Router/Cache or Content Server
1.c Mobility information
IE
DE
EE
Old/New Content Router/Cache

Figure 5.8: OConS operations related to Multi-path for ICNs

The messages that carry the request for content and the content themselves are extended to incorporate the usage of multiple paths. These messages carry additional information to the different OConS entities to setup the content forwarding strategies:

- Information Entity
  - Use of local (not over network) messaging to get information about paths, user preferences and content requests
  - Use of content request messages to get policy and currently accessible content caches for mobility information
- Decision Entity
  - Use of the content request messages to carry the identified strategies
- Enforcement Entity
  - Use of request and content messages to coordinate the enforcement of strategies

The information needed by the decision process is listed in Table 5-4.

Table 5-4: Information needed for Multi-path for ICNs

Resources	Networks	• Paths information • Adherence policies • Content caches
Requirements	Applications	• Bandwidth aggregation • Reliable content delivery
Policies	Users/Operators	• Selection of paths based on user preferences and current connectivity status

5.2.5 Efficient Multi-Path over MPLS for IPTV services

MPLS technology is widely deployed as a core technology of IP networks. MPLS support for Traffic Engineering capabilities (MPLS-TE) is very attractive, and where traffic can be routed over Label-Switched Paths (LSPs). The LSPs can be configured to support QoS, or as backup paths for timely recovery of path failure.

The LSPs are set up through non-default routes. Using routing protocols such as OSPF-TE, an MPLS-TE router is provided with a map of the network topology and with information about the bandwidth available on each link. Each ingress router uses this map in order to find a route with sufficient bandwidth to an egress router, for each traffic aggregation unit. The ingress router then establishes an MPLS LSP over this route and uses it for delivering the traffic aggregate through the network.

The LSPs can also effectively split one flow into multiple paths (sub-flows). Multi-path arrangement is beneficial for load balancing, avoiding emerging congestions. It can also split high-bandwidth service (e.g. IPTV, or HDTV) over multiple low-bandwidth LSPs (inverse multiplexing). Splitting a traffic aggregation unit through multiple paths is possible because it consists of many sub aggregates that can be routed over different paths.

These benefits, however, do not come for free; there is a management overhead for the path setup, which must be considered and evaluated. While using multiple LSPs improves the data plane efficiency, it also increases the control load imposed on the network. This control load increases as a function of the number of LSPs and the number of nodes that carry them. Thus, minimising these measures presents a clear trade-off.

We will conduct an optimisation research, in which we identify the extent at which splitting a flow into multiple sub-flows is still beneficial, considering the path setup and maintenance overhead. We will further study this trade-off from both theoretical and practical perspectives. We will define a few formal definitions for this optimisation problem, identify efficient algorithms for solving them, and analyse their performance, eventually concluding with the algorithm that provides the best trade-off.

Our research aims at minimising the control load imposed on the network – by minimising either the number of LSPs, or the number of nodes that carry them.

Our study is not part of the real-time OConS mechanism; our mechanism is an optimisation study that is executed in a simulated or experimental network, in order to gain better understandings regarding the shortcoming of current provisioning, and to identify possible enhancements, new dimensioning, or better configuration for it.

The results of our study can be used as benchmarking that guides network operators with regards to the extent at which Multi-Path is beneficial and is justifiable (as related to the cost associated with Multi-Path). Operators of OConS networks, get bulks of information from time to time (from the IEs). They then perform off-line analysis, utilising our mechanisms which are acting as "remote" DEs. The output of our mechanisms is then fed back into the corresponding EEs (e.g. in the form of new configuration), which are then transferred to the corresponding EEs, as part of the operation and maintenance tasks of the network.

The information needed by the decision process is listed in Table 5-5.

Table 5-5: Information needed for Multi-Path over MPLS for IPTV services

Resources	Networks	• A map of network topology • Bandwidth available on each link from the network topology
Requirements	Applications	• Source and Destination addresses (per flow) • Bandwidth requirements (per flow)
Policies/ Preferences	Operators	• Multipath setup with minimal number of LSPs • Multipath setup with minimal number of nodes along each LSP

5.2.6 Multi-Path and Multi-Protocol Transport for Enhanced Congestion Control

5.2.6.1 Context and Motivation

Mobile devices increasingly have a number of heterogeneous wireless interfaces that may be used separately or in parallel to provide “always best connected” networking. On the one hand, the simultaneous use of multiple paths can improve the end-to-end performance (delays, throughput...) but, on the other hand, users of mobile devices are primarily interested in the QoE of their applications, rather than the details of the network performance. A multi-path transport protocol can match both goals by providing a set of options to fit with the application requirements, and by deploying an adequate congestion control mechanism that allows a better use of multiple paths once these options are chosen.

Transport protocols have no information about the application or content carried within it. Therefore for the goal of improved QoE, this information needs to be passed on to transport by either the application or optionally by the user. Additionally, the transport protocol needs to have a number of available options with regard to the type of congestion control (e.g. window based or rate based) and reliability.

A multipath transport protocol should also seek to maximise throughput over all the available paths, while remaining fair to other transport flows that share one or more bottleneck links with it. When sub-flows of an end-to-end multipath flow share a bottleneck along their paths, for fairness reasons they must appear no more aggressive than a single TCP flow through that bottleneck, and when the sub-flows have distinct bottlenecks, they must seek to independently maximise their throughput through the bottlenecks.

We propose two mechanisms; the first one allows the transport protocol to choose the algorithms and the resources according to the user and/or application requirements, and the second one aims at improving the simultaneous use of multiple paths by increasing the end-to-end throughput without jeopardising the performance of the concurrent TCP flows.

Our goal is to design and evaluate (either via simulation or experimentally) mechanisms which enable a number of different types of congestion control and reliability in a multi-path transport environment, adapted to the application requirements. We are thus able to consider the mechanisms to exchange information between the application and transport layer. The optimal choice of congestion control and reliability mechanism applicable to a specific type of application or content is made within the transport protocol, using information provided by the application and/or the user.

5.2.6.2 Example of mechanism with different types of congestion control and reliability

A good application framework for this mechanism is HTTP, as it can carry information about the type of object transferred. A transport protocol framework that can be used as a base for the proposed work as it already has a lot of built-in flexibility is the Stream Control Transmission Protocol (SCTP), with Concurrent Multi-path Transfer (CMT) extensions. Based on the available information from e.g. HTTP, the new transport mechanism would define selectable congestion control and reliability for sub-flows as an extension to CMT-SCTP.

As an example, an HTML5 web page may have four different object types: text, video, audio and image. The browser requests a web page including a number of objects and required reliability (for a specific QoE). This is sent to the web server, and will be fed to the transport protocol on the sender (server) side. The IEs, as defined by the OConS architecture, would be located for this mechanism within the network (for the information about the available paths), the application (for the information about the content type) and, optionally, also the user would can provide direct input via an IE (e.g. with regard to acceptable QoE for various application and content types). Accordingly, both the DE and EE would be integral components of the transport protocol. Likewise, the EE actions will also affect the path's conditions, which are hence observed by the respective IEs (closing thus the feedback loop).

The OConS operations related to Multi-Path and Multi-Protocol Transport for Enhanced Congestion Control are shown in Figure 5.9. Furthermore, Table 5-6 presents the information necessary by this mechanism.

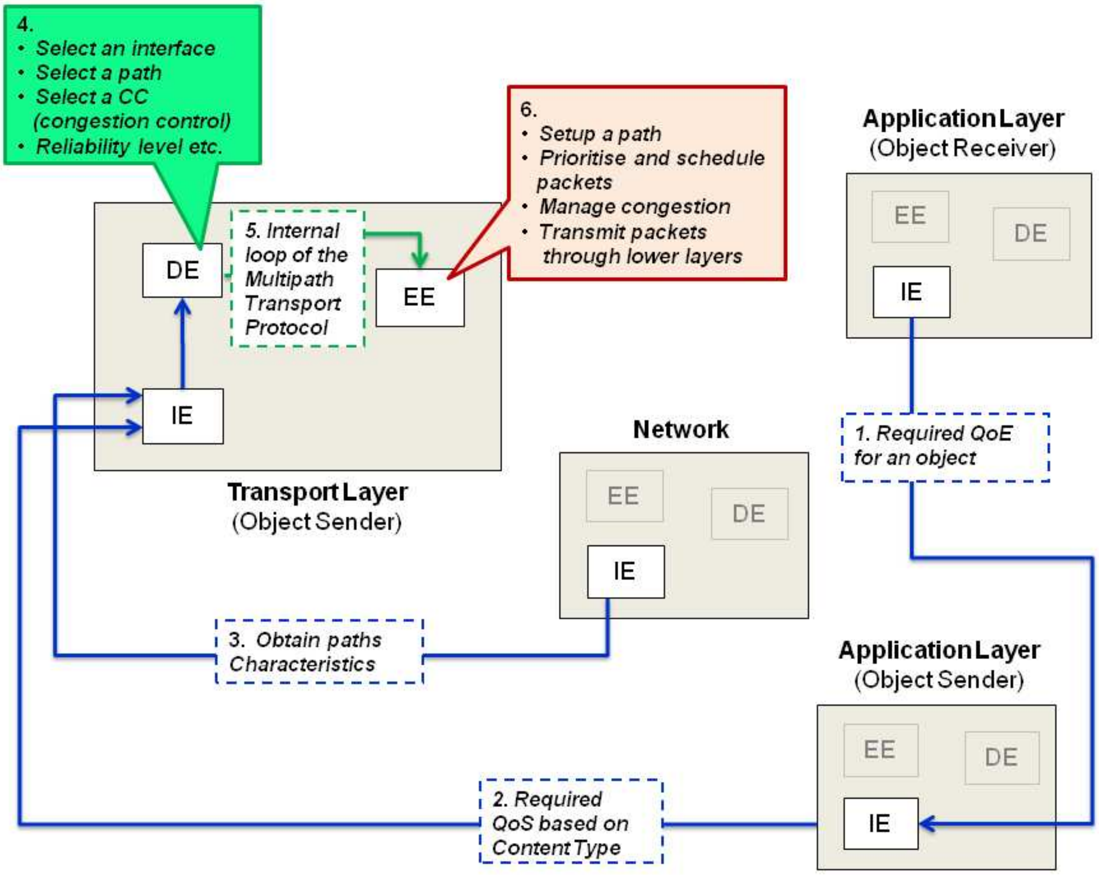

4.
• Select an interface
• Select a path
• Select a CC (congestion control)
• Reliability level etc.
6.
• Setup a path
• Prioritise and schedule packets
• Manage congestion
• Transmit packets through lower layers
Application Layer
(Object Receiver)
EE
DE
IE
DE
5. Internal loop of the Multipath Transport Protocol
EE
IE
Transport Layer
(Object Sender)
Network
EE
DE
IE
1. Required QoE for an object
3. Obtain paths Characteristics
Application Layer
(Object Sender)
EE
DE
IE
2. Required QoS based on Content Type

Figure 5.9: OConS operations related to Multi-Path and Multi-Protocol Transport for Enhanced Congestion Control

Table 5-6: Information needed for Multi-Path and Multi-Protocol Transport

Resources	Application Layer	• Object receiver (HTML5 browser) will primarily request *preferred QoE* (e.g. image, video, text) from the object sender. The object receiver may also optionally negotiate for required congestion control and reliability. • Based on the required QoE and other parameters, the object sender (web server) will decide on the *preferred method of congestion control* and *level of reliability* to generate messages/packets for the transport layer.
	Transport Layer	• Transport layer will use its internal protocol loop to gather per-path information, e.g. *RTT*, *loss* or *available capacity*.
	Network Layer	• Network layer can provide additional information (e.g., *indication of loss* or *congestion over a path*) for the transport layer to select a network interface, path, as well congestion and reliability level management.
Policy	Strategy Goals	• Content-based and/or application *required congestion control algorithm*. • Content-based and/or application *required reliability level*. This will have a direct impact on packet retransmission, scheduling and path selection.

5.2.6.3 Example of mechanism with simultaneous use of multiple paths

The simultaneous use of multiple paths must improve the end-to-end throughput, compared to the use of a single TCP flow over the best path. However, it must be fair at every bottleneck, i.e., it must not get more bandwidth from its paths crossing a common bottleneck than what would get a TCP that uses the best path crossing such bottleneck.

Thus, our second objective is to design a multipath congestion control mechanism that fulfils both these performance and fairness constraints. This mechanism must be able to learn and react appropriately wherever and whenever bottlenecks are shared or distinct. It must also detect and react to bottleneck shifts caused by traffic pattern changes or re-routing.

Figure 5.10 shows how this mechanism will be implemented at the sender side of the transport layer connection according to the OConS architecture:

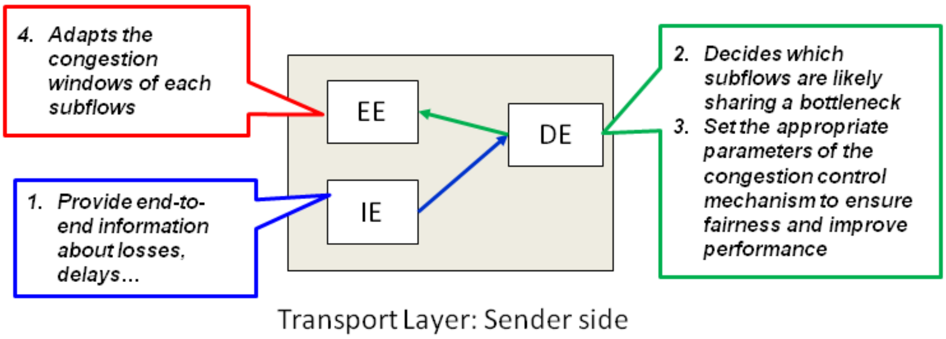

4. Adapts the congestion windows of each subflows
EE
DE
IE
2. Decides which subflows are likely sharing a bottleneck
3. Set the appropriate parameters of the congestion control mechanism to ensure fairness and improve performance
1. Provide end-to-end information about losses, delays…
Transport Layer: Sender side

Figure 5.10: Mechanism at the sender side mapped onto the OConS architecture

5.2.7 Multi-P Decision and Transmission for the Interconnection of 3GPP and non-3GPP Systems

The goal is to investigate the Interconnection (and cooperation) of 3GPP (e.g. LTE) and non-3GPP systems (e.g. Wireless Local Area Network (WLAN)) by means of the Multi-P transmission. Multi-P transmission is an enhancement of the current 3GPP handover strategies between 3GPP and non-3GPP systems. These enhancements involve the simultaneous employment of multiple wireless interfaces of the UEs, to increase transmission reliability and data rates, and, in turn, to achieve better UE performance and optimised network resource utilisation.

The key part of the Multi-P transmission is the decision process: selection of interfaces of the UE and access alternatives (at the network side) that should be used for the transmission and how to split the traffic flows. There are two possibilities of the Multi-P decision as depicted in Figure 5.11 and Figure 5.12: UE-controlled and Network-controlled.

Briefly, UE-controlled Multi-P decision is mainly made by the preferences of the user, e.g. pricing-first or performance-first, Figure 5.11. Since the UEs have little information about the network status (such as traffic load and link congestions), the focus of our investigation is the Network-controlled Multi-P decision which allows for a wider range of decisions, as depicted in Figure 5.12.

In the following, the Multi-P Transmission process is described. From an architectural point of view, the interconnection between 3GPP and non-3GPP (in our case we have chosen LTE and WLAN as case study) is possible at the Packet Data Network Gateway (PDN GW). Hence, it becomes a reasonable location for the DE of the Network controlled Multi-P decision. For the LTE side, the IEs are located at the eNodeB (eNB) and the access gateway (aGW). For the WLAN side, the IE is located at the Wireless Access Point (WAP). UE also has IE and DE to gather the downlink information for the Multi-P decisions and to execute the Multi-P transmission.

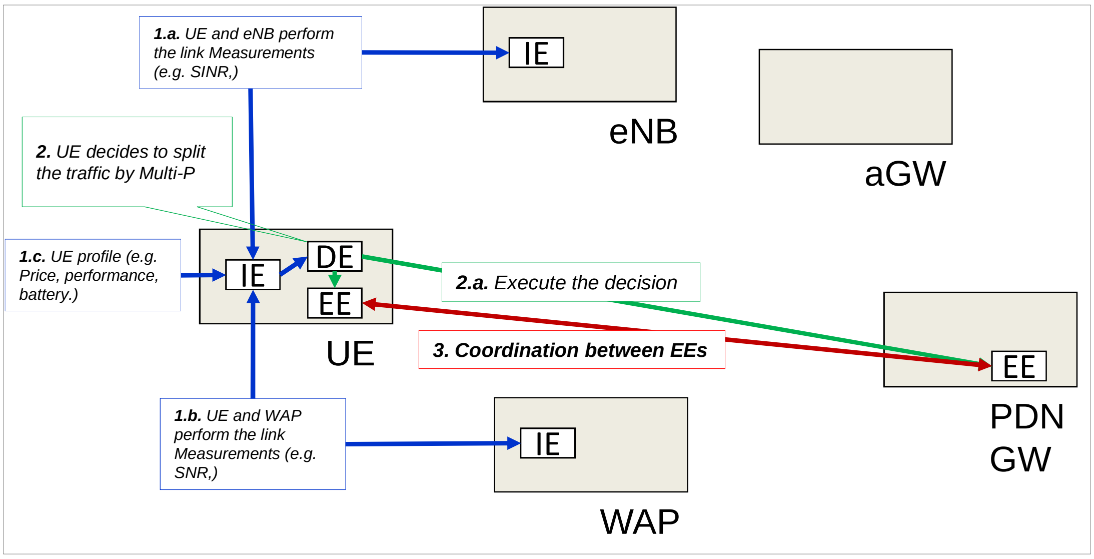

1.a. UE and eNB perform the link Measurements (e.g. SINR,)
IE
eNB
aGW
2. UE decides to split the traffic by Multi-P
1.c. UE profile (e.g. Price, performance, battery.)
IE
DE
EE
UE
2.a. Execute the decision
3. Coordination between EEs
EE
PDN GW
1.b. UE and WAP perform the link Measurements (e.g. SNR,)
IE
WAP

Figure 5.11: UE-controlled Multi-P decision

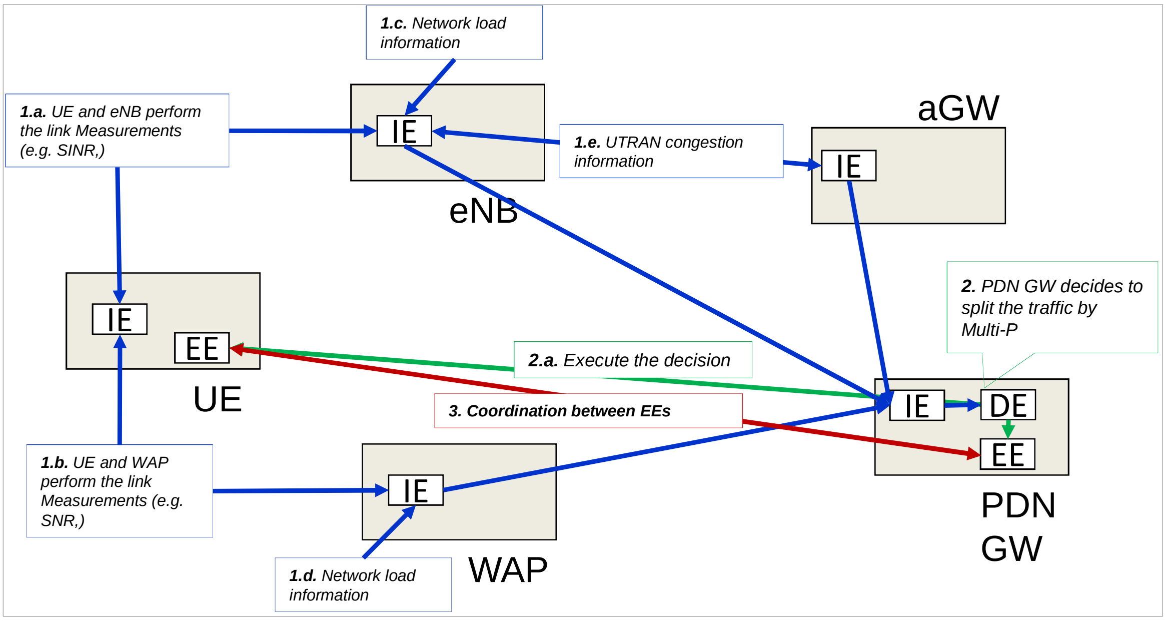

1.c. Network load information
1.a. UE and eNB perform the link Measurements (e.g. SINR,)
IE
eNB
1.e. UTRAN congestion information
aGW
IE
IE
EE
UE
2.a. Execute the decision
2. PDN GW decides to split the traffic by Multi-P
3. Coordination between EEs
IE
DE
EE
PDN GW
1.b. UE and WAP perform the link Measurements (e.g. SNR,)
IE
WAP
1.d. Network load information

Figure 5.12: Network-controlled Multi-P decision

Information gathering mechanisms are briefly described as follows:

- ***1.a.*** UE and eNB measure the LTE link quality (Signal to Interference Noise Ratio) on downlink and uplink, respectively.
- ***1.b.*** UE and WAP measure the WLAN link quality.
- ***1.c.*** eNB gathers network information such as traffic load and number of users in each cell
- ***1.d.*** WAP gathers the network load in its coverage.
- ***1.e.*** eNB and aGW measure the congestion status by using e.g. One Way Active Measuring Protocol (OWAMP) [RFC4656].

The information gathered by 1.a-1.e is to be sent to the IE in PDN GW (this could be done by using a proper proprietary signalling protocol). Based on the information collected in the IE, the DE in PDN GW makes the Multi-p decision according to the optimisation target(s). Basically, the criteria of the optimised Multi-P decision include:

- UE performance: data rates, delay, etc.
- Network performance: load balancing, coverage, admission and congestion control etc.

Finally, the decision is to be cooperatively executed by the EE in UE and PDN GW on the uplink and downlink respectively.

6 Traffic Steering Services, Mobility and Security

The aim of this chapter is to develop new network management paradigms that can interact with both legacy transport protocols and with the novel Advanced Connectivity Services presented in Chapter 5. This chapter proposes a set of mechanisms to manage and control these connectivity services in an efficient and scalable way, specifically investigating the following areas of research: Dynamic Mobility Management and Security & Mobility Framework (SMF).

Section 6.1 applies the OConS Decision Making mechanisms to manage mobility in a new dynamic perspective taking into account from the one side conditions of the multiple networks, and from the other side the different needs of users and devices. Several interdependent problems are addressed: access network selection for efficient handovers, distributed and dynamic mobility management, optimisation of mobility management in an operator's network and handover management in Multi-hop Wireless Networks.

Section 6.2 introduces the Security & Mobility Framework (SMF). This SMF is used to classify the variety of mobile and wireless mobility management and security solutions driven by the network heterogeneity and hide their complexity. The SMF provides significant support for the creation and evaluation of the security and mobility management concept. It will assist in both deciding on the appropriate security and mobility management solution and in demonstrating the outcomes of parameter variations.

6.1 Dynamic Mobility Management

The focus of OConS Management of Connectivity component is the Decision Making: i.e. how OConS can exploit all the information collected from the network entities, to make decisions on a number of different aspects of network connectivity and to enforce these decisions in the network devices and user equipment. Thus, among the expected innovations, the OConS provide the means to expose and activate on-the-fly the appropriate mobility services (triggering, decision, and execution) and, importantly, reacting in the most suitable way to changes in the networking conditions whenever needed; furthermore, the OConS mobility mechanisms and services support different selection and handover decision models for each applicative flow, as well as different execution models per-flow. Besides, it is worth highlighting that, taking advantage from the OConS functionalities, these mechanisms are not only considering the information affecting the access networks, but also the end-to-end paths, which could have a strong impact on the communication performances. The distributed decision process and the cooperation mechanisms which are brought about by the OConS framework will help us to provide the end-users with the most appropriate communication means.

The Flash Crowd scenario (see [SAIL-D.A.1]) gives a nice example of a situation in which OConS management of connectivity can be exploited: dynamic network conditions, multiple access networks, different user and device needs and multiple networks type (e.g. cellular, mesh networks). Hence, an important characteristic of the algorithms and mechanisms developed is that they will be able to cope with the high dynamicity of this environment.

After an overview of how this decision making process can be mapped onto the network functionalities, this section proposes a Dynamic Distributed Mobility Management approach. Then a UE-centric mobility management and a network-centric mobility management are taken in consideration. At the end a Multi-P Decision and Transmission for the Interconnection of 3GPP and non-3GPP Systems is described.

6.1.1 Access selection and decision algorithms

The starting assumption is that any end-users might be able to connect to a wide range of access alternatives, which in addition to using different technologies, might also belong to different operators. The process by means of which a connection is established depends on a combination of several different parameters, some of which are static (like policies, preferences, etc.), while others are of dynamic nature (e.g. requirements of the current service/application, particular load of the network, etc.). Furthermore, some of the information which might be used in order to make an appropriate decision is not directly accessible and, therefore, it should be acquired from nodes within the network.

Although this challenging framework embraces many potential issues (e.g. analysis of required overhead to get information and execute decisions, validity of the decisions taken, performance gains as compared with less complex mechanisms, etc.), our focus is put here on the decision process, as the most challenging and least addressed aspect of the framework. Thus, we need to deeply analyse how the various information elements may be combined in order to ensure an optimum performance.

Finally, the framework will also be used so as to explore the possibilities of establishing additional mechanisms, (especially on the network side). These include price selection (based e.g. on load information), cooperation strategies, etc. Figure 6.1 shows an overview of the proposed mechanism mapped on the OConS architecture.

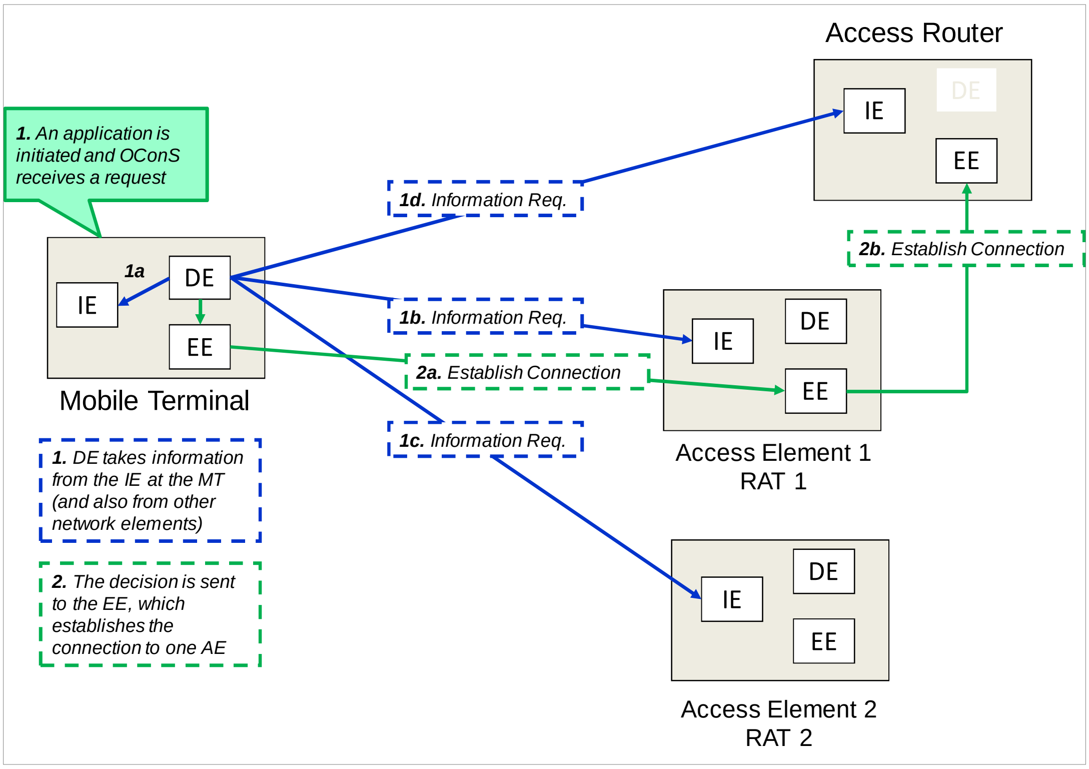

Access Router
IE
DE
EE
1. An application is initiated and OConS receives a request
1d. Information Req.
2b. Establish Connection
1a
IE
DE
EE
1b. Information Req.
IE
DE
EE
2a. Establish Connection
Mobile Terminal
1c. Information Req.
Access Element 1
RAT 1
1. DE takes information from the IE at the MT (and also from other network elements)
IE
DE
EE
2. The decision is sent to the EE, which establishes the connection to one AE
Access Element 2
RAT 2

Figure 6.1: Dynamic mobility management mechanism mapped onto the OConS architecture

The main goal of this investigation is to analyse the potential combination of various parameters which may be used so as to decide the access element to be connected to; in this sense, there should not be many limitations on the information needed for the decision process. In particular, the following information should be considered:

1) Resources offered by the access elements (capacity) and required by the services. This parameter heavily depends on the involved technology, e.g. the number of subcarriers for Orthogonal Frequency Division Multiple Access (OFDMA) based technologies, codes in Code Division Multiple Access (CDMA) systems, time-slots, etc. Therefore, we are using an abstraction, and the resources will be characterised in terms of discrete variables.

2) Policies for the end-user and networks. These involve pricing strategies, white and black lists for specific operators, cooperation mechanisms between networks (for instance, to perform load balancing).
3) Requirements for the services. The straightforward requirement is the resources needed, however we may also consider security, real time needs, etc.
4) Particular characteristics of the scenario. One clear example is the link quality (also abstracted to enable fair comparisons) that the end-user terminal has with all the potential access elements. Other elements to be considered reflect the characteristics of the network up to the destination, which may necessitate gathering the information from various nodes/entities within the network.

We start with the assumption that the decision to be acted on is not necessarily restricted to the end-user equipment, but it can be also distributed to the network elements.

The basic scenario comprises a highly heterogeneous deployment, with a Mobile Terminal (MT) equipped with various interfaces able to access a number of access elements (Access Point/Access Router). These employ different technologies and are operated by various entities.

The OConS entity at the MT receives a request from an application/service, with a set of certain requirements; this request is transferred to the DE, which uses the available information so as to make a decision on the Radio Access Technology (RAT) to handle such application or service. This information might be straightforwardly available (the IE can be periodically populated based on certain initial configuration – for instance, the quality perceived on the links with the available RATs) or might be retrieved on-demand (there might be an issue with the delays in this case).

Once the DE takes the decision, this is communicated to the EE, which executes the required actions so as to initiate the flow through the selected RAT. This triggers a set of message exchanges between the mobility-related entities.

Furthermore, additional messages may be exchanged between the entities to ensure a proper monitoring of the quality perceived for the application and to be able to take appropriate measures when these are needed.

Note that, for the sake of clarity, some of these messages are not reflected in Figure 6.1.

6.1.2 Dynamic Distributed Mobility Management

Based on the decision making approach introduced above, OConS provides a highly scalable mobility framework considering mobility decision and execution functions in a distributed and flow based approach. Specifically, the following mechanisms are considered: flow scheduling/path-selection, per-flow handover-decisions, and per-flow anchor selection and activation.

In our view, the "optimal" balance between host-centric and network-centric decision points can be dynamically obtained for each application flow and depending on a given communication context (i.e., resources, requirements, policies). Likewise, for the execution part, we are minimising the maintenance of unnecessary traffic encapsulation, mobility anchors and mobility-related context. Figure 6.2 further exemplifies this mechanism.

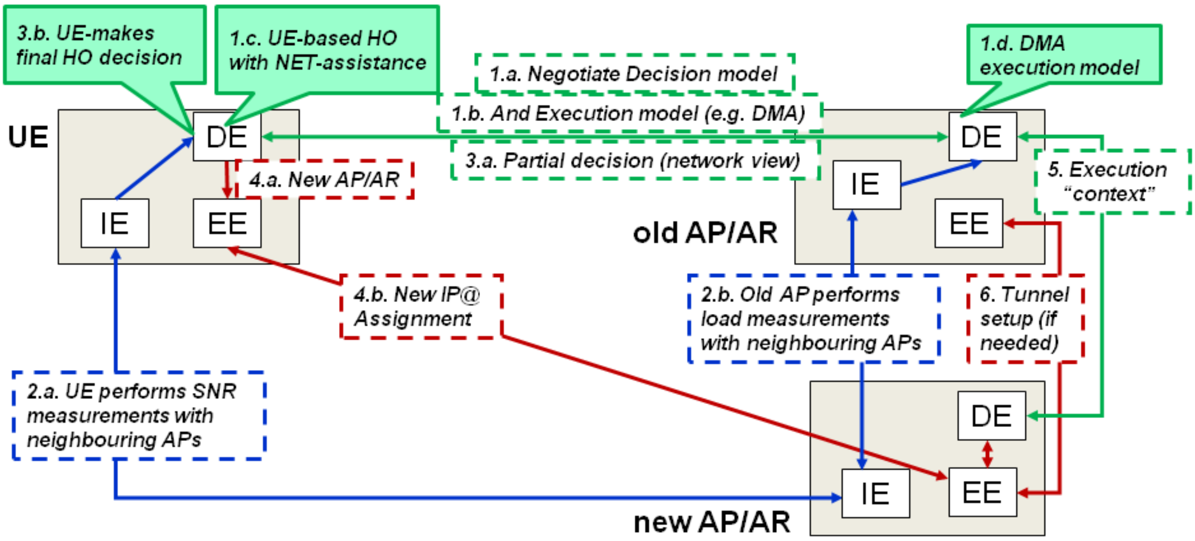

3.b. UE-makes final HO decision
1.c. UE-based HO with NET-assistance
1.a. Negotiate Decision model
1.b. And Execution model (e.g. DMA)
1.d. DMA execution model
UE
DE
IE
EE
4.a. New AP/AR
3.a. Partial decision (network view)
5. Execution "context"
old AP/AR
DE
IE
EE
4.b. New IP@ Assignment
2.b. Old AP performs load measurements with neighbouring APs
6. Tunnel setup (if needed)
2.a. UE performs SNR measurements with neighbouring APs
DE
IE
EE
new AP/AR

Figure 6.2: Dynamic Distributed Mobility Management mapped onto the OConS architecture

The information needed by the decision process is as follows:

- Resources and corresponding characteristics on the terminal:
  - List of available interfaces and access networks for each interface
  - Neighbours APs/BSs and their types (802.11b/g/n, UMTS/LTE)
  - Current measurements per interface (mainly throughput and delay)
- Resources and corresponding characteristics on the network side:
  - Current load for an AP/BS (the radio load, but also the number of connected terminals and the CPU load) and their throughput/bandwidth towards the ARs
  - Current load for an Access Router (number of flows handled, per-flow context stored, CPU load) and their throughput/bandwidth towards their peers and higher routers in the topology
- Applications and other requirements: per-flow application minimum QoS (e.g. minimum throughput needed)
- Policies/Preferences:
  - User preferred interface and access-network/operator for an application flow (e.g., depending on throughput/bandwidth and price)
  - Operator preferred access-network for a user and an application flow

This mechanism is composed from the following phases/steps for each application flow:

Negotiation:

- **1.a/c** Negotiate the Decision model, e.g., UE-based handover with Network-assistance
- **1.b/d** Negotiate the Execution model, e.g. Dynamic Mobility Anchoring

Information Collection:

- **2.a** UE performs measurements with neighbouring APs/BSs
- **2.b** Old/Current AP/BS performs load measurements with neighbouring APs/BSs

Handover Decision Making:

- **3.a** Partial decision based on information available on the network side
- **3.b** UE-makes final decision combining its information with the one from the network

Handover Execution and Enforcement

- **4.a** The terminal switches this flow to the new AP/AR (and possibly a new interface)
- **4.b** If necessary (e.g., for a new flow) a new IP address is assigned

Handover Execution Optimisation

- **5.** Execution “context” is updated between the involved APs/ARs
- **6.** A tunnel is set up only if needed (i.e., depending of a given application flow)

The detailed messages/primitives are left for future work; however we can already suggest some of the possible solutions to use as a starting point:

- Between the IEs and the DEs, we can e.g., adapt the IEEE 802.21 and the IETF Common Control and Measurement Plane (ccamp)
- Between the DEs and the EEs, we may use an enriched ANDSF (i.e., OMA-DM) and an enriched Access Node Control Protocol (ANCP)
- And among the EEs, a modified version of MIPv6/PMIPv6 is to be developed.

6.1.3 Mobile-driven Network Selection and Flow Scheduling

A mobile UE is usually equipped with multiple network interfaces allowing it to connect to networks based on different technologies. Though most commonly distributed systems select one network based on arbitrary criteria (e.g. “favour Wi-Fi over 3G”), using multiple interfaces at once depending on the demand may prove to be a better solution.

How to define “better” in this context is a problem of its own, but considering UE are terminals for interaction with a user, we consider the user's perceived quality (quality of experience, QoE) of their application to be a relevant metric. In addition, the battery lifetime of the device, and the price incurred by using the connected networks can also be of importance to the user.

Therefore, UE-centric mobility management can be seen as the appropriate selection of a set of networks to connect to, given the requirements of the currently running application to provide an acceptable quality to the user, while maintaining acceptable power consumption and usage costs. An emerging problem also concerns the distribution of application flows over the thus selected networks uplinks.

We now describe our mechanism according to the OConS framework. Hence, the UE is in charge of collecting information from its own subsystems, and the networks in range, in order to estimate the offer and demand in terms of network quality of service (QoS), energy consumption and price. It then decides which networks should be associated to each interface, and how application flows should be distributed to optimise the conflicting goals of high QoE for each application but low battery consumption and monetary cost.

The user may also have preferences driving this decision process such as an important application, or a very long battery lifetime requirement.

As the decision process may be computationally expensive, it is envisioned that decision brokers could provide processing power so the UE delegates this part of the process to an external entity.

Finally, the UE establishes the uplinks, and interacts with legacy mobility protocols such as MIPv6 to update address and flow bindings to the new configuration.

Figure 6.3 provides an example of the mechanism. To the outside world, the UE only implements one of each IM, EE and possibly DE. However, the figure includes a similar modelling of the internal subsystems of the UE as they are ultimately the components which either generate or consume the information, while the UE's entry point only performs a routing function towards the relevant entities.

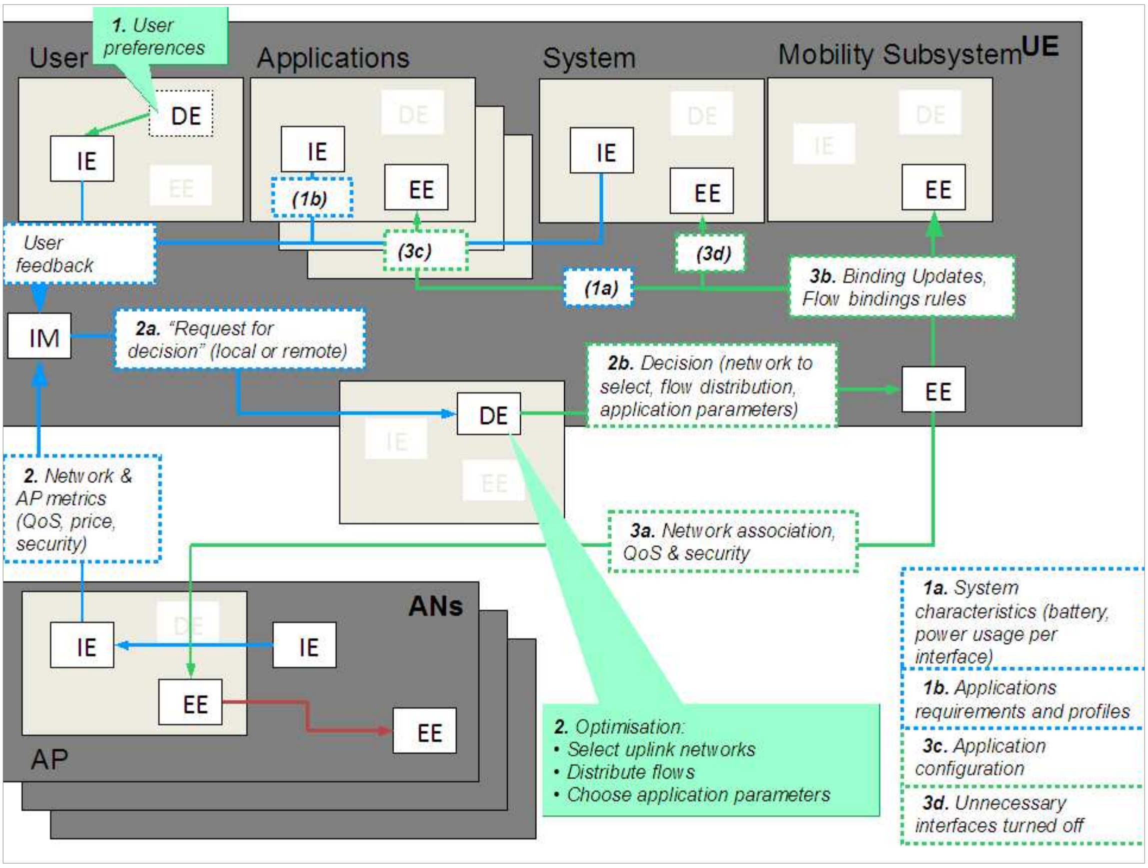

1. User preferences
User
Applications
System
Mobility Subsystem
UE
DE
IE
EE
(1b)
(3c)
(3d)
(1a)
User feedback
IM
2a. "Request for decision" (local or remote)
3b. Binding Updates, Flow bindings rules
2b. Decision (network to select, flow distribution, application parameters)
2. Network & AP metrics (QoS, price, security)
3a. Network association, QoS & security
ANs
AP
1a. System characteristics (battery, power usage per interface)
1b. Applications requirements and profiles
3c. Application configuration
3d. Unnecessary interfaces turned off
2. Optimisation:
• Select uplink networks
• Distribute flows
• Choose application parameters

Figure 6.3: Mobile-driven Network Selection and Flow Scheduling mapped onto the OConS architecture

The mechanism can be separated into three main phases:

1. *Information collection by the UE's IE:* acquisition of local context and available network characteristics,
2. *Decision request and reply:* the DE, be it UE-local or remote, is given the aggregated information, and provides an adequate configuration for the subsystems, and
3. *Application of the decision:* the relevant parts of the decision are sent to the UE and network's EE to perform the required configuration tasks.

In this simplified model, it is assumed that information about the Access Network (AN) providers is obtained through the closest Access Points (APs) of said network. However, there may be cases where the information is not readily available, for technical or political reasons. An external collaborative database could provide a fall-back solution by implementing an IM to provide aggregated measurements by other users of an AN/AP, to transparently (but possibly less accurate) address this issue.

The information needed by the decision process is listed in Table 6-1. Then, Table 6-2 details the expected outputs from the decision mechanism.

Table 6-1: Information needed for mobile-driven network selection and flow scheduling

Source	Information	Details
User	Preferences/ Priorities	Weighting of the decision criteria in reaction to their relative importance and the observed performance
Application	Quality profile	Mapping of available QoS and possible application parameters to the achievable QoE
System (including sensors)	Current state	Battery level Movement: speed, direction
	Typical metrics	Expected battery usage for each interface
AN/AP (or collaborative database)	Achievable QoS	Capacity (up/down), delay, losses; possibly more finely detailed depending on the destination network
	Pricing	Access/use pricing policies (usually unavailable from 3G networks as already know by the user, this information may then come from a local database on the UE)

Table 6-2: Expected outputs from selection and scheduling decision mechanism

Information	Destination
Network associations	UE's connectivity system, and potentially AN/AP to prepare for the association; all non-associated interface may be turned off to save battery
Flow distribution	Mobility subsystem; to be updated with the anchor point (*e.g.* home agent) at the same time as the locators get updated with successful network association
Application parameters	Application quality profiles; may include adjustable application parameters; the decision system evaluates the QoE based on the achievable QoS in light of those parameters which then have to be fed back to the application for appropriate adaptation

6.1.4 Centralised optimisation of mobility management with a self-adapting network

One fundamental capability of OConS is to enable the optimisation of mobile networks operations, leveraging the information they collected about nodes and users and taking specific decisions on the treatment given to the data traffic, on a per-user basis.

In fact, a large amount of data on the performance and load of the network are available on several entities (e.g. access and core routers, radio elements, authentication servers etc.). Moreover, the network could also know the user behaviour, using information coming from different sources: from the control plane, it can infer user’s “mobility patterns” (e.g. handovers performed by the UE); from other service devices, like Deep Packet Inspection (DPI) engines (which are increasingly being deployed in fixed/mobile networks to inspect data traffic), it can e.g. know if a user is data- or voice- centric. The network has also the contractual data for the user (i.e. tariff the user is paying).

Based on all the available information, OConS can optimise many aspects of mobility management and mobile traffic routing: access network selection, interface selection on the UE, mobility anchor point location, applications flow routing through multiple interfaces and a number of mobility parameters. Moreover, OConS can make the network *self-adapting*, automatically changing its parameters, in function of the above-mentioned information, reacting to the dynamic conditions of network load and usage.

The following paragraph contains two examples, in which the OConS framework is used to implement the concept of self-adapting network. OConS could have two "modes" of operation:

- ***Reactive:*** When some high load condition occurs (on one or multiple nodes), OConS Decision Entities can react to re-distribute the load on the less congested nodes, in a transparent way for the connected users (e.g., see OConS use-cases from [SAIL-D.A.1]).
- ***Proactive:*** A simple scenario of proactive OConS is as follows: User A and User B both own multi-mode terminals 4G/Wi-Fi; User A generates a lot of data traffic, but only a few voice calls as he usually moves from home to work and from work back to home; User B makes a lot of circuit voice and VoIP calls being a mobile worker. User A does not need specific QoS guarantees, so the network (i.e. OConS Decision Entities) decides to allocate a mobility anchor point on Wi-Fi Access Router using on Dynamic and Distributed Mobility Management (DDMM) scheme. Optionally, the network could even decide that for User A, no mobility anchor is needed, because he can tolerate a session disruption. Figure 6.4 describes the architecture where the mobility management is accomplished with such centralised approach (there is only one centralised "Decision Manager"), and no centralised anchor point is selected for the UE; instead a local AR is chosen.

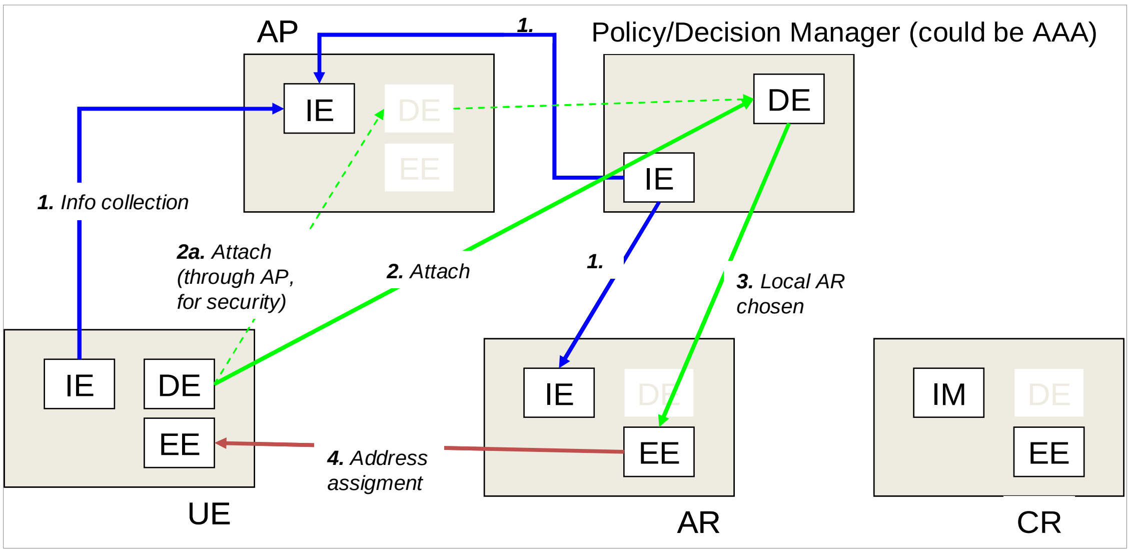


Figure 6.4: Centralised optimisation of mobility management mapped onto the OConS architecture

As User B needs better QoS and a more reactive Mobility Management, the network decides to allocate an anchor point on a Centralised Router. Figure 6.5 describes the anchor point selection in this case: the Decision Manager again takes a centralised decision and a centralised anchor point is chosen.

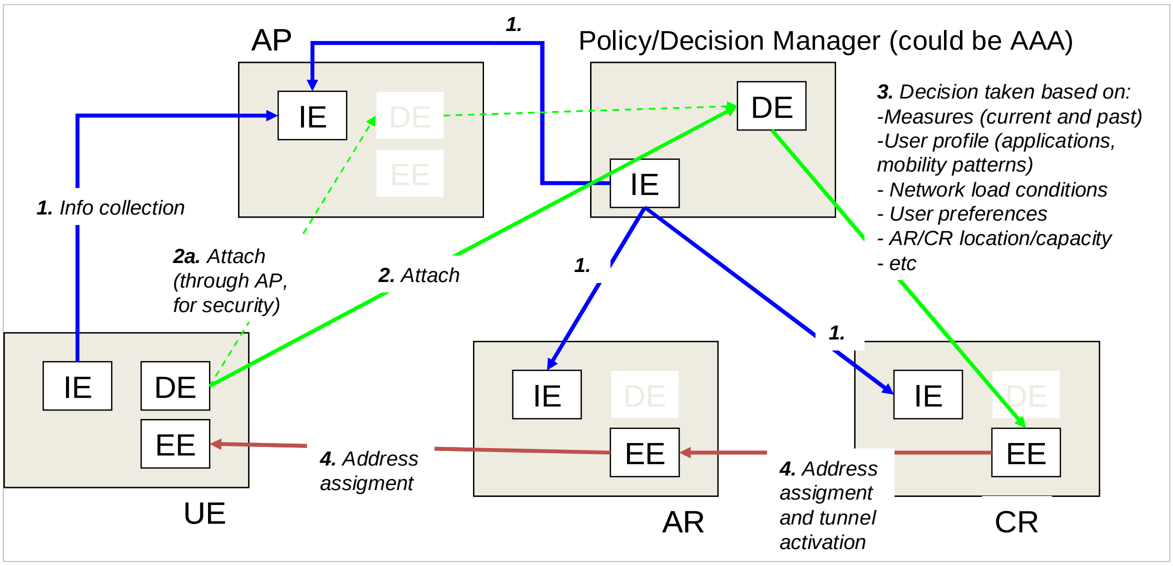

AP
1.
Policy/Decision Manager (could be AAA)
IE
DE
EE
DE
IE
3. Decision taken based on:
-Measures (current and past)
-User profile (applications, mobility patterns)
- Network load conditions
- User preferences
- AR/CR location/capacity
- etc
1. Info collection
2a. Attach (through AP, for security)
2. Attach
1.
1.
IE
DE
EE
IE
DE
EE
IE
DE
EE
4. Address assigment
4. Address assigment and tunnel activation
UE
AR
CR

Figure 6.5: Anchor point selection on a Centralised Router

More complex scenarios are possible, e.g. when multiple Wi-Fi networks are available and the device must choose among these. There are a number of open questions related to such scenarios, e.g. UE can decide how to route the different flows on the available network, but the network needs a mechanism to enforce this decision. Furthermore, additional issues to be investigated include the deployment of policies which include user preferences, needed for decision making.

Table 6-3 describes the information elements needed for the mechanisms explained above.

Table 6-3: Information needed for optimisation of mobility and anchor selection

Resources	Terminal	Radio and Network capabilities Active interface list Location information (e.g. GPS, accelerometer) Measurements (on a interface-basis, includes also neighbour cells/APs) Mobility management (MM) context
	Network	Load statistics of network nodes (CPU load, link occupation, current load in terms of throughput, number of users) Topological information for AR/CRs Mobility events (e.g. change AP) Location information (e.g. last known cell/AP/tracking area of a user) MM context User traffic profile (i.e. applications used and usage patterns)
Requirements	User	Best QoE for heavily used applications
	Operator	Optimal usage of network resources Data load distribution across available resources

Policies	User/Client	User preferences (preferred network, preferred interface, typical usage –data centric, voice centric, nomadic etc) User application preferences (i.e. where to steer an application flow, when multiple interfaces are available)
	Operator	Operator preferences (e.g. preferred interface) Anchor point selection policy (e.g. for voice centric users → centralised router) Mobility parameters setting

6.1.5 Efficient Handover in Multi-hop Wireless Networks

In a multi-hop wireless network, continuous connectivity is provided, among other things, by handover mechanisms, which maintain user session during the transition from one Base Station (BS) to another. In a multi-hop wireless network, a wireless channel is a scarce shared resource, which should be used efficiently. Hence, the handover mechanism for such a network, should not only provide uninterrupted service, but also keep in mind minimal consumption of extra wireless bandwidth.

Efficient handover in mobile networks usually considers short handover delays, small packet loss, or small buffer sizes. In this research, we would like to focus on efficient use of network resources that are allocated for the handover. This is extremely important since the wireless links are used for both access and backbone connectivity.

Several handover techniques have been developed. These techniques can be classified as soft handovers and hard handovers.

- In soft handover, the old and the new BSs transmit the same data simultaneously, in order to minimise handover delays and losses. In this make-before-break approach, a connection to the new BS is established before the mobile node is disconnected from the old one. This technique is mainly suitable for voice and other delay/loss-sensitive applications. However, for data traffic, such as web browsing, soft handover leads to inefficient bandwidth utilisation, and thus, it is not the preferred approach for delay/loss-tolerant applications.
- In hard handover, a connection with the old BS is ended before the mobile node receives data from the new BS. This break-before-make approach is more bandwidth-efficient than soft handover, but it causes longer delay and higher losses. As a result, it is more appropriate for traffic of non real-time applications.

In our research, we focus on hard handovers, and evaluate what is the best approach for the old BS: should it silently drop the packets that are destined to a mobile host that is no longer connected to it, or should it forward them to the new BS. Clearly, forwarding such packets avoids TCP wasteful and time-consuming recovery, at the expense of extra bandwidth used, bandwidth that might not be available. Therefore, forwarding should be selectively conducted, only when bandwidth is available, in order to maximise throughput.

Our research models this problem as an optimisation problem, in order to better understand the tradeoffs between resource utilisation and performance. In Figure 6.6 is depicted the mapping into the OConS architecture of the proposed handover mechanism.

The information required for the scheme, at a base station level, is summarised in Table 6-4.

The scenario depicted involves a Mobile Host (MH) that is leaving BS1, and joining BS2. Due to wireless channel conditions, MH makes the decision to switch to BS2 (its DE entity). It then notifies BS1 that it leaves (step 1), and BS2 that it joins (step 2). BS1 then notifies the First Common Parent node (FCP), to start rerouting packets destined to MH, towards BS2 (step 3).

At this point, BS1 should determine if forwarding any pending packets to BS2 is beneficial, considering the cost and the benefit of such forwarding. This algorithm is the core of our research, and it is running on BS1 DE. For an educated decision, BS1 relies on the following info to be provided by MH (step 4):

- The flow involves real-time or not real-time application; for streaming of real-time, forwarding should be considered; for Data services, it is better to let TCP timeout mechanism to recover (i.e. BS1 drops all the packets). This information is available at the MH, or at the sender, but not at intermediate nodes
- Number of pending packets which are on the way. Our analysis shows that at the worst-case scenario, the number of pending packets equals the Congestion Window (CW) window size. BS1 does not know the last-used CW value, and thus needs to query MH (same step 4)

BS1 then continues the evaluation of the forwarding process. Relying on nearby topology information and load which is constantly collected, it resolves the link towards BS2 (towards which it will be forwarding the pending packets), the current load on these links and the capacity of those links (step 5). It then computes a profit function that guides BS1 to silently drop the pending packets or to forward them (and also decides on the minimum number of packets to be discarded or forwarded) (step 6).

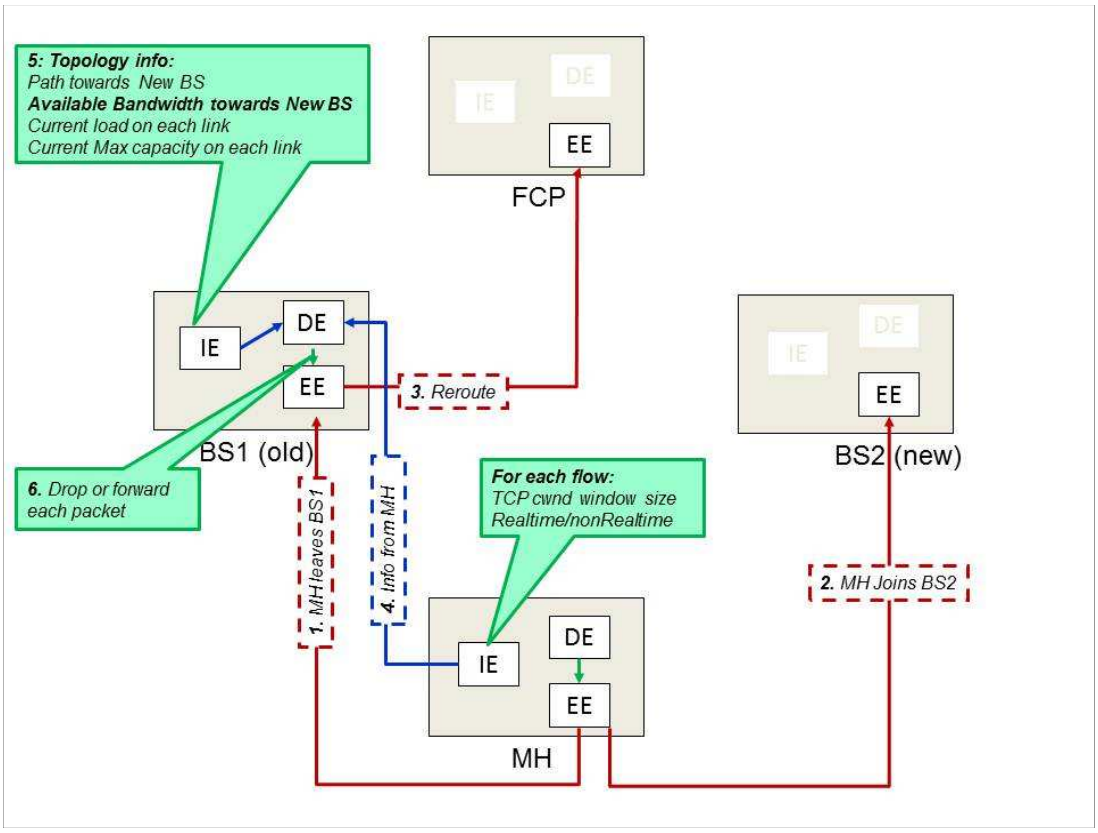


Figure 6.6: Handover in Multi-hop Wireless Networks: The handover mechanism mapped onto the OConS architecture

Table 6-4: Information needed for handover decision in Multi-hop Wireless Networks

Resources	Networks	• Current topology • Current load on each link • Current maximum capacity of each link
	Context	• For each active flow with pending handover: Type of payload: data or real-time (voice/video) Current TCP CW window size
Requirements	User	• Uninterrupted service during handover, including real-time applications (voice, video) • Maintain maximum possible throughput for data services (maximum TCP window size possible, enabled by minimal packet loss / out-of-order packet delivery
Policies/Preferences	Operator	• Maximum throughput: most efficient use of bandwidth

6.2 Security and Mobility Framework

The aim of the Security & Mobility Framework (SMF) is to figure out a comprehensive and structured line-up of options in security and mobility management and their side effects to support selections of the right options. The SMF describes and classifies the various processes, entities, elements, mechanisms and information used in mobility management, hereinafter referred to as ‘items’. Furthermore, it assesses the pros and cons of these items in conjunction with potential side effects of jointly-implemented items.

The SMF is focused on the structuring and encompassing of mobility management items in general. It includes in its scope the general OConS decision mechanisms, but it is not limited to decisions and selections mechanisms from Section 6.1. Accordingly, in the SMF’s scope we describe and classify the relevant security and mobility management items applicable to different use-cases in a structured way. Thus, the SMF addresses mobility management items with a broader scope than the specific mobility management work, including e.g. multi-protocol and RAT heterogeneity.

6.2.1 Motivation and Intended use of SMF

The challenge in OConS is to ensure end-to-end service continuity to and from the mobile device, via interconnected heterogeneous wireless communications in order to provide the required QoE. When nodes in different networks communicate, or the network conditions change, interworking of mobility management and security mechanisms can be used to overcome the consequences of heterogeneity. The variety of mobility management and security solutions at this juncture creates a highly complex set of alternatives and interactions. The SMF counters this complexity by limiting the technical depth to a level that is suitable to classify security and mobility management items in general; both for describing and evaluating the items and for facilitating a substantiated selection of the alternatives.

The SMF provides significant support for the creation and evaluation of the security and mobility management concept. We note that the SMF will first assist in deciding on the appropriate security and mobility management solution and, second, in demonstrating the consequences of parameter variations.

The intention is for the SMF to be used as an analytical tool that allows comparing different security and mobility management concepts. Since mobility management procedures are influenced by a variety of parameters, an optimised management can make the difference to the quality of a system using mobility functionalities. SMF is based upon an almost comprehensive list of parameters that have an influence on the OConS mobility management.

Being complete, the SMF helps a user of this analytical tool to decide for the appropriate mobility management solution under a set of given values for the parameters and shows the influence of parameter variation on the mobility management. Additionally, the SMF allows the comparison with alternative solutions that may exist.

6.2.2 Description of approach and its rationale

A specification of secure mobility management aspects for OConS, including realisation variations and alternatives, is very extensive. Additionally, OConS is influenced from different directions that would make a framework specification extremely complex to understand. Thus, the SMF uses the framework of abstraction from the Reference Model of Open Distributed Processing (RM-ODP) standard as a descriptive structuring instrument. The five generic and complementary viewpoints of RM-ODP on OConS and its environment will be used to structure the above-mentioned aspects of the mobility management options.

The RM-ODP has been jointly developed by the International Organisation for Standardisation (ISO), the International Electrotechnical Commission (IEC) and the ITU Telecommunication Sector (ITU-T). The RM-ODP family of recommendations and international standards (namely ITU-T Rec. X.901 | ISO/IEC 10746-1 to ITU-T Rec. X.904 | ISO/IEC 10746-4, please see [X.901-X.904]) defines essential concepts necessary to specify open distributed processing (ODP) systems from five viewpoints and provides a well-developed framework for structuring of specifications for large-scale, distributed systems. It supports distribution, interworking, platform and technology independence, and portability, together with an enterprise architecture framework for the specification of ODP systems. The RM-ODP is based on precise concepts derived from current distributed processing developments and, as far as possible, on the use of formal description techniques for specification of the architecture.

The viewpoints that are used within the SMF are aligned to be able to specify any aspect of mobility management in a system perspective. The five viewpoints enable separation of concerns, which means that each viewpoint can focus on the details which are of concern to it, and ignore the details of the other viewpoints. This is the basis of abstraction in system design. On the other hand, the viewpoints are used to describe the same system, and therefore cannot include descriptions that are contradictory; the viewpoint descriptions have to be consistent, as they are describing different aspects of the same 'item' used throughout mobility management processes, or the same 'item' in different ways.

6.2.3 Definition of Viewpoints and their interpretation within the SMF

Figure 6.7 provides a high-level summary of the viewpoints that are used within the SMF.

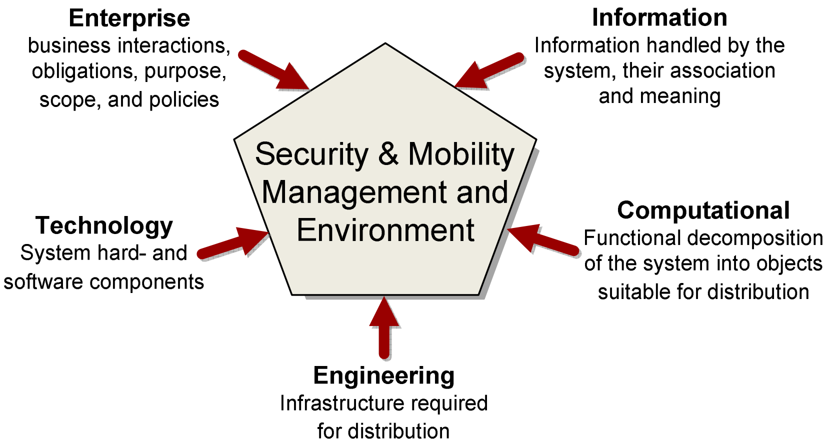

Enterprise
business interactions, obligations, purpose, scope, and policies
Information
Information handled by the system, their association and meaning
Security & Mobility Management and Environment
Technology
System hard- and software components
Computational
Functional decomposition of the system into objects suitable for distribution
Engineering
Infrastructure required for distribution

Figure 6.7: System for security & mobility management system and its environment: high level summary of the viewpoints

In the following sections we will define the extent and delimitation of each of the five viewpoints in reference to mobility management aspects.

6.2.3.1 Enterprise viewpoint

The enterprise viewpoint focuses on business interactions and associations, obligations, purpose, scope, and policies of a mobility management supporting system. It provides the context and overall environment within which the system will be built, and therefore constraints, and obligations that must apply in all other viewpoints.

The term 'enterprise' comprises any entity that is involved in mobility management processes. The enterprise viewpoint can be used to specify entities such as involved parties, systems, information and identity brokers, providers, and infrastructural elements etc. that are necessarily involved within mobility management processes and that provide essential functions within these processes. The enterprise viewpoint observes actors, roles, aims and intentions of these entities within the system, and defines how and why those actors behave. These actors will be defined by basic objects that will be in the system. Activities of the actors are also identified.

Enterprise viewpoint identifies and describes policies and processes according to business benefits and obligations for the different entities. Such policies might be set internally by the entities themselves, but policies may also be defined externally to the enterprise, which are constraints such as government laws or rules set by regulatory bodies, safety rules, industry agreed codes of practice.

Communities and Federations are also identified. Communities are groups of objects or actors which together perform some objective. Federations include several enterprises that have their own enterprise specifications and perform jointly to provide services to their common customers.

6.2.3.2 Information viewpoint

The information viewpoint defines and terminologically describes 'items' used throughout such mobility processes. The categories for these 'items' are:

- Information Objects – Definition of information or data objects (e.g. identities, credentials, measurement, and trigger values) that are used in security and mobility management as well as objects from the enterprise viewpoint description. These include enterprise viewpoint actors, process and technology elements.
- Associations – Definition and description of relationships between information objects.
- Procedures – Terminological definition of processes in security and mobility management concepts and their parameters.
- Impact and outcome – Definition and description of result categories for impact and outcome of associations and procedures.

6.2.3.3 Computational viewpoint

The computational viewpoint provides functional decomposition of the overall mobility management supporting system into objects that interact at their interworking or programmatic interfaces. This decomposition will provide natural lines along which a system may be partitioned related to mobility management tasks. The computational viewpoint is concerned with the separation in components, their interfaces and the interaction patterns between the components. The computational viewpoint specification includes:

- Boundary conditions, raw process descriptions and involved entities (e.g. user, handset, network node, router, backend, etc.) of mobility management processes,
- Scheme views on processes and their involved objects in terms of activity diagrams, and

- Descriptions of liaisons that can be created between objects for interactions that occur during mobility management processes.

6.2.3.4 Engineering viewpoint

The engineering viewpoint focuses on the detailed deployment aspects of a system supporting mobility management. In contrast to the computational viewpoint, which merely enables distribution implicitly, distribution is explicit in the engineering viewpoint.

The engineering viewpoint is a technical view on how processes are implemented in technology, hardware, interfaces, etc. and how technology, hardware and interfaces support these processes. It focuses on the mechanisms and functions required to support distributed interaction between objects in the system by describing the capabilities and characteristics of mobility 'items'. An engineering specification is expressed in terms of (1) a configuration of engineering objects, structured as clusters, capsule and nodes, (2) the activities that occur within those engineering objects, and (3) the interactions of those engineering objects.

An engineering specification that corresponds to a computational specification defines engineering structures that provide transparency of the distribution (e.g. in terms of access, location, etc.) in the computational specification.

The mobility management system is visible from the engineering viewpoint in terms of:

- Specifications of the behaviour of the engineering objects like the end-user device including detailed specification of protocols (message sequence charts), specification of the precise representation of the abstract data types identified in the computational description, and QoS requirements,
- Description of characteristics and capabilities of mobility management related 'items', and
- Constraints between the behaviour at the reference points that reflect the specified computational activities.

6.2.3.5 Technology viewpoint

The technology viewpoint is concerned with the choice of technology in that system. This entails the choices of specific hardware and software components compliant with options and requirements. The technology viewpoint focuses on specifics for particular implementations and the use of standards of the mobility management system. As there may be many sets of technology chosen to implement such management processes, either at one time, or as future technology becomes available, there could be many different technology viewpoint specifications for any one system.

In a mobility management system specification in which the choice of technology is left open, the technology specification does not provide full details of specific technology resources. A technology specification expresses how the specifications for such a system is implemented, and identifies specifications for technology that is relevant to the construction of the systems.

The mobility management system is visible from the technology viewpoint in terms of the technologies and available products for implementation and provision of the underlying infrastructure, the processes and the distinct elements. It will include the description of the specification of radio access technologies and protocols used for mobility management and security.

7 Resource Management and Enhancements

One of the most relevant characteristic of the OConS architectural framework is its flexibility to cope with the particular requirements of various processes, mechanisms and techniques, either the present or the forthcoming ones. In this Chapter we introduce a number of advanced resource management mechanisms, which we aim at integrating in the OConS framework.

We start with one of the most promising advancements of wireless technologies: the cognitive use of the spectrum. Although cognitive radio was originally proposed by Mitola already in 1999, the recent appearance of infra-used spectrum bands (i.e. television white spaces) has gathered the attention from the scientific community; we illustrate how the OConS framework may be used so as to streamline the process of spectrum sensing, which may lead to heavy improvements in terms of efficiency

Then, we present an algorithm to optimally allocate the capacity devoted to channel quality indicator messages, which are deemed as fundamental for a proper adjusting of the modulation and coding scheme of OFDMA systems, which are nowadays present in the most relevant wireless access technologies (for instance, LTE).

On the physical layer, the last technology we use to challenge the OConS framework is the use of virtualised resources. Virtualisation has been traditionally attributed to computing resources (for instance, storage capacity), but it might be also become really beneficial for the communication realm; in this sense, we illustrate how the OConS architecture may help to appropriately deal with virtualised connectivity within heterogeneous wireless access environments.

Afterwards, we focus on wireless mesh networks and DTNs, which have recently appeared as novel extensions of legacy (usually based on single wireless hop) access alternatives; in particular, we analyse the problems of channel assignment and energy awareness, and we explore the benefits which might be brought about by using context information so as to enhance routing and forwarding mechanism over DTNs.

Finally, we also use the OConS framework for two particular techniques which improve legacy core network connectivity services, in particular routing and overlaying for data-centre interconnections. First, we show how the OConS approach can be used to overcome some of the shortcomings of current routing strategies between Autonomous Systems; the BGP, the most widespread solution, employs policy-based routing, but it does not consider other parameters, which can be included in the decision process supported by OConS. Last, we will discuss how OConS can be used so as to integrate some of the currently developing solutions to facilitate data-centre interconnections by means of overlays.

7.1 Spectrum sensing, Physical channels, and Virtualisation Techniques

7.1.1 Cognitive Radio Systems through Spectrum Sensing Techniques

We aim at exploring spectrum sensing techniques in order to derive useful information regarding wireless channel occupation. We will implement and evaluate techniques and algorithms to detect and identify wireless channel activity and provide this kind of information to any interested decision mechanism.

The radio spectrum is suboptimally used in many areas. Cognitive radio networks are considered as a promising technology to overcome the spectrum scarcity problem. In order to improve the efficiency of wireless networks, the scientific community has made an effort to develop and exploit the opportunistic access to the physical environment through Cognitive Radio technologies.

The “smart” spectrum access property will allow secondary market users to access dynamically spectrum channels if primary user communications are not degraded during this process. In this sense, it is extremely important to shield the primary users from any interference that might be generated by secondary users accessing to occupied spectrum channels. The reliability when determining the presence or absence of primary users is thus essential.

The general scheme proposed for cognitive radio and opportunistic radio access is depicted in Figure 7.1. The components involved across the scenario are, on the one hand, the Network Nodes (up to N nodes) and, on the other hand, the Manager Node. It must be noted that the roles adopted by these elements are not static, since any node could act as Manager Node if required by a particular context, service, or architecture.

The Network Nodes have the functionality of Spectrum Monitoring or Spectrum Sensing capability, which corresponds to the IE as defined in the OConS architecture. According to the capacity of each node, the sensing process can be applied to a single channel *K*, or to a wideband signal composed of several radio channels (Wideband Spectrum Sensing). Let us assume, for the sake of simplicity, that each node only monitors a single channel *K*, provided that it does not affect the general scheme nor the message-exchange procedure. By applying signal power estimation techniques such as energy detection or waveform-based power estimation, the signal power level *S* is obtained.

After nodes monitor and estimate the signal power level *S*, the DE of each node will compare this estimated level with a certain pre-established threshold, to decide whether channel *K* is occupied or not. This strategy is based on a pure hard decision of each network node. Nevertheless, and due to the inherent unreliability of the estimation problem with low Signal to Noise Ratio (SNR), it is more accurate to assign the estimation a metric representing the probability of channel occupancy. Using this “soft” decision, each network node relaxes the responsibility of the final decision, relying on the aforementioned Manager Node for such task. Therefore, it is the DE of the Manager Node which elaborates a final (hard) decision based on the occupancy probabilities received by every other node, as shown in Figure 7.1.

This final decision about the occupancy of channel *K* will be performed fusing all received metrics, and obtaining a final more reliable probability to be compared to a critically selected threshold. Through this collaborative scheme two objectives are achieved:

- Reduce the uncertainty in the decision of each node, especially with low SNR levels and for networks with large number of nodes.
- Significantly reduce the probability of the "hidden node" effect. The “hidden node” problem in Spectrum Sensing environments occurs when a node does not detect the presence of a signal power due to obstruction walls, non-visibility areas, etc. This misdetection is the main issue related to standalone spectrum sensing techniques, and it is commonly avoided using cooperative sensing schemes.

Finally, the Manager Node broadcasts the information related to channel *K* occupancy and enforces an operational change of radio channel/technology if a harmful external interference is present.

The information needed by the “decision” process is summarised in Table 7-1.

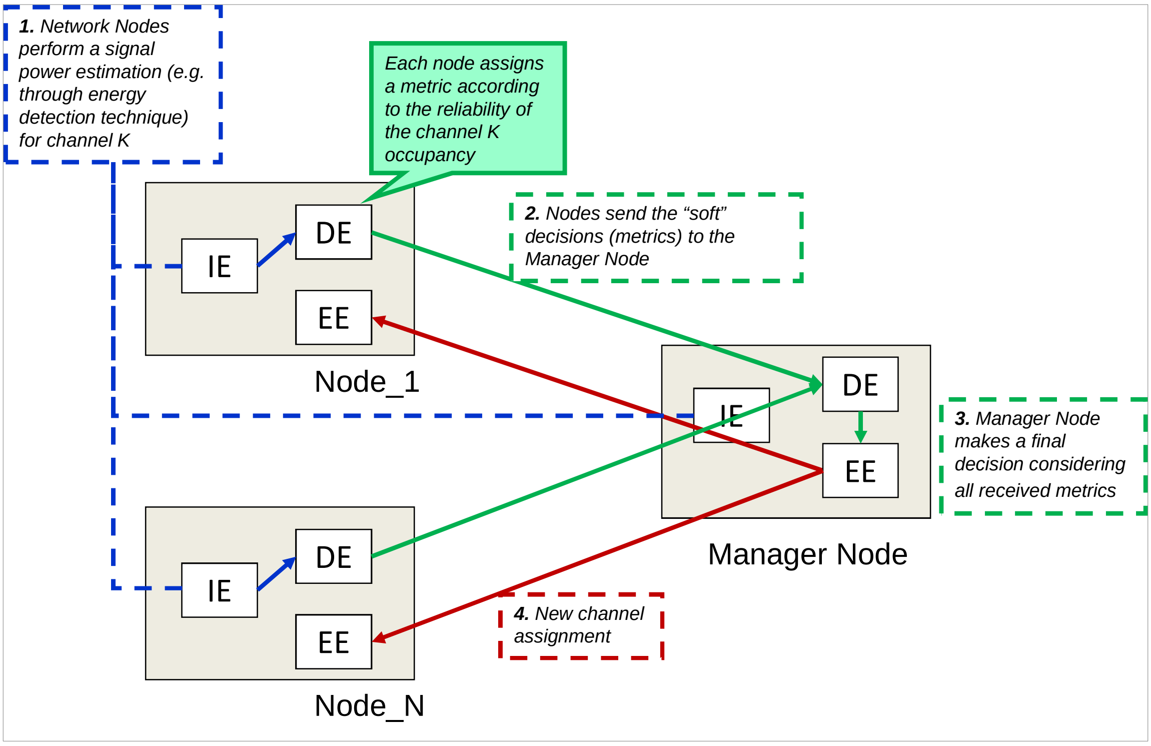

1. Network Nodes perform a signal power estimation (e.g. through energy detection technique) for channel K
Each node assigns a metric according to the reliability of the channel K occupancy
2. Nodes send the "soft" decisions (metrics) to the Manager Node
IE
DE
EE
Node_1
IE
DE
EE
Manager Node
3. Manager Node makes a final decision considering all received metrics
IE
DE
EE
Node_N
4. New channel assignment

Figure 7.1: Cooperative Spectrum Sensing mapping onto the OConS architecture

Table 7-1: Information needed for cooperative spectrum sensing

Resources	Node resources	• Type of connectivity
	Node capabilities	• Maximum Input signal bandwidth
	Radio channel resources	• Available channels.
	Monitored characteristics	• Signal power level.
	Network resources	• Number of nodes
	Context	• Propagation environment.
Policy	Strategy goals	• Maximise the probability of channel detection. • Minimise the probability of false alarm. • Simplicity and power consumption

7.1.2 CQI channel allocation in OFDMA networks

To ensure that the QoS requirements of each application are met under varying channel conditions, OFDMA networks adjust the Modulation and Coding Scheme (MCS) for every frame to the wireless channel condition of the intended receiver. When the channel condition is good, a more efficient MCS can be used. However, when the channel condition deteriorates, a more robust and less efficient MCS is appropriate.

To help the Base Station (BS) determine the appropriate MCS, every Mobile Station (MS) measures and sends Channel Quality Information (CQI) to the BS.

The BS allocates a CQI channel to every active MS. The CQI bandwidth is a scarce resource, whose allocation must be adjusted to the actual needs of the MSs. However, allocations and de-allocations of CQI channels require expensive signalling messages between the BS and each of the MSs, and therefore should be minimised. The goal is to improve efficient allocation and bandwidth utilisation of the CQI channel, for each active MS in an OFDMA network.

Our work addresses the allocation of periodic CQI feedbacks by the BS. We defined a power-of-2 CQI channel allocation, which is meant to prevent collisions between two different CQI channels (i.e. contain the same slot): rather than using a CQI slot in each frame, our scheme uses only the slots in a power-of-2 frames. A power-of-2 allocation is performed over a complete binary tree, referred to as a CQI allocation tree, while the bandwidth of each super-channel is maintained. The allocated nodes are then assigned with the fraction of the super-channel bandwidth that is assigned to the corresponding CQI channel.

Different bandwidth requirements can be assigned to different MSs by means of different tree levels. Further, our scheme does not allow CQI channel fragmentation, namely, when an MS is allocated 2 different tree nodes, thereby avoiding a non-optimal allocation.

We define an allocation framework, in which collisions and fragmentation are not allowed, every active MS is entitled to its minimum demand CQI channel before any other allocation, and we rely on a function that quantifies the profit of the system from any allocation. We then address the following 3 problems with specific algorithms that optimise the allocation:

1. How to allocate channels to the MS when the tree (super-channel) is empty
2. How to reallocate the bandwidth of a released channel to some unsatisfied MSs
3. How to change the bandwidth of a CQI channel due to changes in the profit values of an MS. Such changes are likely to be consequence of new mobility patterns. This algorithm minimises the amount of signalling messages required for the bandwidth re-assignment.

In Figure 7.2 a complete scheme for the BS is presented, addressing the possible triggers for allocation changes: an MS joins or leaves, and/or the profit function of an existing MS changes. The scheme actually guides the BS how to solve the CQI allocation problem efficiently, using our algorithms. The IE of every MS collects its CQI channel info, and reports it to its serving BS using the current CQI channel allocation (step 1). This information is captured by the DE of the base station, and also stored locally at the BS IE. The IE of the BS always keeps the current values of CQI channel allocation and MSC settings (step 2). The DE of the BS is the one that runs our algorithms, and determines a possibly modified, optimised CQI channel allocation. Such process is triggered by an external event (an MS leaves or MS joins, step 3), but also by a significant change in the value of the profit function for any MS (step 4). The current values of the profit function for every MS are computed and stored at the BS IE, and are used as input to guide the channel allocation settings process for an optimal solution. When the algorithms run on the BS DE conclude a better CQI channel allocation, a request is sent to the BS EE to implement the new allocation, and accordingly, update the MCS to the optimal settings (step 5).

In Table 7-2 the information required for the scheme is presented, at a BS level.

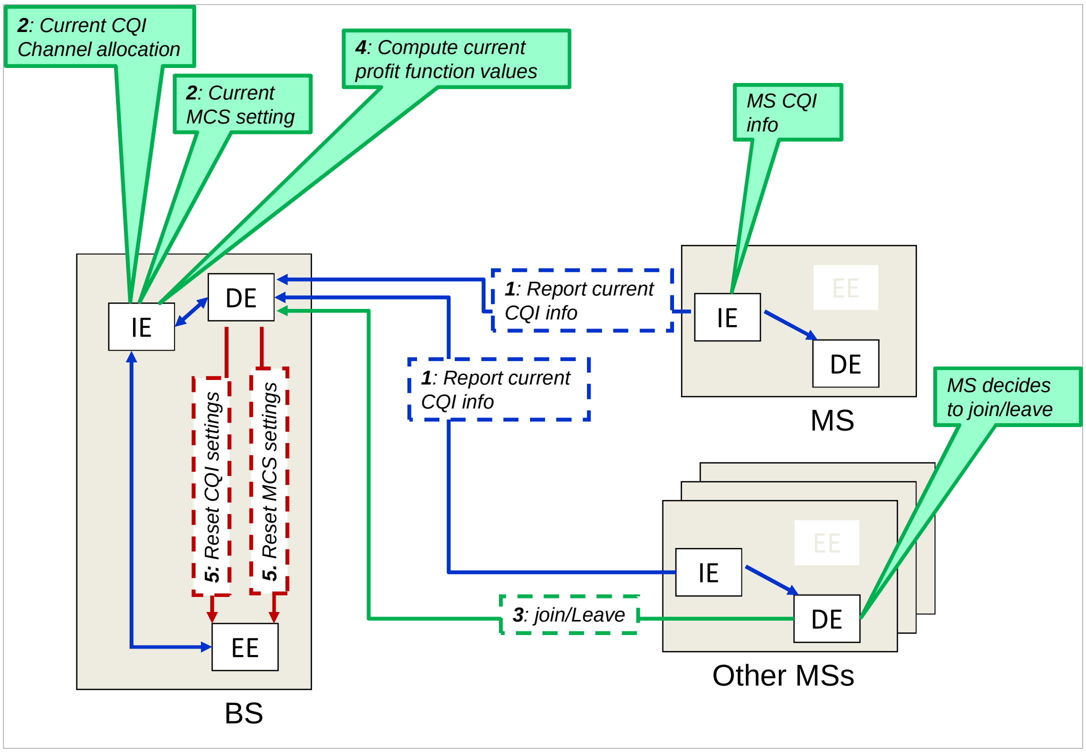


Figure 7.2: Base Station mechanism to determine the appropriate MCS mapped onto the OConS architecture

Table 7-2: Information needed for the determination of Modulation and Coding Scheme

Resources	Context	• Current MSC settings for every frame • Current CQI channel allocation • CQI information from every active mobile node
Policies/ Preferences	Operators	• Efficient and robust use of bandwidth, selecting the most appropriate MCS setting, considering the MS's channel condition • Efficient allocation of CQI channels for all mobile nodes

7.1.3 Dynamic radio resource allocation for virtual connectivity

Dynamic radio resource allocation for virtual connectivity considers the pre-allocation of radio resources within a heterogeneous wireless cluster of Base Stations (BSs), according to a Radio Access Technology priority list provided by the virtual connectivity resource requester (e.g., virtual/cloud network provider). The virtual resources with stringent requirements are periodically monitored, and the decision mechanism adapts their radio resource allocation in order to provide the contracted requirements. For optimised resource utilisation purposes, a minimum utilisation threshold is defined to detect under-utilisation situations, and the consequent reduction of radio resources assigned to the virtual one.

The following description is based on the algorithm mapping onto the OConS architecture depicted in Figure 7.3. In a first step, the Cluster Manager (CM), which can be implemented in one of the BSs of the cluster, receives the virtual resource requirements through the External-Information Entity (Ext-IE) interface (1). These requirements and the BSs' cost (2) are sent, via IE-DE interface, to the CM's DE, which decides the allocation of radio resources (3) to meet those requirements. The decision is sent to the several BSs composing the cluster via DE-IE interface (4).

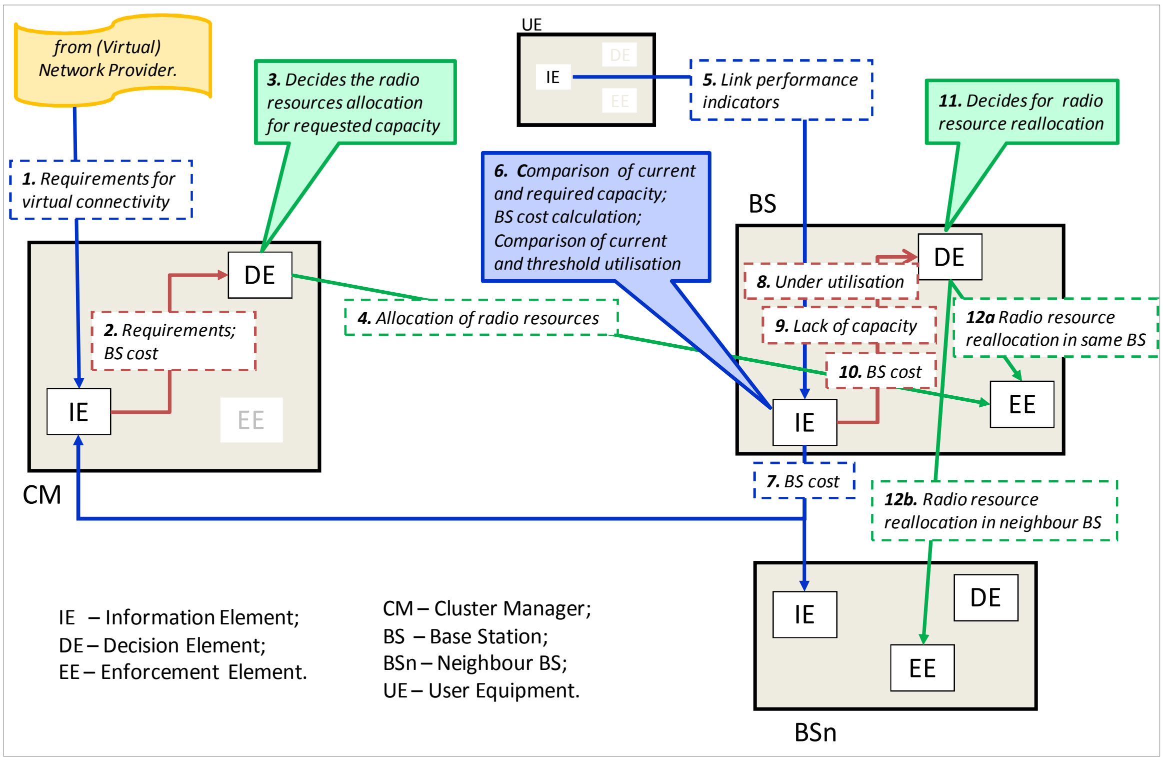

from (Virtual) Network Provider.
1. Requirements for virtual connectivity
3. Decides the radio resources allocation for requested capacity
UE
DE
IE
EE
5. Link performance indicators
6. Comparison of current and required capacity; BS cost calculation; Comparison of current and threshold utilisation
11. Decides for radio resource reallocation
BS
DE
2. Requirements; BS cost
IE
EE
CM
4. Allocation of radio resources
8. Under utilisation
9. Lack of capacity
10. BS cost
12a Radio resource reallocation in same BS
IE
EE
7. BS cost
12b. Radio resource reallocation in neighbour BS
IE – Information Element;
DE – Decision Element;
EE – Enforcement Element.
CM – Cluster Manager;
BS – Base Station;
BSn – Neighbour BS;
UE – User Equipment.
IE
DE
EE
BSn

Figure 7.3: Dynamic radio resource allocation algorithm mapped onto the OConS architecture

After the initial allocation of radio resources for virtual connectivity, the link performance of served UEs on each BS is continuously monitored by the BS's IE, through the IE-IE interface (5). The BS's IE calculates the BS cost, reflecting the BS utility according to a strategy defined by Key Performance Indicators (KPIs) relative importance (6). The comparison of current and required capacity in the IE (6) is made based on this monitored information, for detection of situations when the minimum capacity required by stringent virtual resources cannot be provided. The utilisation of the virtual resources is compared with a pre-configured threshold, also in the BS's IE (6), to identify low utilisation of the allocated radio resources.

The BS cost is sent to the IEs of CM and neighbouring BSs (7). The under utilisation (8) or the lack of capacity (9) are sent as triggers to the DE by the IE-DE interface, according to the detected situation. The BSs' cost of neighbourhood is passed to the DE also by IE-DE interface (10). When the decision algorithm (11) is triggered, it decides the reallocation of the radio resources for the virtual connectivity. In case of under utilisation, the quantity of radio resources allocated is reduced on the same BS accordingly, via DE-EE interface (12a). In addition, the DE identifies the BS with lowest cost and enough radio resources available to compensate for any lack of capacity. The result of the decision may be to enforce the reallocation of radio resources in the same BS (12a), or in another neighbour BS (12b), both through the DE-EE interface. Although this decision process is depicted just for one BS, it is running in all the BSs of the cluster.

The mechanism uses mainly the following interfaces, see Figure 7.3:

- Ext-IE -> to send requirements for new/adapt/delete a virtual resource.
  - Requirements for virtual connectivity (1): type of QoS requirements (stringent/best effort); minimum data rate; maximum delay; maximum loss rate; maximum jitter; maximum price; priority list for Radio Access Technologies. This message can be used for a new virtual resource, and for adapting or deleting an existing one.

- IE-IE –> to send requests or measurement information.
  - Link performance indicators (5): monitoring information updates from the link status, like data rate, delay, etc.
  - BS cost (7): cost value for a BS. This information is updated for the neighbouring nodes and CM, each time a BS cost changes according to the monitored information.
- IE-DE -> to send requirements or triggers to the decision process.
  - Requirements and BS cost (2): requirements for virtual connectivity received from Ext-IE interface and BSs' cost.
  - Under utilisation (8): trigger with virtual resource ID and percentage of utilisation, so that the DE determines the amount of radio resources to be reduced.
  - Lack of capacity (9): trigger with virtual resource ID and amount of missing capacity, so that the DE determines the amount of radio resources to be added.
  - BS cost (10): send cost values of neighbouring BSs.
- DE-EE -> to enforce the decision results.
  - Allocation of radio resources (4): amount of radio resources allocated in each BS for the specified virtual resource ID.
  - Radio resource reallocation in same BS (12a): to add or reduce the amount of radio resources in the BS for the specified virtual resource ID.
  - Radio resource reallocation in neighbour BS (12b): to add or reduce the amount of radio resources in the BS for the specified virtual resource ID.

The information needed by the "decision" process is summarised in Table 7-3. The resources information, related to each BS, is separated into:

- Static, being specific of each Radio Access Technology, and individual BS. The neighbourhood and the set of BSs in the cluster are considered in this group, since a fixed infrastructure is assumed.
- Dynamic, consisting of the KPIs that are varying with the network operation, thus, reflecting the BS status.

The requirement information is split into application/service requirements and user/client requirements. The requirements considered in these decision mechanisms are essentially QoS ones, for both the service and the client.

The application/services requirements may be fixed, depending only of the application itself, or may be indicated in the request for the service, since they can also depend on other factors, e.g., the capabilities of end-user devices. The user/client – which from this viewpoint may be, e.g., the (Virtual) Network Provider – must indicate the QoS requirements for the virtual resource it is asking for, as well as some additional parameters like Radio Access Technology preferences and maximum price it is willing to pay.

Concerning the policies, the user/client and the operator preferences put together the strategy for BS evaluation and resource optimisation. Other operator policies, like thresholds and borrowing margins, allow for the fine-tuning of the decision algorithm for additional optimised resource management. Related to the service policies, the list of preferred RATs is identified.

Table 7-3: Information needed for dynamic radio resource allocation for virtual connectivity

Resources	Static information related with each BS	Operator ID; Price/cost (deployment/operation); min Delay; min Jitter; Max Data Rate; Mobility level; neighbouring nodes; set of nodes in the cluster.
	Dynamic information related with each BS	Link reliability; link data rate; link utilisation; energy consumption; delay; CPU utilisation; Memory utilisation.
Requirements	Application/Service	Type of QoS requirements (Stringent/Best Effort); minimum data rate; maximum delay; maximum loss rate; maximum jitter.
	User/Client	Type of QoS requirements (Stringent/Best Effort); minimum data rate; maximum delay; maximum loss rate; maximum jitter; maximum price.
Policies	User/Client	Weight for each Key Performance Indicator (e.g., QoS parameters, price); preferred operator
	Operator	Weight for each Key Performance Indicator (QoS parameters, price); Borrowing margins (for radio resource allocation), according to QoS requirements type; Thresholds for balancing mechanisms; cooperation strategies.
	Service	Priority list for RATs.

7.2 Access networking techniques with Mesh and DTNs

7.2.1 Radio Resource Management for Wireless Mesh Networks Connectivity

Wireless Mesh Networks are two-tier architecture with several nodes, the so-called Mesh Points, building a Radio Backhaul Network that provides Internet access to Users Equipment through a Radio Access Network. Multi-radio Mesh Access Points (MAPs) support the simultaneous operation of N+1 radio channels. MAPs combine these two functionalities: one Radio Access Network's radio channel operates as a classical Access Point; the remaining N radio channels are used for mesh forwarding, acting as a wireless router and building a multi-hop self-organised Radio Backhaul Network. Some are Mesh Point Portals (MPPs), gateways to the core network.

Each multi-radio node has a *radio agnostic abstraction-layer*, representing the abstraction of a single radio channel to higher layers. It enables the simultaneous operation on multiple radio channels, controlling the specific channels, transmission power levels and physical data rates. The considered nodes operate simultaneously on N=2 radio channels, according to a hybrid policy: operation on a fixed radio channel, announced as receiving stable-channel; dynamic operation on remaining radio channels, periodically switching among them.

A Unified Mesh Radio Resource Management (UMRRM) distributed strategy is implemented on the *radio agnostic abstraction-layer*, integrating various mechanisms for the optimisation of the data rate, power and channel of each radio-interface:

- Load-aware and interference-aware channel assignment strategy, guaranteeing connectivity with any neighbour;

- Rate adaptation, aware of Wireless Mesh Networks traffic load specificities;
- Energy-efficient power control, addressing the non-homogeneity of nodes' data rates;
- Gateway-control, enforcing data rate limits to flows in order to achieve max-min throughput fairness per MAP.

UMRRM maximises the exploitation of network resources, guaranteeing fairness and minimising spectrum and power usage.

The mapping of the UMRRM strategy onto the OConS architecture is depicted in Figure 7.4. The UMRRM strategy is supported by a resources monitoring and sharing mechanism. Each MAP's IE continuously monitors various resources, building and storing Key Performance Indicators (KPIs) (1). Periodically, each MAPs' IE broadcasts a hello control message through the IE-IE interface, advertising information about itself (node ID, geographic positioning, KPIs) and from nodes of its extended neighbourhood (2). This procedure enables each MAP to discover and update, in its IE, the neighbourhood's resources usage table (3). Periodically, this information is passed onto the DE by the IE-DE interface, for evaluation of the node's resources (4). The UMRRM strategy (combining rate adaptation, power control and channel assignment) is used by the DE to optimise the MAP operating radio channels and respective physical data rate and transmission power levels (5). In case a reallocation of resources is needed, DE enforces it in the EE, through the DE-EE interface (6).

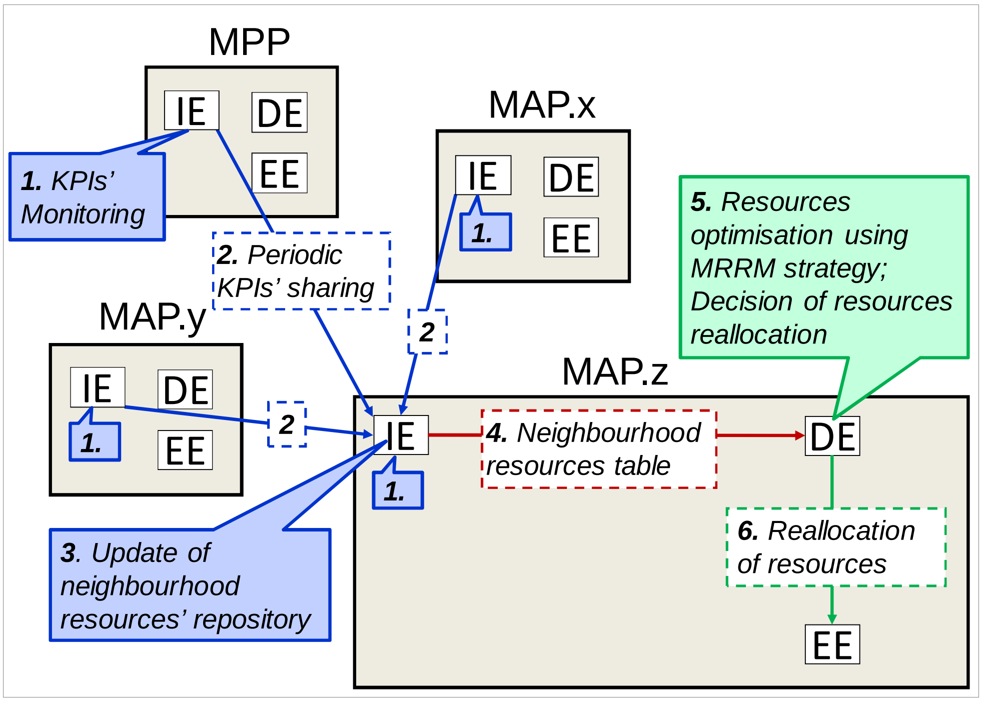

MPP
IE
DE
EE
MAP.x
1. KPIs' Monitoring
2. Periodic KPIs' sharing
5. Resources optimisation using MRRM strategy; Decision of resources reallocation
MAP.y
2
2
MAP.z
1.
4. Neighbourhood resources table
6. Reallocation of resources
3. Update of neighbourhood resources' repository

Figure 7.4: UMRRM strategy mapping onto the OConS architecture.

The presented UMRRM mechanism uses the following interfaces identified in Figure 7.4:

- IE-IE: it is used to share the node's KPIs with its neighbours. It is supported by Hello messages broadcasted on all radio channels, to reach every neighbour.
- IE-DE: it is used to pass, within each node, the information needed by the UMRRM strategy from IE to DE. It consists of the neighbourhood's resources usage table that contains resources' information of neighbouring nodes.

- DE-EE: it is used, within each node, to enforce the decisions by DE in the EE. It is supported by the radio agnostic abstraction-layer, which controls the operating radio channels of the node, enforcing the allocated resources as decided in DE.

The information needed by the decision process is summarised in Table 7-4.

Table 7-4: Information needed for deciding the UMRRM strategy

Resources	Node resources	• Number of simultaneously supported radio channels. • Number of radio channels for mesh forwarding.
	Node capabilities	• Mesh Point wireless router. • Mesh Access Point. • Mesh Point Portal.
	Radio channel resources	• Available channels. • Available data-rates. • Transmission power levels.
	Monitored characteristics	• Key Performance Indicators weighting activity, rate, number of neighbours, hop distance to MPP. • Transmission power level.
	Network resources	• Number of MAPs and location. • Number of gateways. • Topology.
	Context	• Propagation environment. • Subscribers' density and overbooking factor.
Requirements	Mesh node	• "Max-min fair" aggregated throughput.
Policy	Strategy goals	• Guarantee connectivity with neighbours. • Minimise interference. • Minimise energy consumption. • Maximise throughput.

7.2.2 Routing and Forwarding Strategies in DTNs

Exploring self-* and connectivity properties of nodes belonging to a DTN (context awareness) can lead to an optimised strategy to improve transport and service performance in this specific type of networks.

Connectivity in DTN scenarios implies that nodes do not have permanent physical paths to certain destinations, but only to some of their close neighbours instead. Our work aims at the implementation of a methodology that helps the node making decisions regarding packet routing and forwarding. That is, when a node receives a packet to a certain destination it needs to analyse its available information and make a decision regarding several aspects:

- Accept or discard the packet due to buffer constraints.
- Verify if connection path to destination exists.
- Store the packet and wait for a suitable forwarding instant (based on the probability of reaching destination node in a certain moment).
- Forward the packet immediately to an intermediate node with higher probability of finding the destination (based on connectivity or mobility pattern estimation, self-learning parameters, etc.).

- Combine the packet with other packets previously stored, headed to the same or different destination node, by using NC techniques (see Section 5.1.1).

There is a wide range of combinations that could be validated for several specific situations where delay tolerant transmissions can be optimised so as to be characterised by a certain expected QoS, for instance. We will derive and implement an algorithm that makes use of a valuable subset of these properties and is able to exploit them for a smart management of the connectivity in DTNs formed by human-carried devices. We have studied the benefit from applying knowledge about human behaviour and mobility (useful patterns of common daily routines of people) to the forwarding strategy followed by their DTN mobile devices.

We have designed the mechanism used by all DTN nodes to derive and learn both self and someone else's statistics regarding connectivity patterns. We have based these metrics in two key parameters: the inter contact time between each pair of nodes, and the duration of pair wise contacts. This is an adaptive algorithm, which means that each node is permanently learning and updating its contact database and metrics, but also interchanges information with neighbours so as to take advantage of them for smart learning. In Figure 7.5 the information exchange among DTN nodes is represented in two phases or steps. Step 1 corresponds to the self-learning process, where each node timestamps its neighbour contacts in order to estimate powered mean values of inter-contact and contact duration times with them. A rating algorithm is based on these pair of estimated values so that neighbours are rated with a certain probability of contact. As can be seen, DTN Node A stores its table of contacts, rating each of them with a numeric value/probability.

Step 2 of the picture represents how Node A incorporates information learned from other DTN Nodes in a collaborative fashion. If Node B (or C) has a higher rating/probability of reaching Node X than the rating value stored by Node A, the latter will update Node X's value with that one learned from Node B (or C) and will choose that path if it has packets headed to Node X in the future.

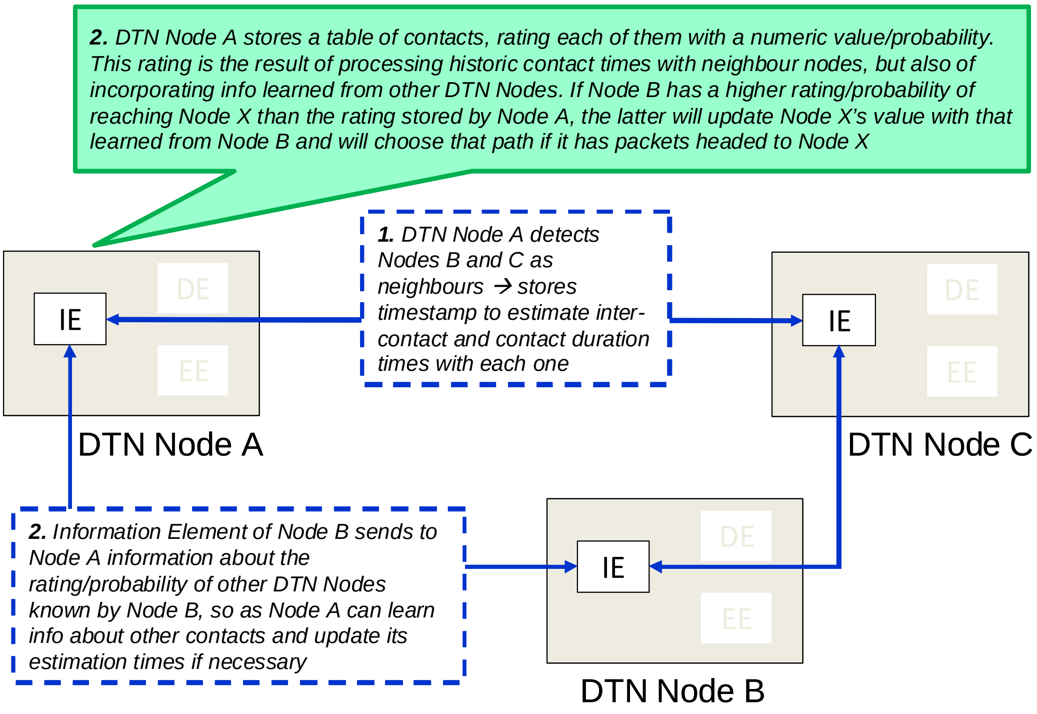

2. *DTN Node A stores a table of contacts, rating each of them with a numeric value/probability. This rating is the result of processing historic contact times with neighbour nodes, but also of incorporating info learned from other DTN Nodes. If Node B has a higher rating/probability of reaching Node X than the rating stored by Node A, the latter will update Node X's value with that learned from Node B and will choose that path if it has packets headed to Node X*
1. *DTN Node A detects Nodes B and C as neighbours → stores timestamp to estimate inter-contact and contact duration times with each one*
IE
DE
EE
DTN Node A
IE
DE
EE
DTN Node C
2. *Information Element of Node B sends to Node A information about the rating/probability of other DTN Nodes known by Node B, so as Node A can learn info about other contacts and update its estimation times if necessary*
IE
DE
EE
DTN Node B

Figure 7.5: Self and collaborative learning mapping onto the OConS architecture

Figure 7.6 summarises the process of collecting all criteria influencing the decision of a path selection (forwarding path), and enforcing the channel establishment with the selected next hop. We can see that Step 1 of the decision algorithm collects all kind of parameters to be considered from the IE of each local node (if this mechanism is combined with NC as described in Section 5.1.1, both decisions would be applied at this stage in the process).

In Step 2 of the sequence, DTN Node A takes the decision of sending packets headed to Node X (not its direct neighbour) via Node C (best rated candidate/route to reach Node X). Decision is taken according to several criteria (probability of reaching Node X sooner, buffer capacity, available autonomy in terms of energy, etc).

Step 3 is the enforcement action taken by EE of Nodes A and C in which a transmission channel is established to actually send the packet.

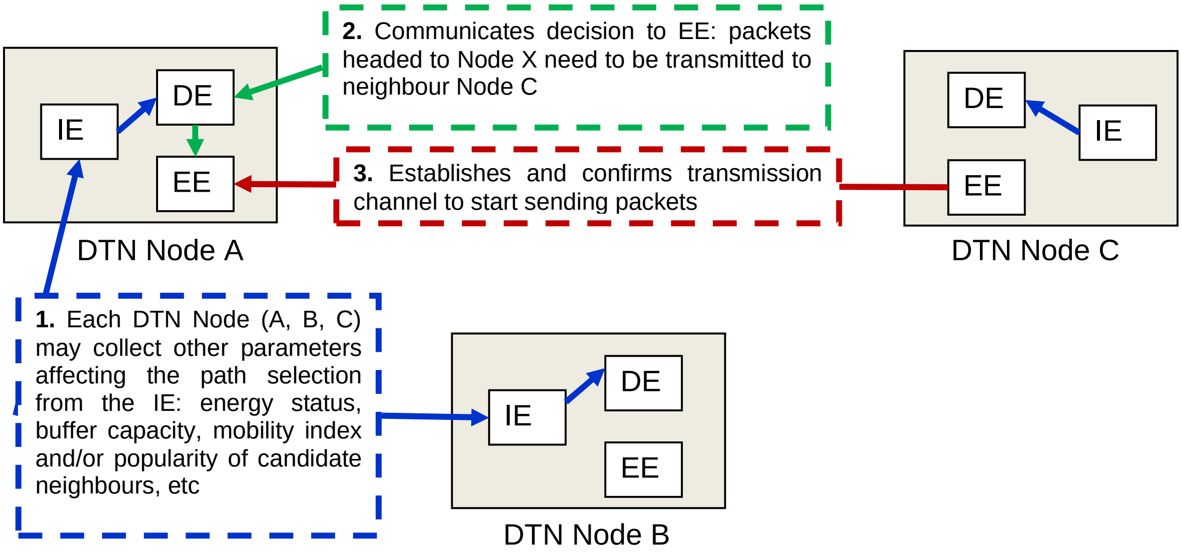

2. Communicates decision to EE: packets headed to Node X need to be transmitted to neighbour Node C
DE
IE
EE
DTN Node A
3. Establishes and confirms transmission channel to start sending packets
DE
IE
EE
DTN Node C
1. Each DTN Node (A, B, C) may collect other parameters affecting the path selection from the IE: energy status, buffer capacity, mobility index and/or popularity of candidate neighbours, etc
IE
DE
EE
DTN Node B

Figure 7.6: Next hop selection mapping onto the OConS architecture

In Delay Tolerant Networks the inner model implements a distributed approach, where most decisions are taken in a hop-by-hop basis, and its scope could be seen as quite short-ranged in a sense. For the routing and forwarding strategy we think that this is also applicable, since each node maintains its own list of contacts and needs to make its own forwarding decisions locally. However, it may also occur that for specific situations an external entity could enforce a certain policy (establish priorities to certain parameters considered within the algorithm) to be applied by all or by a subset of nodes.

The kind of information we need to use can be summarised in the following aspects: probability of contacts between nodes (derived from previous history and/or learned during runtime), energy consumption, popularity index of neighbour nodes (send packets to the most popular neighbour if no path is available), mobility index of neighbour nodes, available buffer size, fragment/reassembly capacity of nodes, quality of previous connections with neighbour nodes (delivery ratio in previous attempts), etc.

More specifically, the information needed by the decision process is summarised in Table 7-5.

Table 7-5: Information needed for Routing and Forwarding strategies in DTNs

Resources	Node resources	• Available buffer size • Available batteries • Mobility index
	Node capabilities	• Fragmentation/Reassembly
	Network resources	• Supported transmission technologies
	Context	• Inter-contact time and contact duration estimations • Popularity index
Requirements	Application	• Packet size estimation / Preferred policy
Policy	Strategy goals	• Minimise energy consumption • Maximise throughput • Prioritise frequent and/or long contact duration times

7.3 Core networking techniques for Routing and Data-Centres Interconnection

7.3.1 Policy based routing enhancements

The target is to manage and control the Advanced Connectivity Services in an efficient and scalable way, specifically investigate Policy-based Routing enhancements.

One of the main reasons that BGP is heavily used in current Internet is that it supports policy-based routing. Policy-based routing allows ASs to deploy routing schemes that reflect the commercial agreements they have with peering ASs. However, when deploying policy based routing, other desired properties are not taken into account: for example, routing along shortest paths. The shortest path routing is important for efficient use of routing resources. It also contributes to reduced packet latency that is crucial for interactive voice and video services.

Another undesired property of policy based routing is that the concatenation of two legal paths may be illegal due to policy constraints. For example, a direct path A->B may be legal, a direct path B->C may be legal, but the path A->B->C may be illegal (if B is a customer of both A and C).

We consider the deployment of routing middle points or service gateways. These in-network devices can be used by the flows to enable shortest path or to validate path concatenation, so in the above example - if such a device is located in AS B, then one could realise the A->B->C path.

Overlay routing is a very attractive scheme that allows improving certain properties of the routing without the need to change the standards of the current underlying routing. However, deploying overlay routing requires the placement and maintenance of overlay infrastructure. This gives rise to the following optimisation problem: find a minimal set of overlay nodes such that the required routing properties are satisfied.

In our research we rigorously study this optimisation problem. We show that it is NP hard and derive a non-trivial approximation algorithm for it, where the approximation ratio depends on specific properties of the problem at hand. We examine the practical aspects of the scheme by evaluating the gain one can get over the BGP routing problem, and we show, using up-to-date data reflecting the current BGP routing policy in the Internet, that a relatively small number of relay servers are sufficient to enable routing over the shortest paths from a single source to all ASes.

The mechanism studied here is not meant to be executed over the real-time OConS network infrastructure. Our mechanism is an optimisation study that is executed in a simulated or experimental network, in order to gain better understanding regarding the shortcoming of current provisioning, and in order to identify possible enhancements, new dimensioning, or better configuration for it.

The results of our study can be used as benchmarking that guides network operators with regards to optimised setup of overlay routing. Our mechanism, acting as a "remote" DE, runs off-line, with information provided by IEs. This analysis is then used to change the overlay routing setup in the real-time OConS network.

The information required for the proposed scheme, at a base station level, is summarised in Table 7-6.

Table 7-6: Information needed for policy-based routing

Resources	Networks	• A list of all Autonomous Systems (AS) • Their pair-wise commercial relationships (Peering, Transit) • Their routing table (we can do without it, since it can be compiled from the above info)
Policies/ Preferences	Operator	• Routing along shortest path • Concatenation of any two valid paths must be valid • Minimal set of overlay nodes to satisfy the above-listed two requirements

7.3.2 Overlay for Data-centres Interconnection

Migrating from current overlay networks used for Data-centre interconnection to OConS is likely to be a multi-step process. Adoption will be based on the technology's maturity and on the effective gain it provides for the Network Provider/Operator in terms of:

1. Better bandwidth utilisation.
2. Efficient equipment that reduce the footprint (better density) and power budget (green).
3. Ease of management reducing the OPEX.
4. Technologies that foster the re-utilisation of the deployed hardware and transform costly network migration tasks into a simple software update.

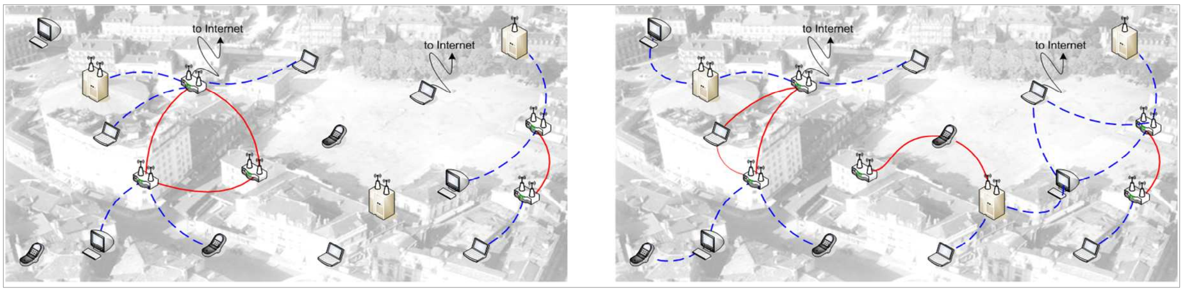
A possible migration path for Data-centre interconnection technologies is:

1. Extensions to VPLS to extend virtual bridges.
2. Transparent Interconnection of Lots of Links (TRILL), see [RFC5556].
3. OpenFlow, when it becomes a carrier grade technology; e.g., Split Architecture Carrier Grade Networks (SPARC) EU-FP7 project is currently investigating the requirements and extensions to OpenFlow, in order to make it a carrier grade technology.

8 Mapping the OConS Architecture on Use-Cases

8.1 Creating and Sustaining the Connectivity in Wireless Challenged Networks

The first considered use-case deals with “Creating and Sustaining the Connectivity in Wireless Challenged Networks” [SAIL-D.A.1]. Consider several (heterogeneous) wireless nodes willing to build a multi-hop network in order to provide the end-users with connectivity between them and towards a fixed Internet infrastructure. This communication environment is often under adverse conditions, Figure 8.1a, e.g., expectations of connectivity between certain nodes no longer holds, or congestion is experienced on some links because of the multiple simultaneous requests from the crowd. The sharing of nodes’ resources in a cooperative and self-organised way enables the optimisation of the whole network, Figure 8.1b. Innovative techniques are needed that explore the resources and communication conditions in the best way, creating and sustaining the connectivity. A detailed discussion of this use case is available in deliverable DA1 [SAIL-D.A.1].



a) b)

Figure 8.1: Wireless Challenged Network: (a) Spontaneous network with limited connectivity; (b) Spontaneous network after optimisation of connectivity thanks to innovative techniques

Figure 8.1b illustrates a wireless challenged network, where the architectural functional entities are represented. The nodes’ operation follows self-organised management principles:

- Newly added nodes self-configure themselves in a plug-and-play fashion.
- Nodes regularly self-optimise their resources in response to changes in the network.
- In the event of a node failure, self-healing mechanisms are triggered in the surrounding nodes to alleviate gaps of connectivity, coverage or capacity.

These principles require more complexity on the nodes’ operation since strategies are challenged to operate with local information only. Nevertheless, they enable the flexible and spontaneous deployment of a network.

Nodes follow a distributed management, where the different functional entities and interfaces are identified in Figure 8.2:

- Nodes monitor a set of Key Performance Indicators (KPIs), such as used channels, load, distance to the nearest gateway, data-rate and power (1).
- KPIs or metric(s) weighting multiple KPIs are shared within the node’s neighbourhood, through message broadcast through the IE-IE interface (2).
- Neighbourhood KPI’s information is collected and compiled by each node, being dynamically updated (3).
- Based on the collected information (4) and on specific strategies, a node takes a decision (IE-DE) (5).
- The decision is then enforced locally on the same node (DE-EE), or communicated to another node where the enforcement must be done (DE-EE) (6).

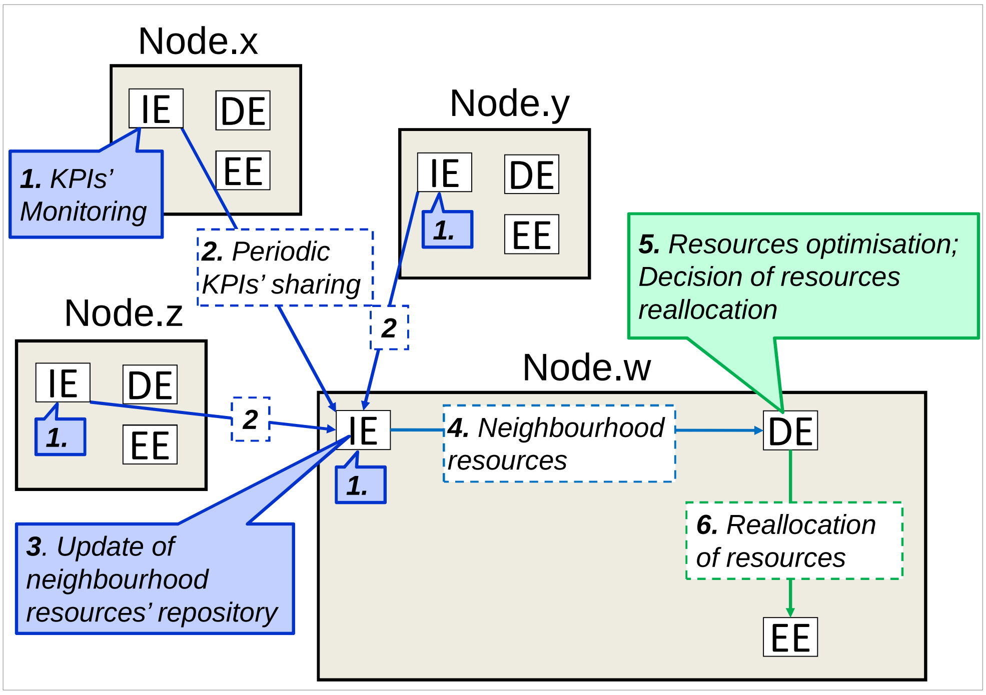

Node.x
IE
DE
EE
1. KPIs' Monitoring
Node.y
IE
DE
1.
EE
2. Periodic KPIs' sharing
2
5. Resources optimisation; Decision of resources reallocation
Node.z
IE
DE
1.
EE
2
Node.w
IE
4. Neighbourhood resources
DE
1.
3. Update of neighbourhood resources' repository
6. Reallocation of resources
EE

Figure 8.2: Mapping the OConS architecture on Wireless Challenged Network use-case

The mechanisms used do address this use-case are as follows:

- Network coding for M-to-N routing in DTNs (Section 5.1.1)
- Network coding and transport over wireless (Section 5.1.2)
- Efficient handover multi-hop wireless networks (Section 6.1.5)
- Cognitive radio systems through spectrum sensing techniques (Section 7.1.1)
- CQI channel allocation in OFDMA networks (Section 7.1.2)
- Radio Resource Management for Wireless Mesh Networks Connectivity (Section 7.2.1)
- Routing and Forwarding Strategies in DTNs (Section 7.2.2)

8.2 Multiple Path/Protocol (Multi-P) Transport for Optimised Service Delivery of Heterogeneous Content

The objective is to adopt mechanisms to deliver content to users over multiple paths and using multiple protocols to improve the user experience. With Multi-P, the OConS enabled transport mechanisms are made aware of the requirements of the upper layers of the protocol stack, which may be legacy as well as future network architecture based. The OConS Architecture in respect to the enabling of Multi-P therefore performs the following tasks:

- Information is collected by the IEs located at different entities to provide information such as available path information, application/user requirements/preferences, policy information and mobility information
- The DEs then decide what protocols and what paths to use based on the available information (got from IEs) and by applying the selection algorithms available to the DEs
- Finally, the DEs inform the EEs to configure and setup paths and protocols to carry content along the different paths utilising the protocols identified by the DEs

The different elements of the OConS Architecture may reside in the network or in the user terminal or at both locations. Further, multiple instances of some elements may also be active at the same time and, therefore, coordination mechanisms will ensure that no conflicts will arise when deciding the paths and the protocols to use.

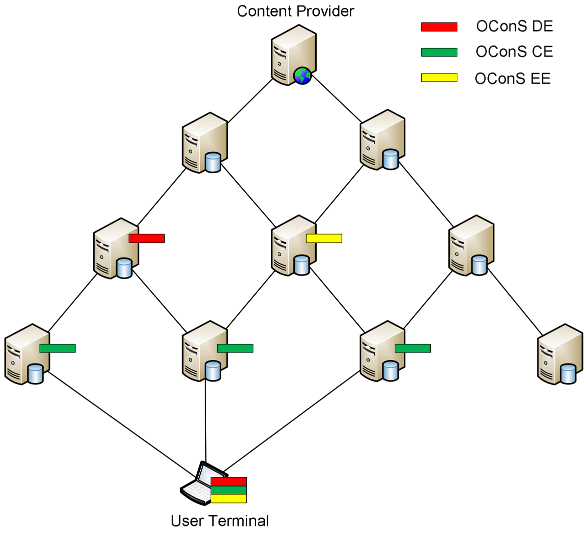


Figure 8.3: Mapping the OConS architecture on a Multi-P-enabled Network

From an end-to-end perspective, IE, DE and EE are contained within a single entity of the transport layer protocols. The EE is in charge of actually sending packets along the selected paths, following decisions from the congestion control and packet scheduling and distribution algorithms (the DE). Information is collected by observing and collecting feedback from the peer, in order to characterise the paths in use.

From an OConS perspective, this closed loop is made more open to allow sharing transport-local information with other OConS entities. Similarly, it could also benefit from externally available information (Figure 8.4). Therefore, end-to-end multipath transport protocols can be seen as the aggregation of all IE, DE and EE.

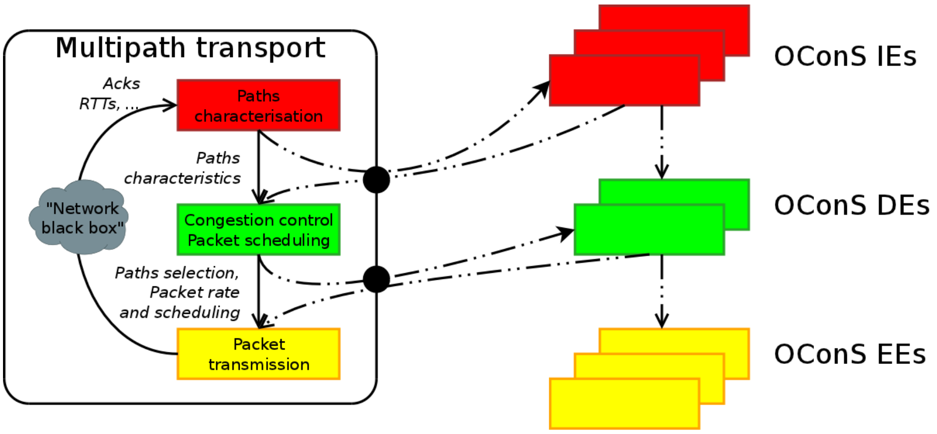


Figure 8.4: End-to-end transport seen as a logical group of all three types of OConS entities

The following mechanisms are developed to handle Multi-P related transport for legacy and future networks:

- Multi-Path Extensions for Information-Centric Networks (Section 5.2.4)
- Multi-Path and Multi-Protocol Transport for Enhanced Congestion Control (Section 5.2.6)

8.3 Optimising the QoE for End-users with adequate management of the Cloud/Network services

Within the Flash crowd scenario, diverse services are provided to the end-users through several network infrastructures, such as 3G/4G Base Stations and Wi-Fi Access Points, where these may dynamically become available but also disappear. Another possibility for the end-users is to gain the connectivity through the accesses provided by a self-organised community (i.e. mesh network) in a public location, and which could complement the operator-managed communication infrastructures. Furthermore, ad-hoc networks (e.g. using Wi-Fi Direct) may also be available, yet technology failure or power shortage may sometime disable one or more technologies or devices. Obviously, some of these access alternatives might be operated by different actors, with whom a given end-user could have (or not) a business agreement.

Accordingly, this use case deals with the decision making mechanisms to "optimally" choose the interfaces, the access networks and/or the paths (and hence to schedule/map/route the traffic flows accordingly) in order to achieve the highest possible quality-of-experience (QoE) for the end-users.

Besides, this will likely imply appropriate management of connectivity and optimisations (i.e., trade-off and/or tuning) at the Cloud/Network providers/operators level, as well as at the Service/Application level. Likewise, these decision mechanisms need to cope with (sometime) contradictory goals of the involved actors, because various policies from network providers/operators and end-users, as well as requirements from service providers (which might include e.g. the NetInf providers) need to be considered.

Taking as generic scenario the one introduced in Figure 1.1, we depict a possible example in Figure 8.5 to show how the different actors (e.g., network operators and service providers) exchange various pieces of information in order to optimise the QoE provided to the end-users; Consequently, they are using their respective IEs to collect the diverse information from the networks nodes, the devices and the services/applications (including e.g. NetInf or CloNe operators/providers).

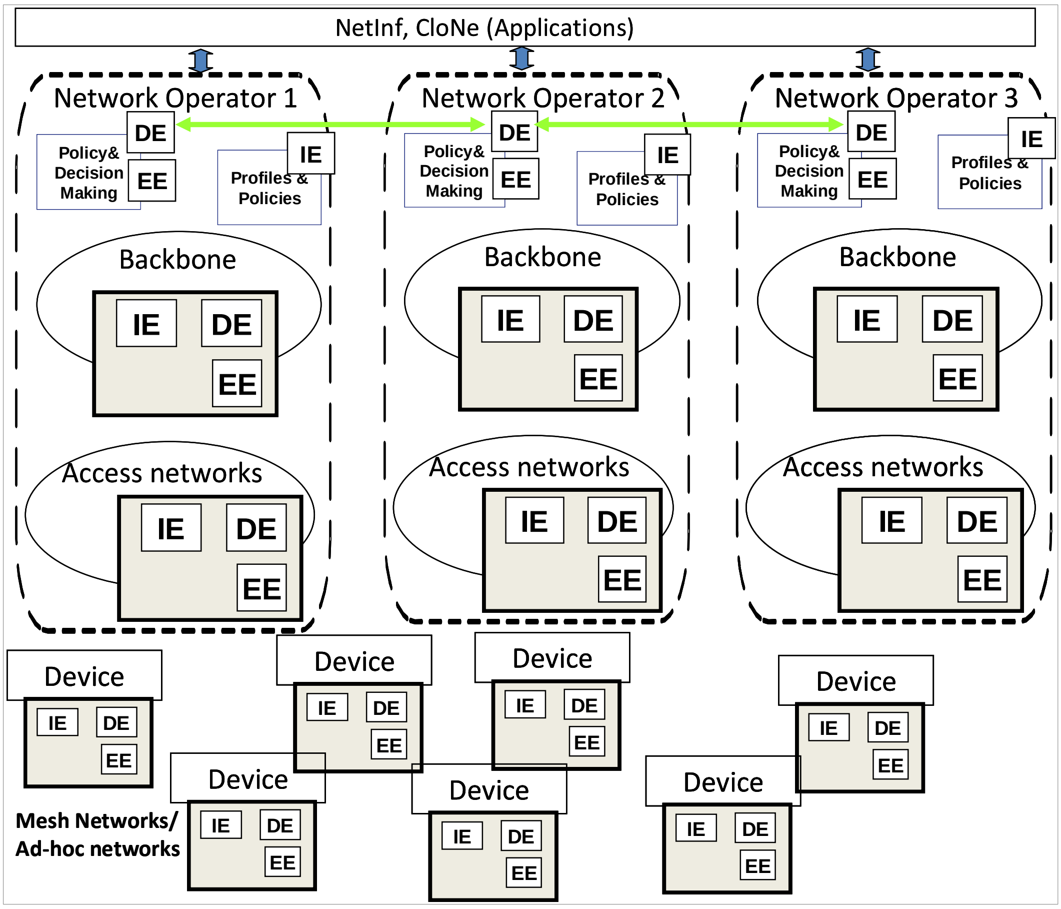


Figure 8.5: Optimising the QoE for end users

In Figure 8.5 only the network operators are explicitly described, but a plethora of other actors can coexist, as previously introduced.

Based on the collected information and using their own internal policies and algorithms, the different network operators can thus exchange information and decisions regarding the networking context through their DEs. Moreover, the DEs can possible make their decisions based also on the information received from Clone and NetInf.

Within the OConS framework, each time a decision is made, it has to be enforced with the appropriate EEs, i.e. the DEs are interfacing (choosing, activating, and steering) with the corresponding execution protocols and procedures (e.g. mobility anchoring and tunnelling, or multi-path/protocol transport services before mentioned).

With this use-case we mainly address the mechanisms described in Section 6, namely:

- Access selection and decision algorithms (Section 6.1.1)
- Dynamic/Distributed mobility management (Section 6.1.2)
- Mobile-driven network selection and flow scheduling (Section 6.1.3)
- Centralised optimisation of mobility management with a self-adapting network (Section 6.1.4)

8.4 (Autonomous) Interoperation and Connectivity of Cloud and NetInf data-centres – Data-Centre Interconnect

When connecting two or more data-centres a great deal of fluctuating traffic can occur. In [Benson10] the authors performed an empirical study of the network traffic in 10 data-centres (DC) differentiated in three categories: university, enterprise campus, and cloud data-centres. Of course the traffic depends heavily on the kind of application, either customer-facing applications, such as Web services, file stores, authentication services, custom enterprise applications or data intensive applications, such as MapReduce and search indexing. However some findings are as follows:

1. Data-centre traffic is statistically different from wide area traffic. This has an impact on design and implementation of techniques for data-centre networks
2. Link utilisations in data-centres are subject to time-of-day and day-of-week effects, variations in some data-centres are nearly an order of magnitude more pronounced at core links than at edge and aggregation links.

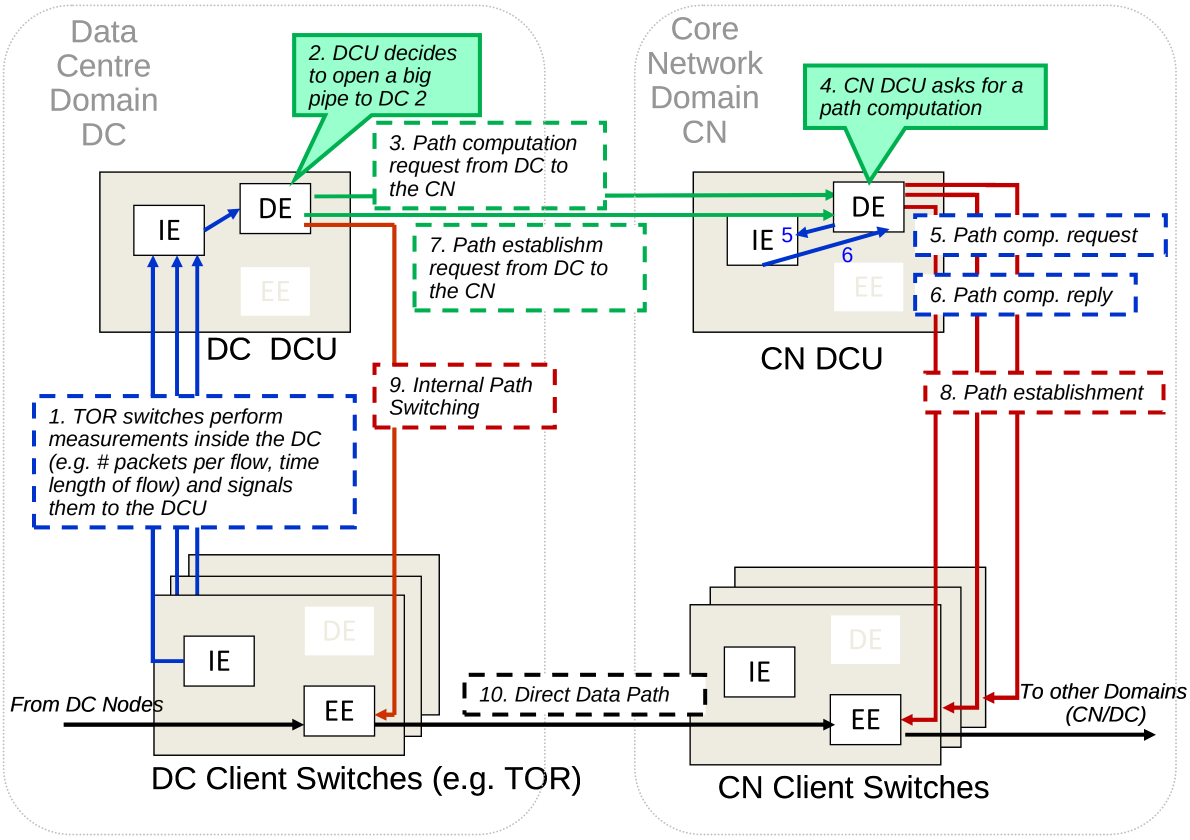
While this is just an example of 10 data-centres, at least it gives a hint what the trend is. As data-centres go distributed in the future, more and more of that traffic will be seen between them. Furthermore, data-centres sites will also become bigger in term of involved nodes. OConS has to react to such varying network loads and there are two possibilities to handle this issue:

1. Application (data-centre) agnostic, i.e. the network performs measurements and reacts accordingly to application traffic changes
2. Application controlled, i.e., it needs a control interface between the application and the network control. The application could be a whole data-centre, e.g., the data-centre OMC.

The first solution relates more to a control plane treatment within the wide area (core) network, while the second option requires a control or even management interface to cooperate between the DC and the WAN. However the lines between both solutions are often blurred with advantages and disadvantages on each side. Although we have focused on the first approach so far, our proposed solution can easily cope also with the second possibility, e.g. in case of a planned database backup, VM movement or other maintenance controlled by the DC management. Then the trigger to set up the network comes from the DC OMC.

Our aim is to design new flexible path finding algorithm across multiple network domains and/or layers that makes use of additional information like energy per transported bit and other smart management mechanisms for future connectivity.

In Figure 8.6 the information exchange among data-centre (DC) Top of Rack (TOR) client switches, data-centres Domain Control Units (DCU) and their counterparts in the core network (CN) is depicted in two phases comprising ten steps. The first phase is the collection of measurements and the detection of a big chunk of data transport going on, called an elephant flow within the DC. The second phase is the addition of a fat pipe via the CN to a remote data-centre. Note that the network topology construction inside the CN that comprises of IE-IE signalling in the CN is not displayed in Figure 8.6 but is of course needed for the path calculation inside the CN DCU. Also note that the interworking in this figure follows modelling style for the mechanisms and services as introduced in Section 4.2.


Data Centre Domain DC
2. DCU decides to open a big pipe to DC 2
3. Path computation request from DC to the CN
IE
DE
EE
7. Path establishm request from DC to the CN
DC DCU
9. Internal Path Switching
1. TOR switches perform measurements inside the DC (e.g. # packets per flow, time length of flow) and signals them to the DCU
DE
IE
EE
From DC Nodes
10. Direct Data Path
DC Client Switches (e.g. TOR)
Core Network Domain CN
4. CN DCU asks for a path computation
DE
IE
5
6
EE
5. Path comp. request
6. Path comp. reply
CN DCU
8. Path establishment
DE
IE
EE
To other Domains (CN/DC)
CN Client Switches

DC —	Data-Centre	IE —	Information Element
TOR —	Top of Rack	EE —	Execution Element
DCU —	Domain Control Unit	DE —	Decision Element
CN —	Core Network		

Figure 8.6: Data-centre Interconnect with DCU mapping onto the OConS architecture: Example of single domain path establishment procedure (simplified view without explicit border nodes; message responses not shown)

Step 1 corresponds to the measurement process, where each Top of Rack switch collects data about the existing flows. Here we also have the possibility to perform measurements inside the other switches/routers and then in the DC DCU do a sliding window or time based decision, maybe even looking at protocol headers (RTSP, RTMP, MMS, HTML5, …) and port numbers (RTSP port 554 tcp/udp, RTMP port 443 tcp, MMS port 654 tcp), see [TUPN]. The decision Step 2 consists of a threshold algorithm which decides about the need of an additional high bandwidth link (fat pipe) to the remote data-centre. The fat pipe is unidirectional.

Step 3 of the picture represents the path computation request to the CN which, in Steps 4 and 5, computes suitable paths that are delivered in Step6. The final selection of the path to be used is done either in the CN DE or in the DC DE. Finally, with Step 7 the path establishment is initiated and needs to be signalled to the switches along the calculated path (in Step 8). Enforcing the path establishment in the switch consists of establishing the forwarding entry inside the switch. Afterwards the measurement process inside these switches starts. After the DC EE has also been notified of the new path, data transfer via the direct path can start in Step 10.

Note that for a real solution, the exact path calculation algorithms used for domain-internal path calculation must also be described in detail. The central path calculation and enforcement per domain has the benefit to reduce the complexity inside the network as the computation algorithm is realised and executed only once per domain. Moreover new upcoming protocols and new constraints in the path calculation algorithm can be integrated easily. Thereby the aggregate bandwidth consumption and thus cost and energy can be minimised. Policies can be effortlessly enforced as well from the central DCU node.

The control interfaces between the functional entities described in Figure 8.6 have the following functionalities:

- IE-IE for resource state updates from the network to the DCU
  - Measurement information like switch/router buffer occupancy of flows, packet counters of flows, time up of flows, data rates of flows, flows to the same target network (for flow aggregation), (Step 1)
  - New/adapt/delete node n with link l to node m via port x for topology discovery, ports available, port down, port up
  - New/delete/adapt link properties for link l, e.g. utilisation, delay, jitter, energy consumption per transported bit, for resource update, take up/down a link, e.g. a new optical transport link
- DE-DE for triggering of actions intra and inter network domains
  - Inter network provider information on what nodes n can be reached, issues on what information can be disclosed. E.g. a data-centre based DCU can communicate with a network based DCU using the UNI user-network interface for cross-domain control of flows and edge-to-edge optimum connectivity computations;
  - Triggering of (sub-)path computation from node n to node m (Step 3);
  - Execution of (sub-)path establishment based on a previous path computation (Step 7);
- DE-IE for requesting information or an action
  - Path computation request (Step 5)
- IE-DE for delivering requested information or action reply
  - Path computation reply (Step 6)
- DE-EE for enforcement of decisions
  - Establish forwarding entries in the switch/router client and controlling of forwarding (Step 8 in CN, Step 9 in DC) along a path
- Ext-DE for requests from external networking elements
  - External Path request, e.g. from an OMC towards a DCU

The kind of information we need for our algorithm can be summarised in the following aspects: Topology of the network (links and nodes, derived during start-up and/or learned during runtime), link properties like bandwidth, delay, jitter, energy consumption per bit, etc. In particular, the information needed by the decision process is summarised in Table 8-1.

Extensions that need further elaboration of the simplified architectural model above:

- Separation of domain border nodes and domain internal nodes
- Separation of 'path computation' phase (i.e., a sort of information collection) and 'path establishment' phase (i.e., the execution), e.g. for e.g., see the backwards-recursive path calculation in [RFC5441]; we have here for two cases:
    - Multi-DCU case for Multi-domain CN
    - Multi-DCU case for Multi-layer CN
- 'Management' functional phases like
    - DC discovering its DC neighbourhood/topology (in the overlay)
    - Changing the DC network topology (adding/deleting a DC node, advertising of new resources)
    - CN domain/layer discovering its CN neighbourhood domains/layer/topologies or advertising network resources
    - Changing DC network topology
- Address resolution and mapping in inter-layer communication: DC talking to other DCs in terms of "DC overlay address scope" -- request path from the CN in terms of CN address scope to the remote DC
- Abstraction mapping of physical topology information in inter-domain or inter-layer interfaces and communication

Table 8-1: Information needed for data-centre interconnection with DCU

Resources	Node resources	• End point (node) identifier • Capabilities (packet routing, p. switching, label switching etc) • Available ports • Energy consumption per port • Geo-position of nodes
	Link capabilities	• Connecting end points (nodes) • Capacity (bandwidth) • Delay • Utilisation • Energy consumption per link
	Network resources	• [List of] Network Topology (= connectivity of nodes and links) and their properties • [List of] Path/Sub-path and their properties
Requirements	Data-centre Application	• Type of QoS requirements is determined by the preferred policy
Policy	Strategy goals	• Minimise energy consumption of path • Maximise throughput • Minimise delay • Minimise jitter • White list (preferred) or black list (forbidden) of nodes/links/sub-networks/geo areas/operator domains

Finally, we can say that with this last use-case we mainly address the mechanisms on data-centre interconnection, as detailed in:

- WAN Interconnectivity of Distributed Data-Centres for Virtual Networks (Section 5.2.1)
- Control Functions for Multi-Layer Networking and Transport (Section 5.2.2)

9 Conclusion and Future Work

In this final chapter we recall the general context of this work and the adopted approach to attain our objectives, we summarise our main achievements and results obtained so far, we present a brief recap of the relevant future work which needs to be carried out in the subsequent deliverables, and we also attempt at providing a self-assessment for this period.

9.1 OConS Achievements for the First Year of SAIL

We have started our research work with a strong motivation to overcome the design and the performance limits imposed by the current networking architectures. Accordingly, we have first sketched a challenging Flash Crowd scenario, likely to be widely found in the Future Internet, and which posses a number of severe requirements on the networks and their connectivity services.

From the state-of-the-art, we have further seen that most of the current approaches and solutions for offering and managing the connectivity services (e.g., data-transport, routing, mobility, QoS) do not cope well with these requirements, which is due in a certain extent to the inherent guidelines and the principles they were built upon. We have thus dedicated a large part of our effort to discuss and challenge some of these guidelines and principles, and, whenever necessary, we have proposed changes and adaptations for them or even new guidelines to be followed by OConS.

Then, by applying these architectural guidelines (e.g., openness, modularisation, on-the-fly connectivity services, "path" concept for emulating the connections, self- and distributed control, seamless mobility), we have defined our OConS architectural framework; this can be considered as a component-based architecture made from elementary functional entities with clearly identified interfaces among them.

We have further proposed several connectivity mechanisms and services, as well as the approaches and the techniques envisaged for dealing with them; for example, we have defined novel Multi-P mechanisms for Multi-Path/Multi-Point/Multi-Protocol, imagined novel approaches for interconnecting the distributed Data-Centres, and proposed new Network Coding (NC) and Cross Layer techniques. In addition, a great amount of effort was necessary for the mapping of our diversified mechanisms onto the OConS architectural framework.

Moreover, we have developed novel network management paradigms that can interact with both legacy transport protocols and with our novel connectivity services. We have thus specifically investigated the Dynamic Distributed Mobility management, the Security & Mobility framework, Cognitive Radio with Spectrum Sensing, Channel and Radio resource allocation methods, Radio Resource management for WMNs, and Routing and Forwarding strategies in DTNs. Likewise, for each OConS mechanism, the needed information was clearly determined and hints on the possible protocols to be re-used/enriched have been also provided. Besides, these management functions and mechanisms were mapped onto our OConS architectural framework.

Finally, we have provided several examples of Use-Cases from the Flash Crowd scenario, thus showing how we are applying our framework on real cases, how the proposed OConS technologies are used, and what components are required to implement a given use-case. By supporting the proposed connectivity services, we reckon that the networks can more easily evolve than nowadays, OConS being, in our view, well suited to provide the necessary flexibility to cope with the demanding networking requirements, either present or forthcoming.

9.2 Prototyping and Experimentation

In parallel with the architecture specification and the initial design of the OConS mechanisms, protocols and algorithms, we have also carried out prototyping and experimentation activities, while planning these activities also in relation with the dedicated project-wide Theme.

The further architecture work is supported by prototyping, experimentation and demonstration activities in the following clusters in close relation to the OConS use cases:

- Cluster: Challenged Wireless Environment
  - Distributed management of multi-path connectivity in challenged networks: realised by a distributed routing/forwarding protocol for Delay/Disruptive Tolerant Networks based on self-learning and self-management techniques (for mobile resource-constraint devices)
- Cluster: End-to-end Multipath Transport
  - Multi-path support for content delivery in information centric networks: extending current ICN-like implementation with multipath transport mechanisms in wireless and wired network entities using multiple paths to carry content to required locations
  - End-to-end multi-path management for mobile transport: (incl. multi-congestion control) adapting to media content type for advanced apps e.g. web using HTML5; decision making (in a mobile device) for choice of congestion control, reliability, interface and wireless service provider to maximise use of available resources and QoE.
- Cluster: Mobile Connectivity in Access Networks
  - Smart resource management for mobile OConS access to realise a subset of the OConS architecture, interfaces and signalling (with more emphasis at the access part) for smart selection of mobile access and dynamic distributed mobility management (with dynamically allocated mobility anchoring in access routers and optimised mobile per-flow connectivity and forwarding functions related to the flash crowd scenario)
- Cluster: Edge-to-edge Connectivity in Core Networks
  - Managing interconnectivity of data-centres: including multi-layer control and protocols for path and flow optimisation between distributed data centres (CloNe or NetInf nodes) ; client-to-network and cross-layer interaction of IP/MPLS routing and optical switching (metro/core functionality) for management and control of virtualised networks between DCs; inter-provider connectivity for path and flow optimisation between distributed data centres based on OpenFlow concepts.

9.3 Self-Assessment

We have started in OConS with a bottom-up approach, where several specific connectivity services have been investigated by the involved partners. Nonetheless, the integration among these services and mechanisms needs further investigation, including the final design and full specification of: the orchestration functionality which coordinates them, the information exchanged on each identified interface to manipulate the corresponding abstractions (end-points, entities, paths), and the interconnection approaches envisaged among the OConS domains.

Moreover, we also need a top-down approach exemplifying with complete end-to-end/edge-to-edge OConS procedures such as: multi-path/multi-protocol connectivity, traffic steering and mobility and resource management in challenged networks, and autonomous data-centre interconnection; therefore, we need to further capitalise on our proposed architectural framework to show in greater detail how OConS deals with the respective procedures and service requests.

Likewise, by evaluating these procedures through analytical, simulation, and prototyping activities, the expected benefit of the OConS approach can be thus demonstrated through both the functional improvements (e.g., new capabilities, better performances, increased robustness, etc.) and the non-functional incorporated features (e.g., scalability, flexibility, manageability, and so on).

On the other hand, the OConS interactions with NetInf and CloNe needs also proper consideration to see what and how Connectivity Services can be offered, e.g.: multi-path/multi-protocol connectivity and transport with a given SLA, exposure/management/usage of certain networking resources and topologies (including the virtualised ones), notification/monitoring of networking resources and conditions, and naming and resolution services (e.g. Late Name/Locator Binding or DNS) for the OConS entities, end-points or paths.

9.4 Intended Future Work

This deliverable contains the first attempt towards the OConS architectural framework, including the connectivity services mechanisms and the management of these services, in accordance to our planning and the Description-of-Work.

Accordingly, the OConS architectural framework will be further refined in the subsequent deliverables, notably:

- The presented mechanisms, protocols and algorithms will be further specified, and the complete definition of interfaces among the OConS entities will be laid down in detail;
- The orchestration function, which is envisaged in our architecture to coordinate different connectivity services, needs to be further discussed and, subsequently, fully designed;
- Detailed and extensive analysis of some of the proposed algorithms and techniques will be carried out and reported, based on simulation and/or theoretical studies.
- The API to expose OConS functionalities will be detailed in a future deliverable (being too early to specify it now), and the necessary coordination with NetInf & CloNe will be further pursued;
- As the OConS approach is also intended to come with smooth migration paths from the existing solutions, including a possible standardisation of some interfaces and protocols, if required so;
- Selective connectivity services, from the ones presented in this deliverable, will be also prototyped and demonstrated.

References

[4WARD-D5.1] Guillemin F. (et al.), *D5.1 - Architecture of a Generic Path,* EU-FP7 4WARD project，Jan. 2009．

[4WARD-D5.2] Radriamasy S. (et al.), *D5.2 - Mechanisms for Generic Paths,* EU-FP7 4WARD project，May 2010．

[4WARD-D5.3] Woesner H. (et al.), *D5.3 - Evaluation of Generic Path Architecture and Mechanisms,* EU-FP7 4WARD project，June 2010．

[ARCHSTONE] ARCHSTONE, Project wiki website, http://archstone.east.isi.edu/twiki/bin/view/ARCHSTONE

[Becke10] Becke M. (et.al), *Load Sharing for the Stream Control Transmission Protocol (SCTP)*，IETF Internet-Draft (work in progress)，Dec. 2010．

[Benson10] Benson T., Akella A., Maltz D., *Network Traffic Characteristics of Data Centers in the Wild*． Internet Measurement Conference (IMC)，Melbourne, Australia，Nov. 2010．

[Chamania09] Chamania M., Jukan A., *A survey of inter-domain peering and provisioning solutions for the next generation optical networks*，IEEE Communications Surveys & Tutorials，March 2009．

[Cisco-OTV] Cisco, White Paper, *Overlay Transport Virtualization: Technology Introduction and Deployment Considerations*． Jan. 2011, http://www.cisco.com

[Clark09] D.D. Clark, *Toward the design of a Future Internet*． Report v.7，Oct. 2009．

[Dreibholz11] Dreibholz T., Becke M., *SCTP Socket API Extensions for Concurrent Multipath Transfer*，IETF Internet-Draft (work in progress)，May 2011．

[Dunbar10] Dunbar L., Hares S., *Scalable Address Resolution for Large Data Center Problem Statements*，IETF Internet-Draft (work in progress)，July 2010．

[FIArch11] FIArch-Group, *Fundamental Limitations of Current Internet and the path to Future Internet*． March 2011．

[Ford08] Ford B., Iyengar J., *Breaking up the transport logjam*，7th Workshop on Hot Topics in Networks (HotNets-VII)，2008．

[Ford11] Ford A., Raiciu C., Handley M., Bonaventure O., *TCP Extensions for Multipath Operation with Multiple Addresses*，IETF Internet-Draft (work in progress)，March 2011．

[G.800] ITU-T G.800 *Digital networks - General aspects, Title: Unified functional Architecture of transport networks*． Sept. 2007．

[G.805] ITU-T G.805 *Digital networks - General aspects, Title: Generic functional architecture of transport networks* ． March 2000．

[G.8080] ITU-T G.8080/Y.1304 *Packet over Transport aspects - Ethernet over Transport aspects, Title: Architecture for the automatically switched optical network (ASON)* ． June 2006．

[G.809] ITU-T G.809 *Digital networks - General aspects, Title: Functional architecture of connectionless layer networks*． March 2003．

[Grover10] Grover H., Rao D., Farinacci D. *Overlay Transport Virtualization*，IETF Internet-Draft (work in progress)，Oct. 2010．

[Han06] Han H. (et. al), *Multi-path TCP: a joint congestion control and routing scheme to exploit path diversity in the internet*，IEEE/ACM Transactions on Networking (TON), Vol. 14, Issue 6，2006．

[Huang08] Huang Y. (et. al), *TCP Performance in Coded Wireless Mesh Networks*．IEEE SECON．June 2008．pp.179–187．

[IEEE802.21] IEEE802.21 *Standard for Local and Metropolitan Area Networks: Media Independent Handover Services*．Nov. 2008．

[Iyengar06] Iyengar J.R., Amer P.D., Stewart R., *Concurrent Multipath Transfer Using SCTP Multihoming Over Independent End-to-End Paths*，IEEE/ACM Transactions on Networking, 14(5)，Oct. 2006．

[Johnsson06] Johnsson M. (et. al), *Ambient Networks: a framework for multi-access control in heterogeneous networks*，Proceedings of the 64th IEEE Vehicular Technology Conference, VTC，Sept. 2006．

[Kappler07] Kappler K. (et. al), *Dynamic network composition for beyond 3G networks: a 3GPP viewpoint*，IEEE Network, 21(1):47-52，Jan.-Feb. 2007．

[Katti06] Katti S. (et. al), *XORs in the Air: Practical Wireless Network Coding*．ACM SIGCOMM'06． Pisa, Italy，Sept. 2006． pp.243–254．

[Klein11] Klein D., Pries R., Menth M., Scharf M., Soellner M., *Modeling and Evaluation of Address Resolution Scalability in Data Center Interconnect Solutions*，submitted to ITC2011．

[Knight04] Knight P., Lewis C., *Layer 2 and 3 Virtual Private Networks: Taxonomy, Technology, and Standardization Efforts*，Communications Magazine, IEEE, vol. 42, no. 6, pp. 124 – 131，2004．

[Labovitz10] Labovitz C. (et. al), *Internet Inter-Domain Traffic*，in ACM SIGCOMM, New Delhi, India，Aug. 2010．

[MobilityFirst] MobilityFirst, Project website, http://mobilityfirst.winlab.rutgers.edu

[NEBULA] NEBULA, Project White Paper, http://nebula.cis.upenn.edu/NEBULA-WP.pdf (last checked on 20 June 2011)

[Nicira] Nicira-Networks, Company website, http://www.nicira.com/

[Niebert04] Niebert N. (et. al), *Ambient Networks: An architecture for communication networks beyond 3G*， IEEE Wireless Communications, 11(2)，April 2004．

[Niebert07] Niebert N., Schieder A., Zander J., Hancock R., *Ambient Networks: Co-operative Mobile Networking for the Wireless World*，Wiley，2007．

[OpenFlow] OpenFlow, Switching Reference System, http://www.openflowswitch.org/wp/downloads/

[OTV] Lapukhov P. *What is Overlay Transport Virtualization?* http://blog.ine.com/2010/02/15/what-is-overlay-transport-virtualization/

[Perlman04] Perlman R., *Rbridges: Transparent Routing*，IEEE INFOCOM，2004．

[Raiciu11] Raiciu C., Handley M., Wischik D., *Coupled Congestion Control for Multipath Transport Protocols*，IETF Internet-Draft (work in progress)，June 2011．

[REST] Wikipedia Representational State Transfer, http://en.wikipedia.org/wiki/Representational_State_Transfer

[RFC3439] Bush R., Meyer D., *Some Internet Architectural Guidelines and Philosophy*．IETF RFC 3439，Dec. 2002．

[RFC3473] Berger L. (Ed.), *Generalized Multi-Protocol Label Switching (GMPLS) Signaling Resource ReserVation Protocol-Traffic Engineering (RSVP-TE) Extensions*． IETF RFC 3473，Jan. 2003．

[RFC3654] Khosravi H., AndersonT., *Requirements for Separation of IP Control and Forwarding*． IETF RFC 3654，Nov. 2003．

[RFC3724] Kempf J., Austein R., *The Rise of the Middle and the Future of End-to-End: Reflections on the Evolution of the Internet Architecture*．IETF RFC 3724，March 2004 .

[RFC3945] Mannie E. (Ed.), *Generalized Multi-Protocol Label Switching (GMPLS) Architecture*． IETF RFC 3945，Oct. 2004．

[RFC4203] Kompella K., Rekhter Y., (Eds.), *OSPF Extensions in Support of Generalized Multi-Protocol Label Switching (GMPLS)*． IETF RFC 4203，Oct. 2005．

[RFC4204] Lang J. (Ed.), *Link Management Protocol (LMP),* IETF RFC 4204, Oct. 2005

[RFC4208] Swallow G., Drake J., Ishimatsu H., Rekhter Y., *Generalized Multiprotocol Label Switching (GMPLS) User-Network Interface (UNI): Resource ReserVation Protocol-Traffic Engineering (RSVP-TE) Support for the Overlay Model*． IETF RFC 4208，Oct. 2005．

[RFC4655] Farrel A., Vasseur J.-P., Ash J., *A Path Computation Element (PCE)-Based Architecture*． IETF RFC 4655，Aug. 2006．

[RFC4656] Shalunov S. (et al.), *A One-way Active Measurement Protocol (OWAMP)*．IETF RFC 4656，Sept. 2006．

[RFC4762] Lasserre M., Kompella V., *Virtual Private LAN Service (VPLS) Using Label Distribution Protocol (LDP) Signaling*．IETF RFC 4762，Jan. 2007．

[RFC4949] Shirey R., *Internet Security Glossary - v.2*, IETF RFC 4949，Aug. 2007．

[RFC5151] Farrel A. (Ed.), Ayyangar A., Vasseur J.-P., *Inter-Domain MPLS and GMPLS Traffic Engineering - Resource Reservation Protocol-Traffic Engineering (RSVP-TE) Extensions*． IETF RFC 5151，Feb. 2008．

[RFC5392] Chen M., Zhang R., Duan X., *OSPF Extensions in Support of Inter-Autonomous System (AS) MPLS and GMPLS Traffic Engineering*．IETF RFC 5392，Jan. 2009．

[RFC5441] Vasseur J.-P., Zhang R., Bitar N., LeRoux JL., *A backward recursive PCE-based computation (BRPC) procedure to compute shortest constrained interdomain traffic engineering label switched paths*． IETF RFC 5441，Apr. 2009．

[RFC5556] Touch J., Perlman R., *Transparent Interconnection of Lots of Links (TRILL): Problem and Applicability Statement*．IETF RFC 5556，May 2009．

[RFC5810] Doria A., Hadi-Salim S., Haas R., Khosravi H., Wang W., (Eds.), *Forwarding and Control Element Separation (ForCES) Protocol Specification*．IETF RFC 5810，March 2010．

[RFC5921] M.Bocci (Ed.), *A Framework for MPLS in Transport Networks*．IETF RFC 5921，July 2010．

[RFC6181] Bagnulo M., *Threat Analysis for TCP Extensions for Multipath Operation with Multiple Addresses*，IETF RFC 6181，March 2011．

[RFC6182] Ford A., Raiciu C., Barre S., Iyengar J., *Architectural Guidelines for Multipath TCP Development*，IETF RFC 6182，March 2011．

[Sachs06] Sachs J. (et. al), *Migration of existing access networks towards multi-radio access*，Proceedings of the 64th IEEE VTC，Sept. 2006．

[Sachs07] Sachs J. (et. al), *Generic abstraction of access performance and resources for multi-radio access management*，Proceedings of the 16th IST Mobile and Wireless Communications Summit，July 2007．

[SAIL-D.A.1] SAIL Project, *Description of Project-wide Scenarios and Use-Case*, EU-FP7-ICT-2009-5-257448-SAIL/D.A.1, Feb. 2011．

[SAIL-D.A.2] SAIL Project, *Draft Architecture Guidelines and Principles*, EU-FP7-ICT-2009-5-257448-SAIL/D.A.2, July 2011．

[SAIL-D.B.1] SAIL Project, *Network of Information Architecture and Applications*, EU-FP7-ICT-2009-5-257448-SAIL/D.B.1, July 2011．

[SAIL-D.D.1] SAIL Project, *Cloud Networking Architecture Description*, EU-FP7-ICT-2009-5-257448-SAIL/D.D.1，July 2011．

[Schieder07] Schieder A. (et. al), *The reference points of an Ambient Network*，Proceedings of the 16th IST Mobile and Wireless Communications Summit，July 2007．

[STR-D3.2] STRONGEST project *Next generation transport networks: efficient solutions for OAM, control, and traffic admittance*．Deliverable D3.2，Dec. 2010．

[TR23.401] 3GPP TR23.401 *Technical Specification Group Services and System Aspects; General Packet Radio Service (GPRS) enhancements for Evolved Universal Terrestrial Radio Access Network (E-UTRAN) access (Release 9)*．March 2011．

[TR23.402] 3GPP TR23.402 *Technical Specification Group Services and System Aspects; Architecture enhancements for non-3GPP accesses (Release 9)*．March 2011

[TUPN] Wikipedia, List of TCP and UDP port numbers, http://en.wikipedia.org/wiki/List_of_TCP_and_UDP_port_numbers

[X.901] ITU-T X.901, *Information technology – Open distributed processing – Reference Model: Overview*．Aug. 1997．

[X.902] ITU-TX.902, *Information technology – Open Distributed Processing – Reference Model: Foundations*．Oct. 2009．

[X.903] ITU-T X.903 *Information technology – Open distributed processing – Reference Model: Architecture*．Oct. 2009．

[X.904] ITU-T X.904 *Information technology – Open Distributed Processing – Reference Model: Architectural semantics, Amendment 1: Computational formalisation*．March 2000．