Deliverable D4.2

Synthesis of use case requirements

Release 2

Grant Agreement	257513	
Date of Annex I	25-07-2011	
Dissemination Level	Public	
Nature	Report	
Work package	WP4 – Deployment and Impacts	
Due delivery date	01 March 2012	
Actual delivery date	13 April 2012	
Lead beneficiary	TIS	Antonio Manzalini Antonio.Manzalini@telecomitalia.it

SEVENTH FRAMEWORK PROGRAMME

FUTURE NETWORKS

Authors	TIS, Antonio Manzalini (Editor) ALBLF, Leila Bennacer, Laurent Ciavaglia, Samir Ghamri-Doudane, Pierre Peloso, Benoit Ronot ALUD, Markus Gruber TCS, Mathieu Bouet and Gerard Nguengang NEC, Johannes Lessmann and Zarrar Yousaf FT, Christian Destré, Imen Grida Ben Yahia, Zwi Altman TID, Beatriz Fuentes IBBT, Sander Spek, Vânia Gonçalves and Simon Delaere INRIA, Olivier Festor and Remi Badonnel VTT, Teemu Rautio and Jukka Mäkelä UCL, Alex Galis, Stuart Clayman UniS, Stylianos Georgoulas UT, Ramin Sadre, Ricardo Schmidt and Anna Sperotto NKUA, Nancy Alonistioti, Eleni Patouni, George Katsikas, Panagiotis Spapis, Apostolis Kousaridas, Konstantinos Chatzikokolakis, Vagelis Kosmatos and Evangelos Rekkas UPRC, Panagiotis Demestichas, Kostas Tsagkaris, Andreas Georgakopoulos, Vaggelis Thomatos, Dimitrios Karvounas, Nikolaos Koutsouris, Asimina Sarli, Panagiotis Vlacheas

Executive summary

Deliverable D4.2 provides a consolidated synthesis of the requirements related to the use cases defined in WP4 "Deployment and Impact". Specifically, the preliminary requirements (initially as reported in the report D4.1 [1]) have been refined, in some cases extended, and prioritized by using the Quality Function Deployment (QFD) methodology [2]-[9]. Deliverable D4.2 is also reporting the lessons learnt in eliciting the future networks management requirements, the internal project use (mainly towards the work of WP2, WP3 and other tasks of WP4) and external exploitation, mainly towards the standardization activities.

Table of Content

Foreword **5**
1 Introduction **6**
2 Synthesis of UC's requirements **7**
2.1 Synthesis of use cases business requirements **7**
2.1.1 Use Case 1 7
2.1.2 Use Case 2 8
2.1.3 Use Case 3 9
2.1.4 Use Case 4 10
2.1.5 Use Case 6 11
2.1.6 Use Case 7 13
2.2 Synthesis of use case functional and non-functional requirements **15**
2.2.1 Synthesis of use case functional requirements 15
2.2.2 Synthesis of use cases non-functional requirements 24
3 QFD Analysis **28**
3.1 Results of QFD analysis **30**
3.1.1 Use Case 1 30
3.1.2 Use Case 2 32
3.1.3 Use Case 3 35
3.1.4 Use Case 4 38
3.1.5 Use Case 6 39
3.1.6 Use Case 7 44
4 Exploitation **47**
4.1 Project Internal Use **47**
4.1.1 Towards WP2 47
4.1.2 Towards WP3 47
4.1.3 Towards WP4 – Task 4.3 47
4.2 External Exploitation **50**
5 Conclusion **52**
References **53**
Abbreviations **54**
Definitions **56**

Foreword

Deliverable D4.1 [1] provided a preliminary list of the requirements collected via the use-cases defined in the work package 4 (Deployment and Impacts) of the UniverSelf project. Deliverable D4.2 is providing a consolidated synthesis of said requirements (see also intermediate Milestones MS35 and MS39). Specifically, the initial requirements (as reported in deliverable D4.1) have been refined, in some cases extended, and prioritized by using the Quality Function Deployment (QFD) methodology [2]-[9]. Deliverable D4.2 is also reporting the lessons learnt in deriving the requirements and the related internal (mainly towards WP2, WP3 and other tasks of WP4) and external exploitations (mainly towards standardization activities).

Actually, according to the project lifecycle, the derived requirements have been transferred to WP2 to guide the design of the Unified Management Framework (UMF); to WP3 to direct the design of the methods and algorithms of the Network Empowerment solutions; and to other tasks of WP4 for implementation and validation of the integrated solutions, business impact analysis and trust development and evaluation.

This approach has ensured integration between WPs while maintaining a systemic perspective.

The scope of Deliverables D4.1 and D4.2, as per the Description of Work, is as follows:

D4.1 – Synthesis of use case requirements – release 1: this document will represent the first report on the derivation of technical requirements for use cases (outcome of task 4.1). Specifically, it will propose initial results concerning the definition of key use cases for the three identified scenarios and the related requirements (at the system, functional and business level).

D4.2 – Synthesis of use case requirements – release 2: This document will represent the final report on the derivation of technical requirements for relevant use cases (outcome of task 4.1). Specifically, it will provide further results on the definition of key use cases for the three identified scenarios and the related requirements (at the system, functional and business level).

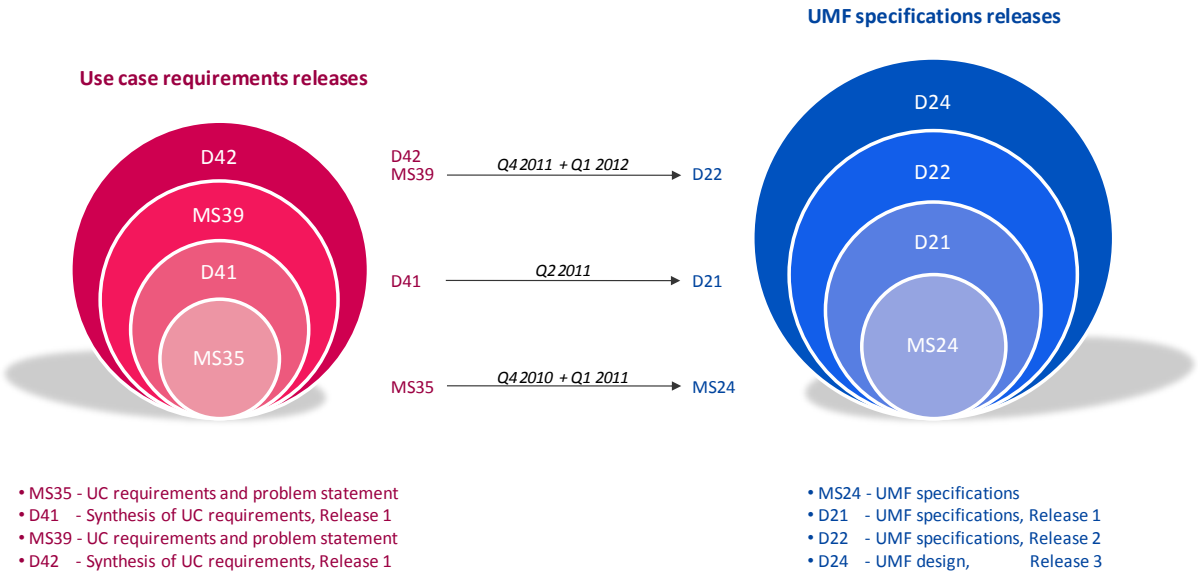

Figure 1 – Documentation roadmap - Synchronization points between use case requirement releases (WP4) and UMF specifications releases (WP2).

1 Introduction

This is the second deliverable (the former one is deliverable D41 [1]) of WP4 (Task 4.1) activity (see also intermediate Milestones MS35 and MS39) about the definition of functional, non-functional and business requirements of selected use cases.

Use-cases definition started from the three reference scenarios (identified during the proposal preparation), i.e. (1) Operators' Service and Data Management, (2) SON for Radio Access and Core Networks and (3) Future Internet Services Management and Network Resource Optimization. These scenarios are reflecting the Network Operators needs of reducing OPEX and improving the return on investment of network equipment and infrastructures. The problems and requirements of the use case 5 and use case 6 as identified in deliverable D4.1 were merged into one revised use case 6.

The consolidated list of requirements has been elaborated by setting a prioritisation of use case problems and related requirements. In order to achieve this objective, Quality Function Deployment (QFD) [2]-[9] has been adopted as a working systematic approach.

The Deliverable is structured in three sections:

- Section 2 reports a synthesis of the consolidated list of business, functional and non-functional requirements;
- Section 3 briefly describes the QFD methodology adopted for setting a prioritisation of use case problems and related requirements; the main results of the prioritisation are also reported;
- Section 4 elaborates some considerations on how this process of use-case requirements derivation and analysis is being exploited within the project activities and in relevant standardization bodies.

2 Synthesis of UC's requirements

This section presents the consolidated list of business, functional and non-functional requirements that have been derived from the set of use cases.

Requirements are named as follows:

- Req_B_x.y is denoted as the *y* Business requirement of use case *x (cf. last section)*,
- Req_F_x.y is denoted as the *y* Functional requirement of use case *x*,
- Req_NF_x.y is denoted as the *y* Non-functional requirement of use case x.

The language adopted for the formulation of the requirements was based on the IETF / ITU-T respective approach. More specifically, the key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in IETF RFC 2119 [10]. Specifically:

- MUST: This word, or the terms "REQUIRED" or "SHALL", mean that the definition is an absolute requirement of the specification.
- MUST NOT: This phrase, or the phrase "SHALL NOT", means that the definition is an absolute prohibition of the specification.
- SHOULD: This word, or the adjective "RECOMMENDED", mean that there may exist valid reasons in particular circumstances to ignore a particular item, but the full implications must be understood and carefully weighed before choosing a different course.
- SHOULD NOT: This phrase, or the phrase "NOT RECOMMENDED" mean that there may exist valid reasons in particular circumstances when the particular behaviour is acceptable or even useful, but the full implications should be understood and the case carefully weighed before implementing any behaviour described with this label.
- MAY: This word, or the adjective "OPTIONAL", means that an item is truly optional.

2.1 Synthesis of use cases business requirements

2.1.1 Use Case 1

There is no update (with respect to deliverable D4.1) to the list of business requirements.

Delivering rich QoS to end-users can take different forms. Realizing self-diagnosis and self-healing can provide new possibilities and means for the network operator/service provider to improve its service offer and quality. For instance, one of the goals of UC1 is to improve diagnosis capabilities, which should enable faster isolation of the failure(s), and mitigate the impact (and even predict the failure and avoid the detrimental effects). Such technological advance should raise the end-user offered QoS by lower Time-to-Repair (and possibly guided/reparation information displayed on the TV screen or mobile device) and thus should lead to increased service availability; it should also reduce the cost per service per customer with fewer calls to the help desk. A unified diagnosis solution which is applicable to multiple IMS services (semantic approach) and to diverse technological domains (wireline/wireless, service platform) should also bring economy of scale to the operator, which should add to the OPEX gains (single/shared solution versus services/networks) for service assurance.

The main business requirements for UC1 are summarized in the following table:

ID	Name
Req_B_1.1	Reduction of time to repair errors/problems
The UMF should enable to decrease the time required to repair problems in the network (this is reducing OPEX).	
ID	**Name**
Req_B_1.2	Reduction of cost per service/per customer
The UMF should enable to decrease the cost per service per customer of repairing problems in the network e.g. with fewer calls to the help desk and less required time to repair (this is reducing OPEX).	
ID	**Name**
Req_B_1.3	Increase of efficiency of operating the network and handling errors
The UMF should enable to improve the overall efficiency of network operation as well as handling errors.	
ID	**Name**
Req_B_1.4	Increase of service availability
The UMF should enable to increase service availability by reducing the errors or problems that occur in the network and by reducing the time required to repair such errors and problems.	
ID	**Name**
Req_B_1.5	Reduction of churn rate
The UMF may enable to reduce the churn rate by increasing the QoE.	

2.1.2 Use Case 2

There is no update (with respect to D4.1) to the list of business requirements.

Dimensions, dynamicity and complexity of today's networks are growing continuously. Controlling and managing dynamic network behaviour in order to meet technical and business objectives is becoming more and more complicated and challenging.

In this context assuring network stability and performance is becoming more and more important and strategic for Network Operators: actually, instability in communication networks may have primary effects both jeopardizing the network performance and compromising an optimized use of resources. These problems are likely to exacerbate in the future, when it is expected that the networks will become highly dynamic and pervasive, capable of interconnecting large numbers of real and virtual resources (e.g. routers, switches, transport nodes, servers, etc), Users' devices (e.g. smart phones, etc) and machines (e.g. sensors, smart things, etc).

It is reasonable to argue that introducing autonomic functions (in terms of optimization methods and control loops) in management and control should improve overall efficiency and should reduce mistakes; on the other hand overall stability of network behaviour has to be off-line validated and on-line monitored and controlled: as a matter of fact, cascading and nesting of self-* mechanisms can lead to the emergence of non-linear network behaviours (e.g. at the basis of network phase-transitions). This will imply the potential existence of multiple phases in network behaviours (i.e. identical local dynamic can give rise to widely different global dynamics) and state/phase transitions might occur (and maybe also due to self-organized criticalities).

The overall goal of UC2 is validating the existence of network stable states (with the related levels of performance) and driving network dynamics (in case of whatever perturbation) to said desired states (e.g. by looking at the network phase space1). In this sense adaptive features for network stabilization should be designed and exploited: given certain Operations and business objectives and constraints, we should assure that the network will be able to converge (whatever internal – external perturbations), within given certain time requirements, to stable desired state(s), characterized by target performance levels.

[1] The dimension of a phase space is the number of variables needed to describe the state of the network as it evolves in time. This number is also called the dimensionality of the network (as a dynamical system).

The main business requirements for UC2 are summarized in the following table:

ID	Name
Req_B_2.1	Costs savings due to reduction of configuration mistakes (before provisioning)
Off-line validation of the effects of interacting self-* features (e.g. Self Organizing Entities) before their activation will allow reducing configuration mistakes and network instabilities. This will determine CAPEX and OPEX reductions.	
ID	**Name**
Req_B_2.2	Costs savings due to reduction of unexpected network instabilities (during operations)
On-line validation of the effects of interacting self-* features (e.g. Self Organizing Entities) will allow reducing unexpected network instabilities. This will determine CAPEX and OPEX reductions.	
ID	**Name**
Req_B_2.3	New revenues from "network and service stability" as a service
Assuring network and service stability (with different levels) in an open context (e.g. against diverse internal and external roots of instabilities) can be seen as new advanced service provided by Operators, thus improving competitiveness: this can be considered potentially as a source of new revenues.	

2.1.3 Use Case 3

Today, most of network management occurs at the core and users are increasingly communicating with the mobile operator's data centres for their services and application requirements. Hence user traffic has to go through the core/backbone/backhaul resulting in consumption of resources as well as making the service dispensation susceptible to various performance impeding bottlenecks. The effective dispensation of services and efficient utilization of resources becomes all the more important for delay/error sensitive and bandwidth intensive mobile applications such as mobile video traffic (streaming and broadcast). The ubiquitous provisioning of such applications is putting a lot of pressure on mobile network operators and their respective infrastructures (especially the core and backhaul), and making the task of efficiently dimensioning of their networks an increasingly difficult and expensive task.

In view of this scenario, the main aim of UC3 is to enable the dynamic (on-the-fly and on-demand) realization of services/functions/gateways nearer to the user by leveraging the cloud computing and virtualization techniques to provide true NaaS (Network as a Service). This shall cause most of the user traffic and service demands to get negotiated and managed nearer to the user without having to traverse the core/backbone. This reduction, or shifting, of load from the core/backbone shall allow the operator to oversubscribe its network resources (bandwidth, processing. storage etc.) and also enhance the utilization of the existing resources in the backhaul and access. This should translate into increased ROI and revenue base while seeing a corresponding reduction in CAPEX. Additionally, slices of the virtualized operator network infrastructure may potentially be made available via specific interfaces to 3rd party service providers so that these may optimize their services by taking advanced network and user information from the operator into account. This may also open a potentially new revenue stream for the operator. Finally, UC3 addresses the need to facilitate new service deployments (and therefore the time to market) even for new services from the operator itself in that the operator may be using his own infrastructure cloud and the associated management platform, which has been designed anyway for easy service deployability.

The main business requirements for UC3 are summarized in the following table:

ID	Name
Req_B_3.1	Increase of ROI
The UMF should lead to the increase of ROI through the reduction/shifting of load from the core/backbone that shall allow the operator to oversubscribe its network resources (bandwidth, processing, storage etc.) and also enhance the utilization of the resources in the backhaul and access (this will determine CAPEX reductions).	
ID	**Name**
Req_B_3.2	Increase of revenue base
The UMF should lead to an increase of the revenue by opening new revenue streams and business models for the operator (e.g. by making available slices of the virtualized operator network via specific interfaces to 3rd party services). Enabling multi-tenancy on even mobile core network equipment is also a revenue possibility for infrastructure owners.	
ID	**Name**
Req_B_3.3	Reduction of churn rate
The UMF may enable to reduce the churn rate by increasing the performance of the network.	
ID	**Name**
Req_B_3.4	Increase of the efficiency of deploying new services
The UMF should enable to facilitate new service deployments.	
ID	**Name**
Req_B_3.5	Decrease of the time required to market (deploy) new services
The UMF should enable to decrease the time to market of new services.	

2.1.4 Use Case 4

Use case 4, "SON and SON collaboration according to operator policies" can be seen as the UMF instantiation for the radio access network segment. LTE and future LTE-A networks will be empowered by self-organizing network (SON) mechanisms that aim at simplifying network management, reduce its cost of operation and increase its performance. According to the SON paradigm, performance gains can be achieved by adapting the network to traffic variations and to conditions of operation. The studies during the first burst of UC4 have provided understanding of different elements needed to design a network empowered by coordinated SON functionalities. These elements are summarized presently and feed the list of the UC requirements.

(i) Control plane solution

Self-optimization functionalities consist of different control / optimization algorithms which control network resources (resource allocation, interference and mobility management). By embedding SON functionalities within the control plane (e.g. in base stations), better reactivity and hence performance gains are achieved. Hence control plane solutions for SON are desired, however, this raises the question of how can the operator maintain its control over the network.

(ii) Governance

To keep control of the network, the operator needs means of governance. Governance provides policies (including rules) for operating the network empowered by the SON functionalities. Via the governance tool, the operator introduces policies, KPI targets, optimization objectives, and other information that allows efficient operation of SON entities. The policies translate operational, performance and business objectives into rules and objectives to the SON entities.

(iii) Coordination

The network can contain a very large number of SON entities. These can have the same or different (conflicting) objectives, and can act upon the same or on different parameters. Coordination of the SON functionalities is clearly an essential element in the design of SON networks. UC4 studies, during the first burst, have shown that SON functionalities can be classified into different cases, according to their mutual impact and the time scale of operation:

1. No coupling between SON functionalities, which can operate independently without mutual performance degradation

2. Loose coupling between SON functionalities, which operate in different time scales and can thus be hierarchically separated
3. Tight coupling of SON functionalities, which operate in the same time scale. These functionalities can be processed together according to policies provided by the operator through the governance interface.

(iv) Performance

To allow the operator to benefit from the SON technology and its large-scale deployment, the SON solutions should be scalable, stable and robust (namely performance degradation during the learning or optimization process should be prohibited or limited to a pre-defined threshold).

The main business requirements for UC4 are summarized in the following table:

ID	Name
Req_B_4.1	Increase of revenues due to improvement of network performance
The UMF should enable to improve capacity, coverage and QoS performance. This will make the network more competitive from a business viewpoint determining an increase of revenues	
ID	**Name**
Req_B_4.2	Improvement of operator ranking
The UMF may enable to improve the operator ranking and consequently increase its competitiveness by improving QoS and QoE provisioning	
ID	**Name**
Req_B_4.3	Reduction of CAPEX
The UMF should reduce CAPEX by delaying or reducing additional infrastructure investments	
ID	**Name**
Req_B_4.4	Reduction of OPEX
The UMF should facilitate the decrease of (additional) efforts related to network operation activity of LTE and LTE-Advanced technology to configure, parameterize and optimize the network. Thus UMF SON mechanisms should lead to OPEX reduction.	
ID	**Name**
Req_B_4.5	Reduction of churn rate
The UMF may enable to reduce the churn rate by improving QoS and QoE for the network users.	
ID	**Name**
Req_B_4.6	Migration to legacy systems
Migration of legacy systems towards UMF take into account economic aspects (e.g. limited impact of UMF cost-of-features and the other costs related to its introduction)	

2.1.5 Use Case 6

Network Operators want to introduce new load (specific services and user classes) to the network. As an example, this may correspond to a music festival that is organized at the congress hall of Piraeus and some of the attendees would like to share the event in real time with their friends and family, in various locations, using real time types of application e.g. RTE (Real Time Experience – Requires end-to-end connection between the smart phones of both the video stream emitter and the receiver) or Mobile-TV.

For tackling such a situation today, operators would rely on processes that are not as flexible as they need and can be, and therefore, they impose costs. In general, the solution of the problem relies on (in very high-level terms): (i) planning and deployment (rollout); (ii) optimization and maintenance.

On the one hand, planning is an essential phase in the engineering of telecommunication systems. Nevertheless, telecommunication systems face changing situations, due to the time variant traffic demand, the occurrence of faults, mobility and radio conditions, in case of wireless access. As a result, handling all the potential situations, based only on planning, means that the worst (most demanding) case has to be considered as the reference one, according to which the network has to be planned. This leads to over-provisioning of resources (e.g. elements, bandwidth, etc.), which negatively impacts the cost (capital expenditures – CAPEX).

Adapting the network to the encountered situation, through management functionality, is the solution to this problem.

On the other hand, management relies on processes that are elaborate but not fully automated, as they have to deal with heterogeneous technologies, which are not adequately integrated. More specifically, the management processes/systems of an operator will typically adhere to specific standards. In general, these systems are heterogeneous, depending on the technology and on the vendor of the technology. This means that, in principle, the management systems of a wireless and wireline access technology will be different. Moreover, the management systems for a specific technology, obtained by two different vendors, will be different. The heterogeneity means that there is little or no integration between the management processes/systems. This negatively impacts the time required for (re-)configuring the infrastructure. Moreover, it means that human intervention is required in the process that leads to cross-technology configurations. This can cause, apart from delays, errors and inconsistencies. Finally, loose or no integration means that the information available to the different systems cannot be readily exploited for the purpose of optimizing the operation of the infrastructure.

In addition, service management and customer relation management also rely on processes that are not fully automated. Several aspects (phases of the overall process) often require manual intervention and/or the use of heterogeneous systems that are not integrated. This increases the cost of managing the customer relations.

Last but not least, manual configuration of network devices that requires strong technical expertise of at least one specialist by network segment is a standard situation that leads to an increase in the OPEX.

The use case aims at finding solutions that will alleviate the above-described problems by automating the involved processes and in particular by:

- Enabling operators to describe their goals and objectives, through high-level means and govern their network
- Achieving policy-based operation of RAN (OFDMA-based) and Backhaul/Core Network (IP/MPLS-based) segments, which is optimized with respect to QoE/QoS efficiency, taking into account metrics and knowledge derived in network nodes and end-user devices and in line with the operator objectives
- Achieving coherence between these segments through cooperation, negotiation and federation.

From the above-described targets of Network Operators and of the use case the following 8 UMF business requirements have been defined (as reported in D4.1):

The main business requirements for UC6 are summarized in the following table:

ID	Name
Req_B_6.1	Reduction of human intervention
The UMF should enable to reduce the need for human intervention.	
ID	**Name**
Req_B_6.2	Reduction of human error ratio
The UMF should enable to reduce the ratio of errors that occur due to human intervention.	
ID	**Name**
Req_B_6.3	Reduction of time required for service/network configuration
The UMF should enable to reduce the time required for deciding and enforcing changes in the configuration of services and the network.	
ID	**Name**
Req_B_6.4	Increase of the efficiency of deploying new services
The UMF should enable to facilitate new service deployments.	
ID	**Name**
Req_B_6.5	Decrease of the time required to market (deploy) new services
The UMF should enable to decrease the time required to market for new services.	
ID	**Name**
Req_B_6.6	Reduction of churn rate
The UMF may enable to reduce the churn rate by increasing the QoE.	
ID	**Name**
Req_B_6.7	Reduction of CAPEX
The UMF should enable to facilitate CAPEX reductions by enabling optimal utilization of resources and delaying or reducing	

additional investments in the infrastructure.	
ID	**Name**
Req_B_6.8	Reduction of OPEX
The UMF should enable to reduce OPEX by reducing the need for human intervention through autonomic self-management procedures and cognitive "traffic" engineering decisions.	

2.1.6 Use Case 7

Telco operators face the increasing challenge of providing higher levels of broadband access to more demanding customers. Nevertheless, the rapid bandwidth growth in the last years has failed to boost the Average Revenue Per User (ARPU). Telco operators have the need to adapt their operations in order to reduce the time to market and the network maintenance costs, while at the same time increasing the customer satisfaction. The management of the physical network infrastructure to enable high-quality new services is an increasingly critical part of the operational processes. Autonomic networks with self-configuration, self-diagnosis and self-healing capabilities will help in the automation of the provisioning and runtime phases, maintaining the quality of the services committed to the customer with minimal human intervention.

The improvements described above should be accompanied by a transformation in the business definition of services and the actual deployment at the network level. The agreement between a client and a service provider is expressed in the form of SLAs (Service Level Agreements), including client requirements as well as provider's assurances. These objectives are expressed in a high-level, service-, or application-specific manner, but should be translated to the low-level, resource specific language of the network elements. Current (semi-) manual practices must be minimized as much as possible, as they always imply certain delay in the delivery of new services. Furthermore, they require highly specialized technicians for the management of the network.

Network and Service Governance use case focuses on mechanisms that address the gap between high-level specification of client performance objectives and existing resource management infrastructures, but also on mechanisms to ensure the trustworthiness of the autonomic infrastructure. Such mechanisms should provide operators with means for decision oriented operational tasks based on the use of policies rather than low level command execution, thus decreasing the human intervention required for deploying new services, configuring and operating the network. This should lead to reduction of time to market as well as OPEX. Furthermore, network and service governance should enable improved QoS and consequently should lead to reduced churn rate and potentially increased revenues.

UC7 aims to demonstrate the feasibility of a policy-based approach for the management of two different types of networks: ADSL wireless access and fixed Fiber-To-The-Home (FTTH). The selection of FTTH is due to the fact that it is considered the network of the future in fixed access segment, able to support ultra-broadband speeds. Indeed, telecommunication operators are making huge investment efforts in fiber-to-the-home deployments. The goal of the use case is to provide a service assurance solution for both wireless and fixed FTTH environments, providing the network elements with self-monitoring, self-diagnosis and self-healing capabilities. All of them governed by means of high-level policies expressed in a friendly language, guaranteeing an efficient management of services and infrastructure.

The main business requirements for UC7 are summarized in the following table:

ID	Name
Req_B_7.1	Reduction of human intervention
The UMF should enable to decrease the human intervention required for the deployment, configuration and operation of new services on FTTH networks	
ID	**Name**
Req_B_7.2	Reduction of high specialized personnel in maintenance tasks
The UMF should enable to decrease the need for high specialized technicians for pure maintenance of the network that could be easily managed and supervised through the Governance framework.	
ID	Name
Req_B_7.3	Reduction of time required for service/network configuration
The UMF should enable to decrease the time required for the deployment, configuration and operation of new services on FTTH networks.	
ID	**Name**
Req_B_7.4	Reduction of the downtime of a service
The UMF should enable to decrease the service downtime, thanks to the proactive self-monitoring, self-diagnosis and self-healing capabilities.	
ID	**Name**
Req_B_7.5	Reduction of OPEX
The UMF should enable to reduce OPEX by reducing the need for highly specialized technicians through autonomic self-management procedures applied to FTTH network elements.	
ID	**Name**
Req_B_7.6	Increase of the efficiency of deploying new services
The UMF should enable to facilitate new service deployments through the use of the Network Governance Framework, which allows the automatic translation from high-level business requirements to network policies.	
ID	**Name**
Req_B_7.7	Decrease of the time required to market (deploy) new services
The UMF should enable to decrease the time required to market for new services, through the use of the Network Governance Framework, which allows the automatic translation from high-level business requirements to network policies.	
ID	**Name**
Req_B_7.8	Reduction of churn rate
The UMF may enable to reduce the churn rate by increasing the QoE of the FTTH customers.	

2.2 Synthesis of use case functional and non-functional requirements

2.2.1 Synthesis of use case functional requirements

This section analyses the UC functional requirements, as derived from the analysis of the UniverSelf use cases.

2.2.1.1 Use Case 1

The redundant Req_F1.13 is removed from the list of requirements defined in deliverable D4.1. There is no other update.

ID	Name
Req_F_1.1	Access mechanism to raw Service data
UMF shall support access mechanisms to any raw service data (Performance indicators, Services alarms, Services configuration, Services semantic, Services messages).	
ID	**Name**
Req_F_1.2	Access mechanism to raw Network data
UMF shall support access mechanisms to any raw network data (Alarms, Protocols configuration, Protocols semantic, Protocols messages, Hardware parameters, Performance indicators).	
ID	**Name**
Req_F_1.3	Elaborated Data/Context Management
UMF shall give a method to access context database for obtaining/storing/updating context information.	
ID	**Name**
Req_F_1.4	Knowledge Data Management
UMF shall give a method to access knowledge base for obtaining/storing/updating knowledge information.	
ID	**Name**
Req_F_1.5	Topology information Monitoring
UMF shall support method for monitoring any topology information data.	
ID	**Name**
Req_F_1.6	Contextual data translation
UMF shall support method for the translation of contextual data (upper layers to lower layers in the hierarchy).	
ID	**Name**
Req_F_1.7	Contextual data filtering/pre-processing
UMF shall support method to perform filtering and pre-processing to contextual data (lower layers to upper layers in the hierarchy).	
ID	**Name**
Req_F_1.8	Diagnosis Embodiment
UMF must enable embodiment of the proactive diagnosis mechanism.	
ID	**Name**
Req_F_1.9	Root cause analysis from alarms
UMF compliant system should be able to pinpoint the root cause among many alarms and identify the problems that need to be fixed	
ID	**Name**
Req_F_1.10	Horizontal data correlation
UMF must support method to correlate Intra/Inter-Domain data.	
ID	**Name**
Req_F_1.11	Vertical data correlation
UMF must support method to enable Cross-Technological and Cross-Organizational correlation.	
ID	**Name**
Req_F_1.12	Time scale data correlation
UMF must support method to enable a Time scale data correlation.	
ID	**Name**
Req_F_1.14	Model for Normality Prediction pattern data exchange
UMF should support model to exchange prediction pattern.	
ID	**Name**
Req_F_1.15	Model for Anomaly diagnosis exchange
UMF must support model to exchange anomaly detection.	

ID	Name
Req_F_1.16	Model for Normality diagnosis exchange
UMF should support model to exchange Normality detection.	
ID	**Name**
Req_F_1.17	Model for Complex data
UMF should support Key Performance Indicator models.	
ID	**Name**
Req_F_1.18	Model for Complex data exchange
UMF should support models to exchange KPI.	
ID	**Name**
Req_F_1.19	Data sharing exchange methods
UMF must provide data sharing methods between self-x enabling elements.	
ID	**Name**
Req_F_1.20	Mitigation policies exchange
UMF should support mitigation policies for each predicted event.	
ID	**Name**
Req_F_1.21	Interface with NMS
UMF shall provide Interface with NMS for upstream data exchange (raw data coming from monitoring, event reporting, evaluation of the system after a mitigation/ reparation plan) and downstream data exchange (high level data monitoring such as KPI or aggregated data, data to be monitored and re-configuration actions).	
ID	**Name**
Req_F_1.22	Predicted event reporting to Human
UMF should provide Self-x enabling elements and Network (NMS) to human interfaces (OSS) for event detection reporting.	
ID	**Name**
Req_F_1.23	Predicted event reporting to Network
UMF must provide Self-x enabling elements to Network entities (NMS) interfaces to share detection reports.	
ID	**Name**
Req_F_1.24	Predicted event reporting to Self-x enabling element
UMF must provide interfaces to share detection reports between Self-X enabling elements	
ID	**Name**
Req_F_1.25	Triggered Mitigation reporting to Human
UMF should provide Self-x enabling elements and Network (NMS) to human(OSS) interface for mitigation triggering reporting	
ID	**Name**
Req_F_1.26	Triggered Mitigation reporting to Network
UMF must provide Self-x enabling elements to Network (NMS) interface for mitigation triggering reporting.	
ID	**Name**
Req_F_1.27	Triggered Mitigation reporting to Self-x enabling element
UMF must provide interfaces to share mitigation triggering reports between self-x enabling elements.	
ID	**Name**
Req_F_1.28	Event prediction algorithms Coordination
UMF must support algorithm coordination between multiple event prediction nodes.	
ID	**Name**
Req_F_1.29	Data Aggregation
UMF compliant system should aggregate the monitored data in order to reduce the amount of data to be analysed.	
ID	**Name**
Req_F_1.30	Traffic Anomaly Detection
UMF must support methods for the detection (proactive or reactive) of events such as network traffic anomalies, faults and congestion.	
ID	**Name**
Req_F_1.31	NMS function to NMS function Interfaces

UMF must provide interfaces for exchanges between NMS functional blocks such as Situation Analysis/ Diagnosis, Candidate Solution Computation and Solution Selection and Elaboration or NMS functionalities.	
ID	**Name**
Req_F_1.32	OSS Interface to NMS
UMF must provide interface for communicating operator's goals to NMS so as to be taken into account for the selection of the mitigation/ reparation plan (human to network interfaces for inserting business goals to be translated into policies).	
ID	**Name**
Req_F_1.33	EMS to OSS interface
UMF must provide Network (EMS) to human interface (OSS) for reporting failure of re-configuration actions.	
ID	**Name**
Req_F_1.34	Human to Network Interface
UMF must provide H2N Interface for reporting users' problems and evaluation of the system.	

2.2.1.2 Use Case 2

There is no update (with respect to D4.1)

ID	**Name**
Req_F_2.1	Active Self-Monitoring
UMF shall support active self-monitoring (i.e. sending events and/or probe packet(s) into the network and measuring responses, e.g. in terms of QoS parameters).	
ID	**Name**
Req_F_2.2	Passive Self-Monitoring
UMF shall support passive self-monitoring (i.e. capturing data as it passes by).	
ID	**Name**
Req_F_2.3	Network Knowledge Extraction
UMF compliant system should be able to collect, filter and elaborate monitored data and events in order to extract network knowledge.	
ID	**Name**
Req_F_2.4	Network Stability Models and Tools
UMF compliant system should have a set of network stability models and tools to be plugged into the run-time environment.	
ID	**Name**
Req_F_2.5	Proactive Self-Stabilization
During Operations, UMF support methods for proactive self-stabilization actions (e.g. prediction and preventive actions).	
ID	**Name**
Req_F_2.6	Reactive Self-Stabilization
During Operations, UMF support methods for reactive self-stabilization actions (e.g. detection and corrective actions).	
ID	**Name**
Req_F_2.7	Human de-activation of self-* features
During Operations UMF shall support humans (e.g. through a specific interface) to take actions to deactivate "autonomic and self-*" features.	
ID	**Name**
Req_F_2.8	On-line self-prevention actions
During Operations, UMF shall support methods for taking on-line self-prevention actions (e.g. coordination, conflict resolution of self-* features).	
ID	**Name**
Req_F_2.9	External knowledge
UMF compliant system should allow exploiting also external knowledge (e.g. vulnerable state descriptions).	
ID	**Name**
Req_F_2.10	Validation of self-* features
During Planning, UMF compliant system should allow to validate the activation of self-* features (e.g. though simulation and to prediction of network dynamics using off-line tool).	
ID	**Name**
Req_F_2.11	Orchestration of self-* features
During Operations, UMF shall support methods for orchestrating self-* features to assure network stability and	

performance.	
ID	**Name**
Req_F_2.12	Map of self-* features
UMF shall have a map (e.g. repository) about all self-* features deployed into the network (Operators may wish to have a full control how and where self-* features are deployed).	
ID	**Name**
Req_F_2.13	Human Interface for on-line
UMF shall have a human interface to assess on-line network stability and to de-activate self-* features.	
ID	**Name**
Req_F_2.14	Human Interface for off-line
UMF shall have a human interface to assess off-line validation of self-* features.	

2.2.1.3 Use Case 3

There is no update (with respect to D4.1)

ID	**Name**
Req_F_3.1	H2N interface
The UMF shall provide a H2N Interface to insert high-level goals, to deliver control and management and to feedback system checks.	
ID	**Name**
Req_F_3.2	Policy language translation
The UMF shall provide translation of operator specified Policies into clear configuration and management actions.	
ID	**Name**
Req_F_3.3	Data Monitoring
A UMF compliant system shall provide/support means for the monitoring of access/backhaul/core networks (nodes and links).	
ID	**Name**
Req_F_3.4	Mobility Management
A UMF compliant system shall support QoS aware mobility management (incl. movement detection and handover execution).	
ID	**Name**
Req_F_3.5	Data processing
UMF shall provide means and support tools for aggregation & processing of monitored data (e.g., contextual data).	
ID	**Name**
Req_F_3.6	Network Monitoring
UMF MUST provide means to monitor and evaluate for End-to-End (E2E) connection/session status/statistics.	
ID	**Name**
Req_F_3.7	Conflict management
UMF SHOULD have some degree of conflict resolution (signalling vs. performance).	
ID	**Name**
Req_F_3.8	Platform management
A UMF compliant system may support multi-homed/multi-interface devices for effective load balancing (e.g., flow mobility) and efficient resource management (e.g., capacity increase).	
ID	**Name**
Req_F_3.9	Capabilities discovery
[1] UMF shall discover Service/network capabilities (e.g., bandwidth, error rates, modulation, energy, processing power, storage, transcoding abilities etc.).	
ID	**Name**
Req_F_3.10	Route management

The UMF may support routing strategies for route optimization.	
ID	**Name**
Req_F_3.11	Fault tolerance
The UMF shall provide necessary context information with global-scope for load balancing (i.e., congestion control/fault tolerance) – in backhaul and access networks. The load balancing strategy will be used to circumvent failed/congested/non-optimum nodes/paths.	
ID	**Name**
Req_F_3.12	Resource management
A UMF compliant system shall provide methods for autonomic resource management (& allocation) – incl. tunnel management. The resource management function shall also include sending resource reports to relevant entities.	
ID	**Name**
Req_F_3.13	Virtualization Management
A UMF compliant system MAY provide control and management methods for content/function/gateway virtualisation/migration.	
ID	**Name**
Req_F_3.14	Service data Monitoring
UMF shall give a method to access, for monitoring purpose, to any service data (performance indicators, services alarms, service's configuration, services semantic, services messages).	
ID	**Name**
Req_F_3.15	Network data Monitoring
UMF shall give a method to access in monitoring purpose to any network data (e.g., alarms, protocol configuration/semantics/messages, hardware parameters, performance indicators etc.).	
ID	**Name**
Req_F_3.16	Elaborated Data/Context Management
UMF shall give a method to access context base for obtaining, storing and updating context information.	
ID	**Name**
Req_F_3.17	Knowledge Data Management
UMF shall give a method to access knowledge base for obtaining, storing and updating knowledge information.	
ID	**Name**
Req_F_3.18	Topology information Monitoring
UMF shall give a method to access in monitoring purpose to any topology information data.	
ID	**Name**
Req_F_3.19	Contextual data translation
A UMF compliant system shall give a method for the translation of contextual data (upper layers to lower layers in the hierarchy).	
ID	**Name**
Req_F_3.20	Contextual data filtering/pre-processing
A UMF compliant system shall give a method to perform filtering and pre-processing to contextual data (lower layers to upper layers in the hierarchy).	
ID	**Name**
Req_F_3.21	Information flow management
The UMF shall provide interfaces for communication, between Knowledge repository and decision engine, between SON entities and Governance tool, and also amongst different SON entities.	

2.2.1.4 Use case 4

The functional requirements below have one updated requirement compared to the ones listed in D4.1, namely Req_F_4.9, which has been modified to better capture the coordination aspects of the network.

ID	**Name**
Req_F_4.1	H2N interface

UMF shall provide a H2N/GUI interface for inserting operator targets and policies.	
ID	**Name**
Req_F_4.2	Information for self-X operation
UMF shall provide Information for operating SON functionalities (network topology including the location of self-X entities, traffic characteristics, performance and QoS indicators).	
ID	**Name**
Req_F_4.3	Interfaces for self-X governance
UMF shall provide Interface between governance tools and self-X entities, to allow inserting, modify, interact with and monitor self-X processes.	
ID	**Name**
Req_F_4.4	Interfaces for self-X operation
UMF shall provide interface for communication between SON entities and between SON entities and the network.	
ID	**Name**
Req_F_4.5	Policy repository
UMF shall provide policy repositories for storing the defined policy rules.	
ID	**Name**
Req_F_4.6	Policy language
UMF shall provide policy language to allow operating self-X functions.	
ID	**Name**
Req_F_4.7	Policy generation
UMF should allow generating policies using a tool accessible via H2N interface.	
ID	**Name**
Req_F_4.8	Self-X triggers
UMF shall allow introducing triggers of self-X functions. Triggers should include events in time, periodic and manual activation, for a predetermined duration.	
ID	**Name**
Req_F_4.9	Policies for self-X operation
UMF shall provide rules and policies for operating self-X entities: to identify the involved self-X entities; to allow the interaction between self-X entities; to define hierarchy between SON functionalities; to activate/deactivate self-X functions; to transfer high level goals to low level operation of self-X entities; to provide information to - and from - self-X algorithms and between self-X and other network entities; to coordinate between SON functionalities (the policy in this case corresponds to utilities, weights and target thresholds for KPIs); and to resolve conflicts between running self-X processes when coordination fails.	
ID	**Name**
Req_F_4.10	Self-X monitoring
UMF shall allow monitoring network performance and parameters (including indicators related to- and parameters modified by SON running processes).	
ID	**Name**
Req_F_4.11	Self-recovery
In case of human intervention in the autonomic system (policy modification, deactivation, update/evolution of monitored KPI set), the system should return smoothly and quickly to the autonomic process.	
ID	**Name**
Req_F_4.12	Policy adjustment
UMF shall allow adjusting policies according to feedback from running self-X processes.	

2.2.1.5 Use Case 6

An extensive list of functional requirements had been defined for both UC6 and former UC5 (known as "Network Morphing") already in D4.1. After the integration of these UCs the requirements of the two lists were further elaborated and finally merged. The merged list of the UC6 functional requirements can be found hereafter. The used template is the common template defined by the project while a field that holds information with respect to the former functional requirements as numbered in D4.1 has also been added under the new/current numbering of the requirement. Current FR6.11 and FR6.16 have also been elaborated and updated during the 2nd burst of UC6.

ID	Name

FR6.1	H2N interface for request and goals expression
Req_F_6.1, Req_F_5.18, Req_F_5.22	UMF should support the means (H2N interface with appropriate GUI) for the operator to express requests and goals.
ID	**Name**
FR6.2	Policies derivation from business goals
Req_F_6.2, Req_F_6.4	UMF should provide functionality for the derivation of policies based on operator's requests and goals
ID	**Name**
FR6.3	Policy conflict resolution
Req_F_6.3	UMF should provide functionality for policy conflict resolution.
ID	**Name**
FR6.4	Policy rules translation and distribution
Req_F_6.5, Req_F_5.11, Req_F_5.12, Req_F_6.11	UMF should enable translation and distribution of policy rules among network elements to enforce policy decisions (e.g. routing path/tables update).
ID	**Name**
FR6.5	Business level entries/request analysis
Req_F_6.6	UMF compliant system should provide functionality for analysing the business /service level requirements and derive (translate them to) technology (network) specific requirements.
ID	**Name**
FR6.6	Candidate solutions discovery/reasoning
Req_F_6.7, Req_F_6.8	UMF compliant system should provide functionality for discovering/determining (and reasoning) the candidate solutions (networks) that can satisfy the derived network requirements.
ID	**Name**
FR6.7	RAN optimization function
Req_F_6.9, Req_F_6.10	UMF compliant system should optimize the provided QoS based on the resource consumption in the RAN segment.
ID	**Name**
FR6.8	Backhaul/Core optimization function
Req_F_6.9, Req_F_5.10, Req_F_5.15, Req_F_5.19	UMF compliant system should optimize the provided QoS based on the resource consumption in the Backhaul/Core segment.
ID	**Name**
FR6.9	Collaboration and negotiation
Req_F_6.12, Req_F_6.28	UMF should provide collaboration and negotiation functions for the establishment of agreements and federation between network segments (RAN & Backhaul/Core), domains, operators and services providers.
ID	**Name**
FR6.10	Conflict resolution mechanisms
Req_F_6.13, Req_F_6.37	UMF compliant system should provide mechanisms for on-line conflicts and dependencies resolution for different self-optimization and/or self-healing actions.
ID	**Name**
FR6.11	RAN & Backhaul/Core Network monitoring
Req_F_6.14, Req_F_6.23, Req_F_6.26, Req_F_6,32, Req_F_5.2, Req_F_5.7, Req_F_5.8	UMF compliant system should provide functionalities and UMF should provide interfaces for monitoring RAN & Backhaul/Core Network parameters (e.g. Capacity, Network Load, traffic flows between segments) to be taken into account in the load optimization and in knowledge building.
ID	**Name**
FR6.12	Conversion of generic configuration into technology-specific configurations
Req_F_6.15, Req_F_6.33, Req_F_5.17	UMF should provide functionality for converting instances of a generic configuration model into technology-specific configurations.
ID	**Name**
FR6.13	Autonomic functions for self-x actions

Req_F_6.16, Req_F_5.13, Req_F_6.32	UMF should provide autonomic functions that will trigger the appropriate self-x (optimization, configuration, healing, etc) actions.
ID	**Name**
FR6.14	Policy repositories
Req_F_6.17, Req_F_6.18	UMF should provide policy repositories and the appropriate interfaces for accessing them.
ID	**Name**
FR6.15	Policy Models
Req_F_6.19	UMF should provide policy models (network policies, routing update policies…).
ID	**Name**
FR6.16	Information and Knowledge management
[2] Req_F_6.20, Req_F_5.1, Req_F_6.21, Req_F_6.29, Req_F_6.30, Req_F_6.27, Req_F_6.31, Req_F_5.9, Req_F_5.14, Req_F_5.16	UMF compliant systems should provide building and UMF should provide storage/retrieval/dissemination of information and knowledge on SLAs, Applications, User classes, RAN (network, resources, configuration), Backhaul / Core (traffic measurements, bandwidth estimations, network configurations), Traffic mobility requirements, and Traffic demand descriptions.
ID	**Name**
FR6.17	RAN and Backhaul/Core QoS level related offers.
Req_F_6.22, Req_F_6.24	UMF should provide interfaces for requesting/receiving QoS level related offers from RAN (responsible RRMs) and Backhaul/Core network segments.
ID	**Name**
FR6.18	Packet marking
Req_F_6.25	UMF should provide modules and interfaces for packet marking in the backhaul/core segment. I.e. Mark packets in order to construct and control traffic classes (profiles) and indicate congestion levels.
ID	**Name**
FR6.19	Traffic aggregation
Req_F_5.3, Req_F_5.4	UMF must support features to aggregate network core traffic data in quasi real-time with control over the aggregation process and the appropriate interfaces for monitoring aggregated traffic data in a streaming fashion with periodical or on-demand export.
ID	**Name**
FR6.20	Bandwidth estimation
Req_F_5.5, Req_F_5.6, Req_F_5.18	UMF compliant system must provide methods to estimate bandwidth needs from aggregated traffic data and UMF must provide interfaces to access the bandwidth estimations (either in pull or push mode).
ID	**Name**
FR6.21	SLA compliance monitoring
Req_F_5.20, Req_F_5.21	UMF compliant system must provide mechanisms and UMF must support interfaces for monitoring SLA compliance, e.g. target data rate.

Additionally to these, during the 2nd burst of the UC, 7 new functional requirements were identified and will be tackled by the end of the project. These are:

ID	**Name**
FR6.22	Service Assessment
UMF compliant system should provide mechanisms for assessing the offered services.	
ID	**Name**
FR6.23	Repository of Running Mechanisms
UMF must support ways for mechanisms identifying and reporting themselves to the Governance. UMF must also provide storage of this information, i.e. of the capabilities of the reported mechanism.	
ID	**Name**
FR6.24	Stability Evaluation

UMF compliant system should provide evaluation of the stability of the system.	
ID	**Name**
FR6.25	Context Aware Policies
UMF must provide means for context awareness in policies.	
ID	**Name**
FR6.26	Embodiment of mechanisms in NE level
UMF should allow the intelligent embodiment of some mechanisms in the network element level	
ID	**Name**
FR6.27	Information Model
UMF must provide an Information Model.	
ID	**Name**
FR6.28	Trust/ Certification
UMF compliant system should support the assessment of the trustworthiness of the system both in terms of trusting that the goal will be achieved with respect to the goals and in security terms.	

2.2.1.6 Use Case 7

With respect to the requirements listed in Deliverable D4.1, Use Case 7 has included a new functional requirement (Req_F_7.22), related to the self-healing actions in wireless networks. The extended list of requirements is presented in the table below:

ID	**Name**
Req_F_7.1	Information model
UMF must provide an information model that allows the representation of all the elements involved in the lifecycle of a service, starting from the business goals down to the network elements.	
ID	**Name**
Req_F_7.2	Knowledge base
UMF must include a knowledge base to store all the relevant information for the modelling, provisioning and runtime phase of a service.	
ID	**Name**
Req_F_7.3	Knowledge base management
UMF must provide mechanisms for the insertion, removal and modification of the information stored in the knowledge base.	
ID	**Name**
Req_F_7.4	Business goals language
UMF must incorporate a language to express high-level business goals.	
ID	**Name**
Req_F_7.5	H2N interface
UMF must incorporate a H2N graphical interface to insert high-level business goals.	
ID	**Name**
Req_F_7.6	Business goals translation
UMF must provide mechanisms for the translation of high-level goals to specific policies of network entities.	
ID	**Name**
Req_F_7.7	Policy language
UMF must incorporate a policy language to provide information and allow communication between the autonomic entities.	
ID	**Name**
Req_F_7.8	Interfaces for policy management
UMF must define interfaces between governance tools and autonomic entities, to allow the insertion, modification, and dissemination of policies.	
ID	**Name**
Req_F_7.9	Policy conflict resolution
UMF should support mechanisms for policy conflict resolution.	
ID	**Name**
Req_F_7.10	Policy dissemination
UMF must support mechanisms for the dissemination of policies to the network elements.	
ID	**Name**
Req_F_7.11	Network context monitoring
UMF compliant system must monitor the operational status of the individual autonomic nodes (QoS parameters should include BER, Jitter, Access Delay, Throughput, Delay).	

ID	Name
Req_F_7.12	Self-diagnosis
UMF compliant system must be able to diagnose the autonomic nodes based on their current operational status.	
ID	**Name**
Req_F_7.13	Self-healing
UMF compliant system must include self-healing capabilities.	
ID	**Name**
Req_F_7.14	QoS calculation
UMF must support mechanisms for deriving the QoS based on the monitoring data collected from the autonomic elements	
ID	**Name**
Req_F_7.15	Behaviour assessment
UMF must support feedback mechanisms to assess that the behaviour of the running autonomic entities is the one that correspond to the high level goals set by the human operator.	
ID	**Name**
Req_F_7.16	Self-optimization/ adjustment
UMF compliant system should provide mechanisms for handling network performance degradation. For example: Incorporation of scheduling actions and mechanisms for policy-based self-optimisation/adjustment of resources to handle the network performance degradation, or suboptimal allocation of network resources to different nodes (e.g. selfish nodes).	
ID	**Name**
Req_F_7.17	Context discovery
UMF compliant system must provide mechanisms/algorithms for the network elements to self-discover their neighbours, using network protocols.	
ID	**Name**
Req_F_7.18	Policy-based trust management mechanisms
UMF must support trust management schemes for detection of faulty/malicious behaviours of network elements based on operator policies.	
ID	**Name**
Req_F_7.19	Network to human notifications
UMF must define interfaces between autonomic entities and governance tools to allow the communication of collected monitoring, information, notification of results of diagnosis processes, notification of self-healing actions, notification of alarms.	
ID	**Name**
Req_F_7.20	Information retrieval
UMF must define interfaces between governance tools and autonomic entities to allow the query of: current status of network elements, configuration information, historical monitoring information, historical diagnosis results, historical self-healing actions, and historical notifications.	
ID	**Name**
Req_F_7.21	Real time monitoring
Real time monitoring data and status of the network elements SHOULD be made available to the operator on the fly.	
ID	**Name**
Req_F_7.22	Self-healing
UMF must support decision making process based on semantic models for self-healing purposes in wireless access network elements.	

2.2.2 Synthesis of use cases non-functional requirements

This section analyses the UC non-functional requirements, as derived from the analysis of the use cases.

2.2.2.1 Use Case 1

There is no update (with respect to D4.1)

ID	Name
Req_NF_1.1	Adaptability to Operators topology changes
UMF must take into account the topology change and /or the configuration change and should be adjustable to the adding	

or deleting of a component.	
ID	**Name**
Req_NF_1.2	Adaptability to end to end management (Horizontal and vertical)
UMF must be applied for cross-layer and end-to-end perspective, both media and signalling, facing multiple network domains and technologies.	
ID	**Name**
Req_NF_1.3	Adaptability to operator organization
UMF must respect present operator's tools, processes and human organization and ensure its evolution.	
ID	**Name**
Req_NF_1.4	Reflexivity
Management Framework should support reflexivity. i.e., Management Framework should expose information regarding managed network and service infrastructures at an abstract level (information related to the identification of the network and service resources, connections, dependencies between services and needed resources for use or QoS)	
ID	**Name**
Req_NF_1.6	Event prediction algorithms Adaptability
UMF must allow algorithm Adaptability to face network/context constraints.	
ID	**Name**
Req_NF_1.7	Event prediction algorithms Robustness
UMF should allow algorithm Robustness to face: Missing data in monitoring processes, Time fluctuations in monitored data analysis.	
ID	**Name**
Req_NF_1.8	Event prediction algorithms scalability
UMF must allow algorithm scalability to face: multiple prediction time scale, Network/Context data amount, complexity, and changes, Topology changes, Organization structure evolution.	

2.2.2.2 Use Case 2

There is no update (with respect to D4.1)

ID	**Name**
Req_NF_2.1	Network Stability visual representation
UMF shall be able to provide a visual representation of the network stability (e.g. with different levels of details), which is easy to be read and understood.	
ID	**Name**
Req_NF_2.2	Performance in self-stabilization
UMF shall actuate self-stabilization actions within a time scale to avoid that instabilities jeopardize network performance.	
ID	**Name**
Req_NF_2.3	Interoperability with legacy network management systems
UMF shall be compatible with legacy network management systems during Operations and Planning.	
ID	**Name**
Req_NF_2.4	UMF supportability
UMF shall be easily updated to accommodate/adapt usage in diverse network and service scenarios (for instance, it should be easy to add new self-* features in the framework for improving stability).	
ID	**Name**
Req_NF_2.5	Security of network and services stability control
UMF shall have the ability to prevent and/or forbid access to a system by unauthorized parties.	
ID	**Name**
Req_NF_2.6	Resilience of network and services stability control
UMF shall be redundant so to provide and maintain an acceptable level of stability control in face of faults and challenges to normal operations.	

2.2.2.3 Use Case 3

There is no update (with respect to D4.1)

ID	**Name**

Req_NF_3.1	Stability of network planning and configuration
The UMF shall ensure uniformity and stability across network segments in terms of high-level network planning and configuration.	
ID	**Name**
Req_NF_3.2	Migration Management
A UMF compliant system shall ensure that migrations (e.g., of servers, functions, services, content etc) are stable and optimised in terms of configurable policies.	
ID	**Name**
Req_NF_3.3	Network Survivability
A UMF compliant system should ensure survivability and maintenance of necessary resources during active sessions against any adverse situation e.g., by providing redundancy and/or failure recovery mechanisms.	

2.2.2.4 Use Case 4

There is no update (with respect to D4.1)

ID	**Name**
Req_NF_4.1	Scalability of Self-X
Self-X functions shall be scalable to allow good operation in large number of network nodes (e.g. self-optimization functions in eNodeBs of a LTE network, including adjacent nodes).	
ID	**Name**
Req_NF_4.2	Robustness of Self-X
Self-X functions shall be robust: QoS/performance deterioration shall be limited to a pre-defined threshold in any self-X process.	
ID	**Name**
Req_NF_4.3	Time scales
Time scales of Self-X functions for converging to a new stable state should be known to allow coordination of running Self-X processes.	
ID	**Name**
Req_NF_4.4	Convergence time
Convergence time of SON mechanisms to reach a desired stable state (individually or jointly –in case of SON coordination) should be minimal.	

2.2.2.5 Use Case 6

During the 2nd burst of the UC, 1 additional non-functional requirement related to scalability has been identified (Req_NF_6.4). The updated list of UC6 non-functional requirements is the following:

ID	**Name**
Req_NF_6.1	H2N GUI friendliness
The H2N GUI should be human friendly for facilitating the expression of the requests and goals.	
ID	**Name**
Req_NF_6.2	Optimized performance of load allocation to ingress/egress nodes
UMF should enable optimized load allocation to ingress/egress nodes.	
ID	**Name**
Req_NF_6.3	Accuracy of load allocation to ingress/egress nodes
UMF should ensure that the proposed mechanisms for routing optimization/load allocation to ingress/egress nodes converge to optimal solution.	
ID	**Name**
Req_NF_6.4	Scalability
UMF should ensure scalability of the system functioning.	

2.2.2.6 Use Case 7

With respect to the requirements listed in Deliverable D4.1, Use Case 7 has included two new requirements (Req_NF_7.5 and Req_NF_7.6) concerning security and scalability issues. The extended list of requirements is presented in the table below:

ID	Name
Req_NF_7.1	Governance Tool Usability
The system through the H2N interface should allow network operators to operate networks with a reduced human effort, without requiring specialized knowledge of the network behaviour and with reduced configuration errors.	
ID	**Name**
Req_NF_7.2	Success of high level goals
The system should enable H2N GUI to improve the success of high-level goals through fine-tuning based on feedback mechanisms which monitoring the underlying networks.	
ID	**Name**
Req_NF_7.3	Adaptability
Self-Diagnosis and self-healing processes should be dynamic and adjustable, following changes in the network context.	
ID	**Name**
Req_NF_7.4	Performance of Trust management
The application of trusted network behaviour should not have an important impact in the performance of the system.	
ID	**Name**
Req_NF_7.5	Security
UMF should provide secured communication mechanisms for governing the autonomic entities, and for the exchange of information between autonomic entities.	
ID	**Name**
Req_NF_7.6	Scalability
UMF should support scalability of the system.	

3 QFD Analysis

When preparing and refining use-cases planning and lifecycles, recommendations from the first intermediate review have been taken into account by pursuing a first concrete step towards prioritisation of the use case problems and requirements. Quality Function Deployment (QFD) methodology has been used for this purpose.

Quality Function Deployment (developed by Y. Akao in Japan in 1966) [2] is a systematic approach to design and develop a product (of any kind, including pieces of software) based on a close awareness of customer desires and requirements, coupled with the integration of functional groups (of a project team or company).

Quoting Y. Akao, QFD "*is a method for developing a design quality aimed at satisfying the consumer and then translating the consumer's demand into design targets and major quality assurance points to be used throughout the production phase. ... [QFD] is a way to assure the design quality while the product is still in the design stage.*"

In essence, quality is the barycentre of the methodology and the ultimate goal is to translate (often) subjective quality criteria into objective ones that can be quantified and measured, and which can then be used to design and develop the product. Basically, QFD allows determining how and where priorities are to be assigned in product development. The intent is to employ objective procedures in increasing detail throughout the development of the product.

The three main goals in implementing QFD are:

1. Prioritize spoken and unspoken customer desires and needs;
2. Translate these needs into technical characteristics, requirements and specifications;
3. Develop and deliver a quality product (or service) by focusing on customer satisfaction whilst optimizing usage of internal costs, resources and teams (e.g. in a project, or in a Company).

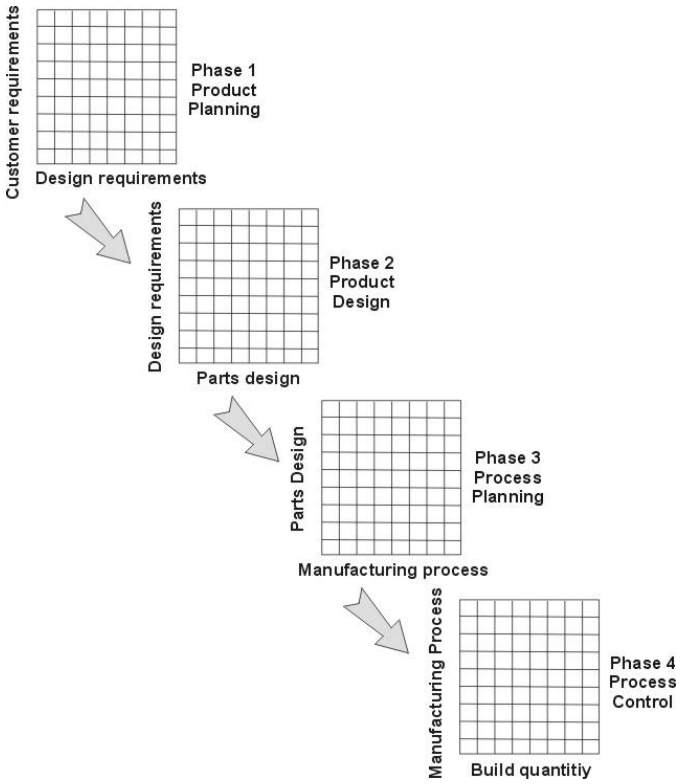

Figure 3-1: Complete flow of the QFD analysis

QFD uses some principles from Concurrent Engineering in that cross-functional teams are involved in all phases of product development. As depicted in Figure 3-1, each of the four phases in a QFD process uses a matrix to translate customer requirements from initial planning stages through production control. These phases are:

- Phase 1, Product Planning. It is also called The House of Quality. Main goals are: documenting customers' requirements, competitive opportunities, product measurements, competing product measures, and the technical ability of the organization to meet each customer requirement.
- Phase 2, Product Design. Product concepts are created during this phase and part specifications are documented. Parts that are determined to be most important to meeting customer needs are then deployed into process planning, or Phase 3.
- Phase 3, Process Planning. During this phase, development processes are planned and flowcharted and the parameters (or target values) are documented.
- Phase 4, Process Control. It concerns the control of the development processes, the maintenance of schedules, and skills training for developers. Also, in this phase decisions are made as to which process poses the most risk and controls are put in place to prevent failures.

For further details see references [2], [9].

For each UC in UNIVERSELF, we evaluated correlations between the problems/needs and the functional requirements. In other words we evaluated how each requirement is contributing to solve each problem. These correlation evaluations were done according to the following seven Quality Functionalities for each UC problem and requirement:

- New Functionality: correlation values related to the system whose requirements are contributing to solve the problems will introduce or not a new functionality.
- Costs of adoption: correlation values related to the cost of adoption of the system whose requirements are contributing to solve the problems.
- Performance: correlation values related to the impact in terms of performance by adoption of the system whose requirements are contributing to solve the problems.
- Flexibility: correlation values related to the impact in terms of flexibility by the adoption of the system whose requirements are contributing to solve the problems.
- Interoperability: correlation values related to the impact in terms of interoperability by the adoption of the system whose requirements are contributing to solve the problems.
- Reducing OPEX: correlation values related to the impact in terms of reducing OPEX by the adoption of the system whose requirements are contributing to solve the problems.
- Overall value: correlation values related to the average of each previous value.

3.1 Results of QFD analysis

This section is reporting a summary of the prioritized problems and requirements obtained with QFD analysis.

3.1.1 Use Case 1

For making the QFD analysis, UC1 adjusted the granularity of the problems as follows:

Enabling Self-Proactive and Reactive Diagnosis for networks and services	**P1.1.1**	End to end , cross layer and local self-diagnosis (including Customer's view)
	P1.1.2	Detection, estimation of possible anomalies/issues/problems before occurring (proactive)
	P1.1.3	Detection, estimation of possible known and occurring anomalies (reactive)
	P1.1.4	Detection, estimation of possible unknown anomalies.
	P1.1.5	Analysis and and qualification of related detection
Enabling Self-Healing for networks and services	**P1.2.1**	Defining mitigation and reparation plans
	P1.2.2	Applying the correct mitigation or reparation plan based on business goals
Controlling Self-Diagnosis and Healing	**P1.3.1**	Self-diag/healing triggered by network/service events and by subscriber events according business goals
	P1.3.2	Enable human to validate diagnosis and reparation/mitigation plan

Table 3-1: UC1 problems

3.1.1.1 Importance of UC1 problems

2 (over 9) problems are emerging as very important:

- P1.1.1 (End to end, cross layer and local self-diagnosis (including Customer's view))

- P1.1.2 (Detection, estimation of possible anomalies/issues/problems before occurring (proactive))

3 problems are following as important:

- P1.1.3 (Detection, estimation of possible known and occurring anomalies (reactive))
- P1.1.5 (Analysis and qualification of related detection)
- P1.1.4 (Detection, estimation of possible unknown anomalies)

4 other problems have a lower than average importance.

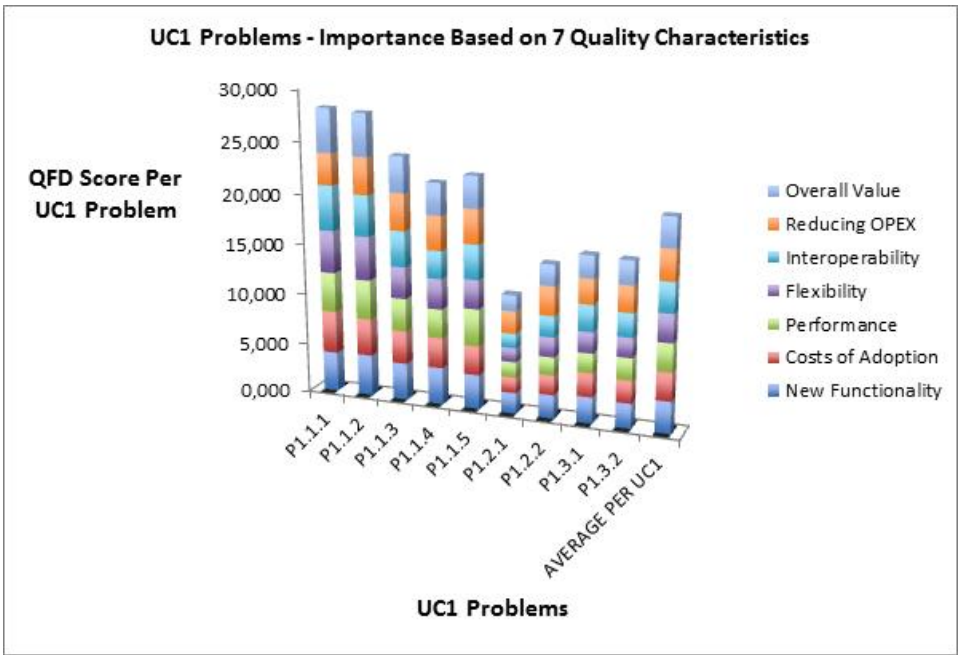

Figure 3-2: QFD score per UC1 problems

3.1.1.2 Requirements contribution

In UC1, we have a set of 33 functional requirements. In order to simplify the QFD processing and for visibility, these 33 requirements were aggregated into 15 aggregated representative requirements (named AFR):

- AFR1.1 corresponds to Req_F1.1+1.2+1.5
- AFR1.2 corresponds to Req_F1.3+1.4
- AFR1.3 corresponds to Req_F1.6+1.7+1.29
- AFR1.4 corresponds to Req_F1.8
- AFR1.5 corresponds to Req_F1.9
- AFR1.6 corresponds to Req_F1.10+1.11
- AFR1.7 corresponds to Req_F1.14+1.15+1.16
- AFR1.8 corresponds to Req_F1.17+1.18
- AFR1.9 corresponds to Req_F1.19+1.20
- AFR1.10 corresponds to Req_F1.21
- AFR1.11 corresponds to Req_F1.22+1.23+1.24+1.25+1.26+1.27+1.31
- AFR1.12 corresponds to Req_F1.28
- AFR1.13 corresponds to Req_F1.30
- AFR1.14 corresponds to Req_F1.33
- AFR1.15 corresponds to Req_F1.32+1.34

Some of the aggregated requirements are identified as “more” important:

- AFR1.11: UMF should provide Self-x enabling elements and Network (NMS) to human interfaces (OSS) for event detection and mitigation triggering reporting and Self-x enabling elements to Network entities (NMS) interfaces to share detection reports. UMF must also provide interfaces to share detection and mitigation triggering reports between Self-X enabling elements and NMS functional blocks. Exchanges concerns also Situation Analysis/ Diagnosis, Candidate Solution Computation and Solution Selection and Elaboration or NMS functionalities.
    - Corresponds to Req_F1.22 1.23 1.24 1.25 1.26 .1.27 1.31
- AFR1.9: UMF must provide data sharing methods between self-x enabling elements. UMF should provide mitigation policies for each predicted event.
    - Corresponds to Req_F1.19 1.20
- AFR1.3: UMF shall give methods for the translation of contextual data (upper layers to lower layers in the hierarchy) and to perform aggregation, filtering and pre-processing to contextual data (lower layers to upper layers in the hierarchy).
    - Corresponds to Req_F1.6 1.7 1.29
- AFR1.5: UMF should be able to pinpoint the root cause among many alarms and identify the problems that need to be fixed.
    - Corresponds to Req_F1.9
- Then, QFD highlights 15 over 33 UC1 Req_F

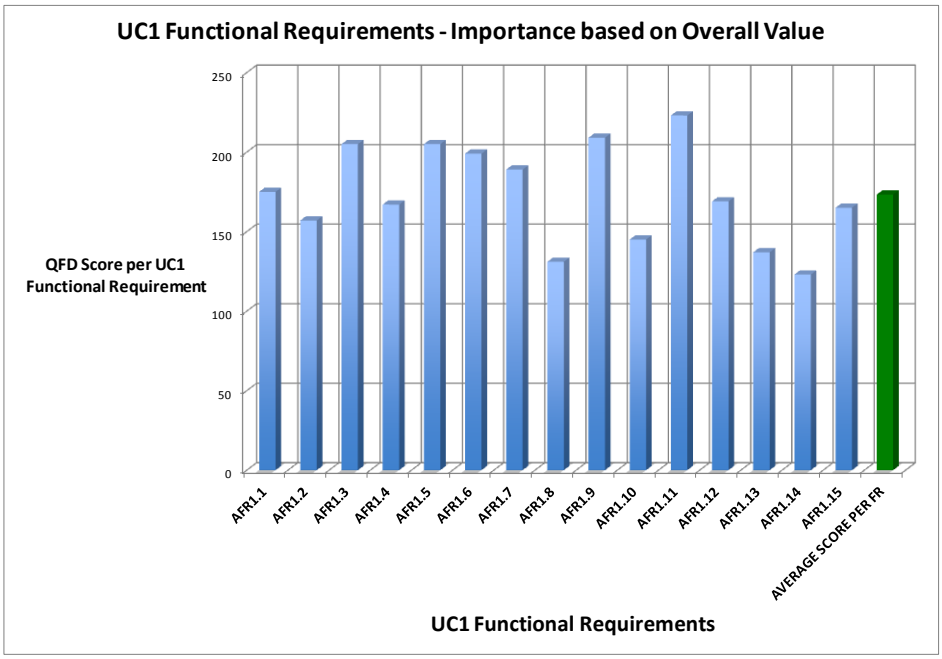

Figure 3-3: QFD score per UC1 Functional Requirement

In conclusion, priority is to first focus on self-diagnosis (proactive and reactive end-to-end cross layer reactive detection, related analysis) and then on the self-healing part.

The QFD requirement analysis shows that communication/integration/interaction (what elements and how) between management/managed entities (including Self-X enabled nodes and NMS) supporting self-diagnosis in E2E/Crosslayer needs to be considered first, supporting diagnosis/detection objectives.

3.1.2 Use Case 2

For making the QFD analysis, UC2 adjusted the granularity of the problems as follows:

PROBLEMS/NEEDS	PROBLEMS/NEEDS (lower level of details)
P2.1 -To have a run time environment for off-line validation and on-line control of self-* features.	P.2.1.1 - To define and collect a set of models, tools for off-line simulations - emulations and in line control of networks stability and performance Also models of embedded self-* mechanisms that can be used in the UMF for tracing the stability of adaptations of those mechanisms
P2.2 - To validate self-* features off-line (according to predefined criteria) before network deployment.	P2.2.1 -To make off-line validations based on simulations - emulations of network stability and performance
P2.3 - To monitor on-line parameters to assess and predict network stability during network operations.	P2.3.1 - To collect on-line data, events. Also to have traces (patterns) of safe and stable adaptations, be able to aggregate those traces and be able to use them in network forensics
P2.4- To analyse-elaborate data about network stability during network operations.	P.2.4.1 To filter and analyse data to assess and predict behaviour of network in terms of stability and performance. Be able to expand the aggregated history traces
P2.5 - To decide and actuate network self-stabilization in case of emerging instabilities	P2.5.1 -To decide the self-stabilization actions for maintaining stability and performance (according to SLA). Support the emergence of collaboration patterns (e.g. by means of triggering collaboration policies (predicates)
	P2.5.2 -To actuate network self-stabilization decision actions for maintaining stability and performance (according to SLA).
P2.6 -To decide and de-activate manually self-* features in case of persistent instabilities	P2.6.1 -To decide conditions for de-activating manually self-* features (e.g. persistent instabilities, network cannot self-stabilized). To define conditions per mechanism (group of mechanisms), in which a Call for Governance must be issued
	P2.6.2 - To predispose for the manual de-activation of self-* features

Table 3-2: UC2 problems

3.1.2.1 Importance of UC2 problems

P2.5.1, P2.1.1 and P2.2.1 contribute most in accommodating/resolving UC2 FRs, if ''Overall Value'' is considered in isolation. P2.5.2 and P2.6.2 contribute least in accommodating/resolving UC2 FRs, if ''Overall Value'' is considered. In synthesis, the problems to be dealt with higher priority are:

- P2.5.1 -To decide the self-stabilization actions for maintaining stability and performance (according to SLA).
- P2.1.1 - To define and collect a set of models, tools for off-line simulations - emulations and in line control of networks stability and performance.

- P2.2.1 -To make off-line validations based on simulations - emulations of network stability and performance.

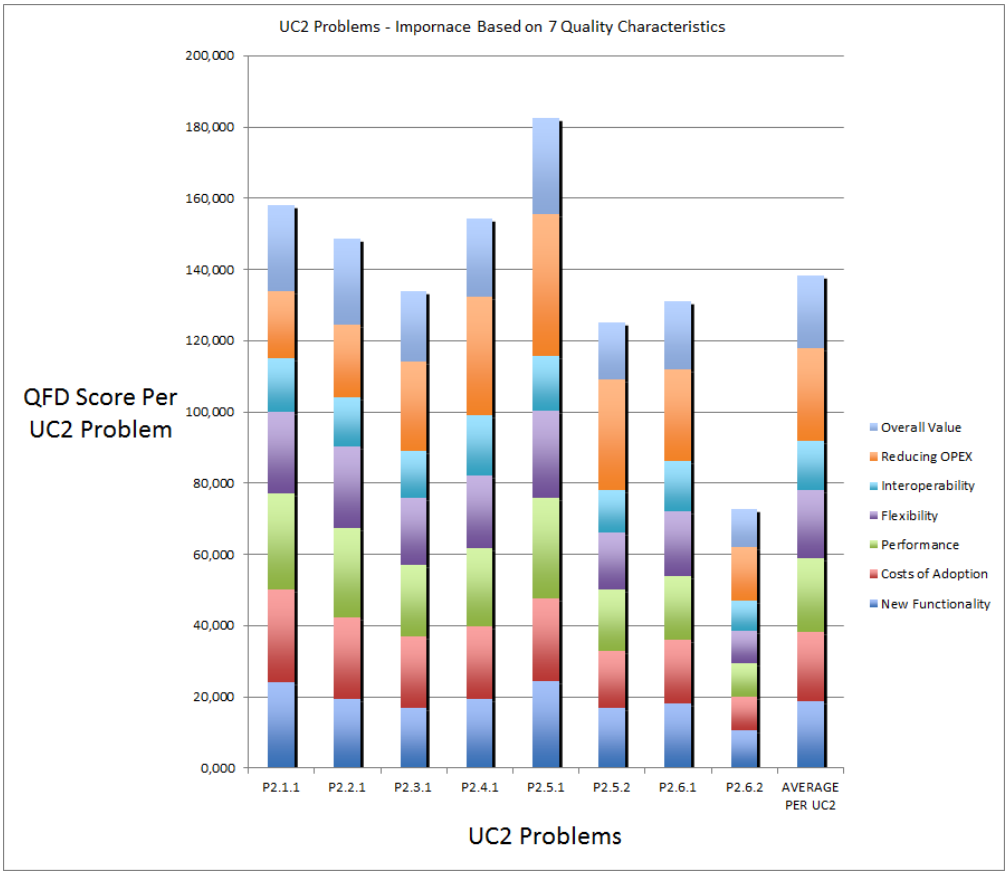

Figure 3-4: QFD score per UC2 problem

3.1.2.2 UC2 Requirements contribution

FR2.3, FR2.4 and FR2.11 contribute most in solving UC2 problems if ''Overall Value'' is considered.

FR2.2, FR2.10, FR2.12, FR2.13 and FR2.14 contribute least in solving UC2 problems, if ''Overall Value'' is considered. In synthesis, the requirements considered as ''more important'' are:

- Req_F_2.3 Network Knowledge Extraction - UMF shall have features to collect, filter and elaborate monitored data and events in order to extract network knowledge.
- Req_F_2.4 Network Stability Models and Tools - UMF shall have a set of network stability models and tools to be plugged into the run-time environment.
- Req_F_2.11 Orchestration of self-* features - During Operations UMF shall orchestrate self-* features to assure network stability and performance.

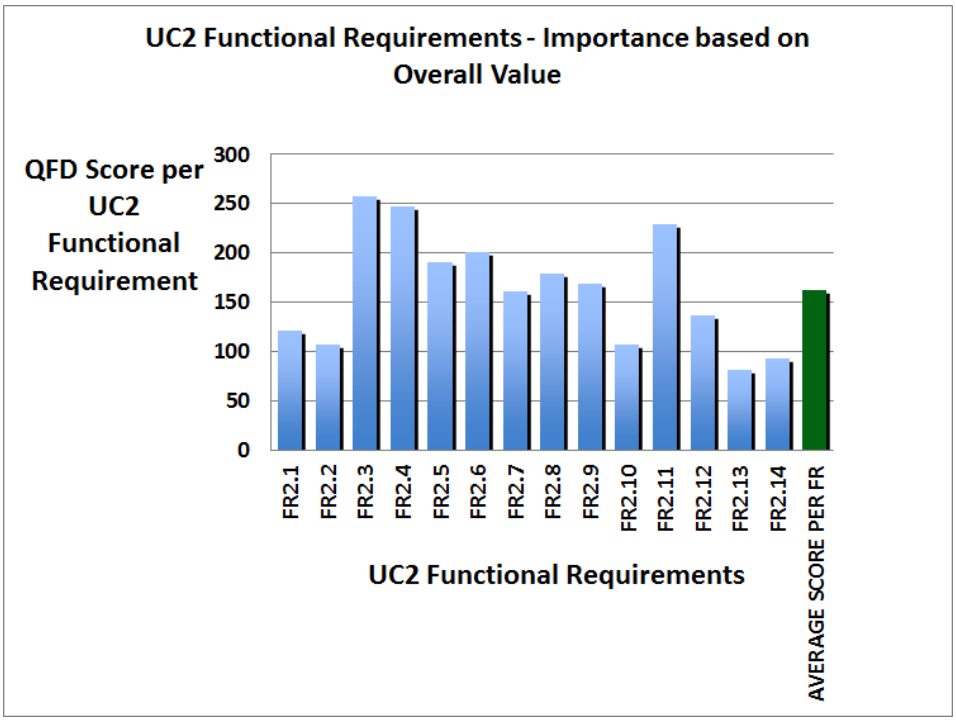

Figure 3-5: QFD score per UC2 Functional Requirement

In conclusion, QFD analysis of UC2 has indicated what are the problems to be dealt first, i.e.: 1) decide the self-stabilization actions for maintaining stability and performance (according to SLA); 2) to define and collect a set of models, tools for off-line simulations - emulations and in line control of networks stability and performance; 3) to make off-line validations based on simulations - emulations of network stability and performance.

UC2 is directing technical activities in this direction in order to provide first an overall picture, a sort of taxonomy, about the potential occurrence of instabilities in an autonomic networks (causes, performance indicators and characteristic parameters); the following step will be defining a set models and tools to make simulations – emulations of instability situations with the goals of identifying the most appropriate methods for pursuing (self-)stabilization.

3.1.3 Use Case 3

For making the QFD analysis, UC3 adjusted the granularity of the problems as follows:

PROBLEMS/NEEDS	**PROBLEMS/NEEDS (lower level of details)**
Load aware instantiation of core network functions and/or entities for resource efficient service delivery.	**P 3.1:** To develop strategies that would reduce the load (traffic, processing, signalling etc.) in the core network segments and data centres for efficient delivery of data/service/application to the mobile user. This would be achieved by decentralizing and migrating frequently used/critical resources/functions/services from the core/data centres towards the access and backhaul network nearer to the user. This issue will be studied and analysed with respect to the best practice solutions of network virtualization techniques for enabling dynamic migration of functions and resources.
Resource/function specification based on user service demand for 3GPP network architectures.	**P 3.2:** Specifying the resources and functions used (or required) by a mobile user (with varied mobility patterns) for accessing commonly used network services and applications (real time and non-real-time) in the context of 3GPP network architectures. Specifying the KPI for each of these services and applications and defining network/configuration/performance parameters for the commonly used services. Identifying the functional and operational enhancements of existing

	segments at the access/backhaul for supporting and hosting the migrated resources/functions/services).
Development of efficient algorithms for network function/entity migration.	**P 3.3:** Develop algorithms that would leverage the virtualization techniques and cloud concepts for seamless migration of resources/services/functions context. Develop novel techniques and algorithms for enabling the seamless mobility of resources/services/functions virtual clouds in sync with the user mobility.
Performance / impact analysis of load aware migration strategies on network architectures and services.	**P 3.4:** Develop simulation models to understand the implications of decentralizing the resources/functions/services from the core, and migrating them towards the access and backhaul segments and their impact on the network architecture and the corresponding network entities.

Table 3-3: UC3 problems

3.1.3.1 Importance of UC3 problems

With reference to 7 quality characteristics, the QFD analysis reveals P3.2 and P3.3 of ''overall'' high importance followed closely by P3.1. On the other hand P3.4 has the least importance (see Figure3-6).

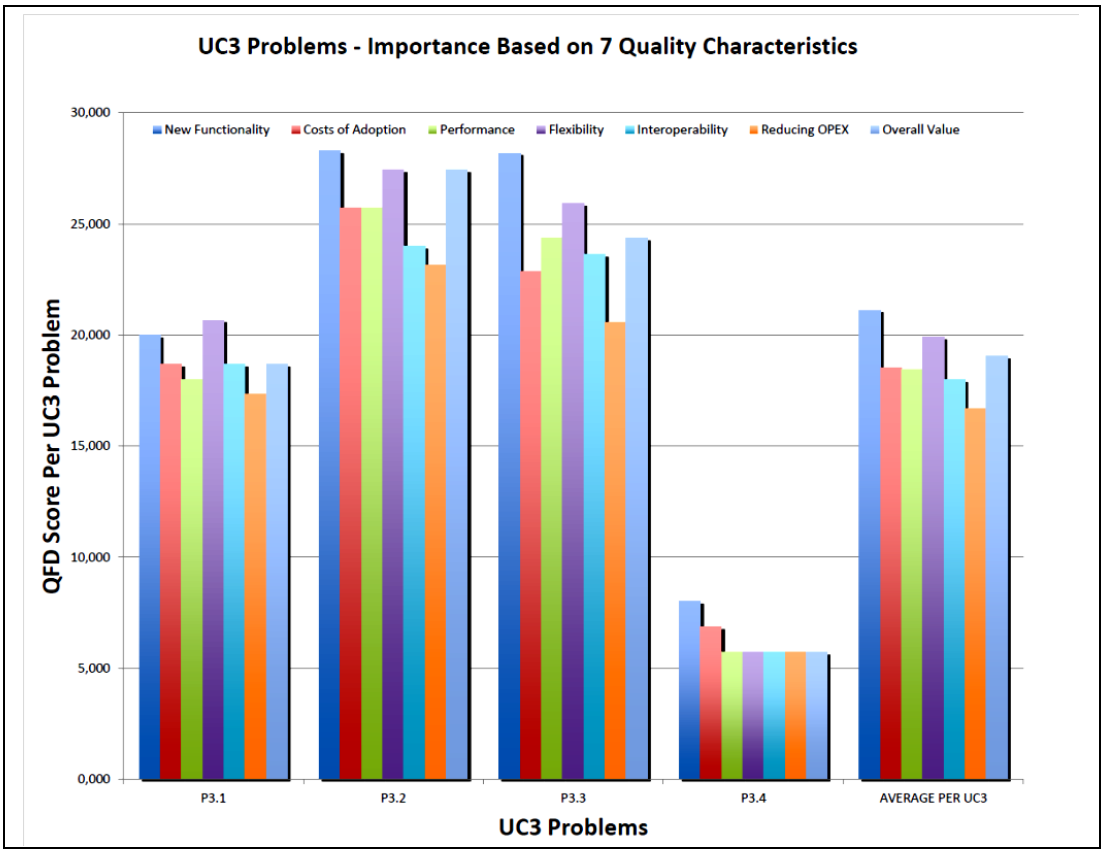

Figure3-6: QFD score per UC3 problem

3.1.3.2 UC3 Requirements contribution

Figure 3-7 shows the ranking of the 21 functional requirements indicated for UC3 based on the QFD scoring.

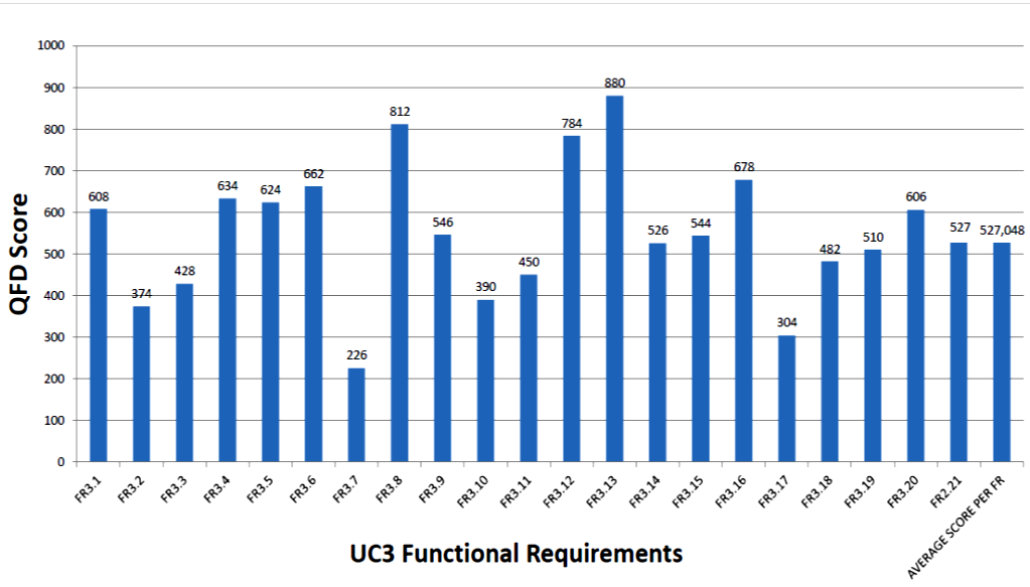

Figure 3-7: QFD score per UC3 Functional Requirements

For the sake of reference and convenience, the top 5 functional requirements are listed below:

Functional Requirement	Name	Description
3.13	Virtualization Management	The UMF MAY provide control and management methods for content/function/gateway virtualisation/migration.
3.8	Platform management	The UMF may support Multi-homed/multi-interface devices for effective load balancing (e.g., flow mobility) and efficient resource management (e.g., capacity increase).
3.12	Resource management	The UMF shall provide methods for autonomic resource management (& allocation) – incl. tunnel management. The resource management function shall also include sending resource reports to relevant entities.
3.16	Elaborated Data/Context Management	UMF shall give a method to access context base for obtaining, storing and updating context information.
3.6	Network Monitoring	UMF MUST provide means to monitor and evaluate for End-to-End (E2E) connection/session status/statistics.

Table 3-4: UC3 top 5 functional requirements

3.1.4 Use Case 4

3.1.4.1 Importance of UC4 problems

UC4 problems (P4.1-3) are listed below. Problems P4.2 and P4.3 are identified as the more important ones for accommodating - resolving UC4 Functional Requirements.

P4.1 - Design of distinct SON functionalities in network nodes to efficiently self-configure and self-optimize network resources. The SON functionalities at a given node (e.g. base station) should allow self-adapting to varying operation conditions, in the presence of other self-organizing neighbouring nodes, to assure stability and scalability.
P4.2 -Design of different SON functionalities operating simultaneously to achieve one or several performance objectives. The solutions should guarantee coordinated operation of the SON functionalities, while avoiding or solving conflicts between conflicting objectives.
P4.3 - Govern radio access networks by means of high-level policies triggering coordinated SON functionalities. Definition of objectives and rules in the different network levels from the OAM down to the SON algorithms embedded in the radio access nodes. Monitor the full SON processes to provide assurance.

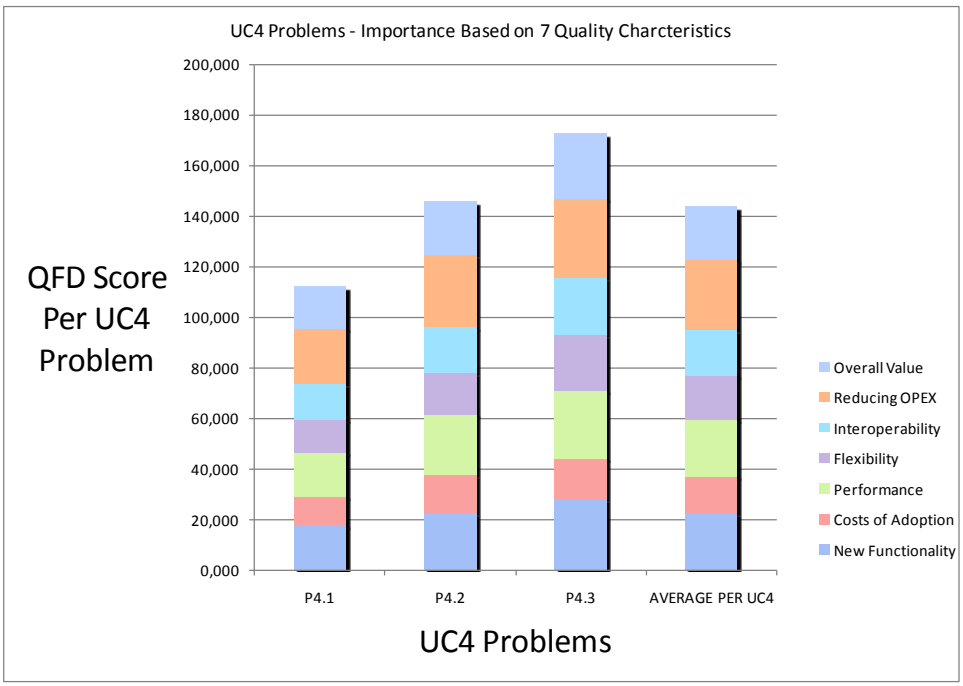

Figure 3-8: QFD score per UC4 problem

3.1.4.2 UC4 Requirements contribution

FR4.9, FR4.4 and FR4.3 (in decreasing order of importance)*, are likely to contribute most in solving UC4 problems:*

FR4.9 UMF shall provide rules/policies for operating self-X entities (to identify the involved self-X entities; to allow the interaction between self-X entities, to define hierarchy between SON

functionalities, to activate / deactivate self-X functions, to transfer high level goals to low level operation of self-X entities; to provide information to - and from self-X algorithms; and between self-X and other network entities; to coordinate between SON functionalities (the policy in this case corresponds to utilities, weights and target thresholds for KPIs), to resolve conflicts between running self-X processes when coordination fails)

FR4.4 UMF shall provide interface for communication between SON entities and between SON entities and the network.

FR4.3 UMF shall provide interface between governance tools and self-X entities, to allow inserting, modifying, interacting with and monitoring self-X processes.

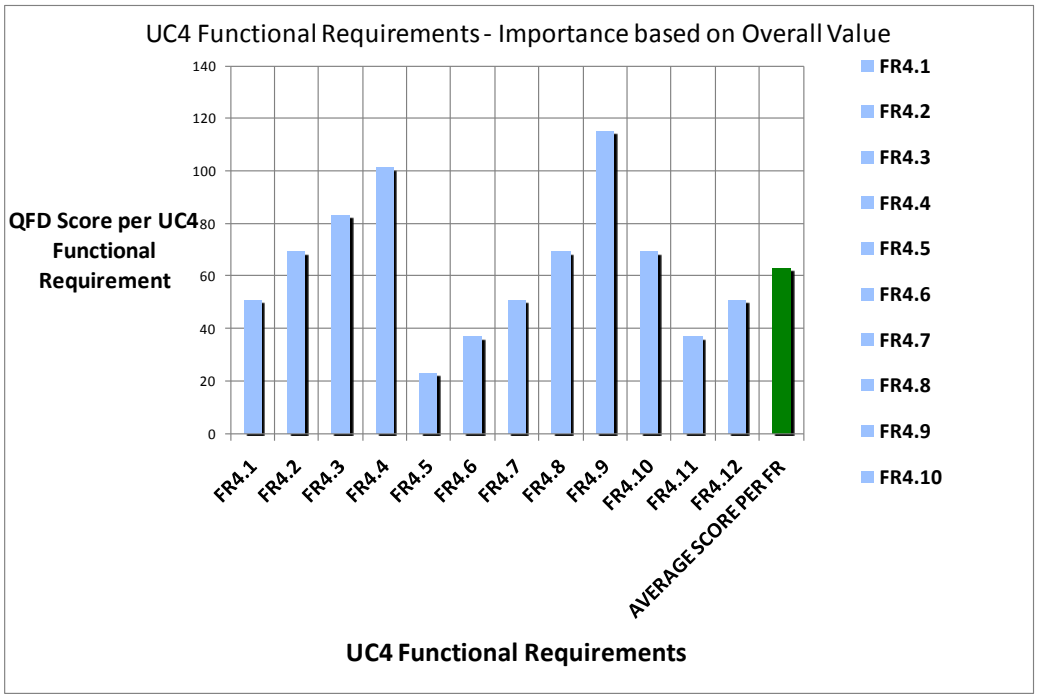

Figure 3-9: QFD Analysis of UC4

3.1.5 Use Case 6

3.1.5.1 Importance of UC6 problems

UC6 has been divided in the following 8 problems of table 3-5.

PROBLEMS/NEEDS	PROBLEMS/NEEDS (lower level of details)
P6.1 - Setting the business goal	According to the trigger, the operator defines business goals/policies, in high-level terms. Policies are then derived according to the higher-level goals, to provide constraints and priorities. The derived policies are assessed against existing goals/policies so as to identify and resolve conflicts (in fact, conflicts can arise if the defined goal/objective/policy are antagonist with respect to previous goals or the impact of these goals on already deployed services). In legacy systems, there is not such human-to-network interface that will be used to introduce the business level goals in high level terms and leave the system to autonomously work out the situation and meet the objectives.

P6.2 - Analysing the business request	The inputs/requirements derived from the business entry need to be elaborated and need to be correlated together with pertinent knowledge stemming from user & service raw data so as to derive technology specific (network specific) requirements. Today, services and networks are managed separately. In addition, the translation of service requirements into network requirements is manual i.e. it is decided and deployed by humans. As a general practice, people in charge of development (R&D + affiliate team) + planning will consider these needs and make assumptions regarding the traffic engineering, the related SLAs and will define the related network and IS requirements.
P6.3 - Determination of candidate solutions	This problem concerns the determination (reasoning with) of the candidate solutions (networks) that can satisfy the derived network requirements. The candidate technologies and networks, which can contribute to this satisfaction need to be, discovered also taking into account existing knowledge that was extracted from raw network data to knowledge. The legacy situation is that network requirements are sent to the operational teams in charge of the network segments (planning, assurance). These operations are in general manually done, with static processes and without considering any accumulated knowledge. Eventually, over provisioning can be seen as the current assumption.
P6.4 - Invocation of RAN	This problem tackles with the invocation of the selected RANs and the request for an offer in terms of the quality, which the RAN can provide. RAN investigates way to accommodate the request (anticipated load). In OFDMA-based (LTE) case this may result in the solution of problems such as radio resource allocation, Admission/Congestion control and scheduling, relay selection in case of multi-hop networks, link positioning, compensation by means of SON mechanisms etc. The situation today would have been resolved by communicating the request (e.g. by calling) to the respective RAN administrator. The respective RAN administrator is responsible for the management of the targeted RANs and RAN elements and has to carry out an estimate of the available network resources, considering a given operating point of the network, typically the worst case. The types of management of and interaction with RANs differentiate because of the existence of completely different RRM mechanisms, but also due to multi-vendor types of elements that have diverse and often proprietary information/data models.
P6.5 - Invocation of backhaul/core segment - The backhaul/core segment is triggered and the general problem is to find the best configuration and accordingly offer of quality, so as to support the solution (offer) provided previously by RAN. This process can further be split into 5 actions.	**P6.5.1** - The backhaul/core investigates way to accommodate the request - At the backhaul side, this may involve LSP configuration in IP/MPLS case. At the core side, it also involves GW (e.g., SGW, PDN-GW) (re)selection/configuration, GW migration/dimensioning. Looking to the current practices, taking into account backhaul/core network aspects while allocating traffic and QoS into wireless access points is not the legacy situation.

	P6.5.2 - Communication with administrator of backhaul/core segments - In particular, the description of traffic demands is sent to the operational teams who are in charge of the network segments (planning, assurance) and if needed the teams interact with elements/network in the backhaul/core segments by using CLI-based remote logins (rlogin) or SNMP-based request/responses in order to obtain the current status that will be used for their offline calculations/estimations.
	P6.5.3 - Sharing of network configuration and costs - The objective of this functionality is to identify potential reconfiguration to be enforced in the network. Equipment, in an individual basis, needs capacity data values to calculate a path (or a partial path) and the cost of this path. To realize this type of operation, equipment needs to emit requests towards other equipment in order to obtain the required complementary information (or part of the final answer). Equipment could be located in a different network layer and might use different protocols. Another criteria could be added to evaluate reconfiguration impact in the network (available resources, need multi-step reconfiguration for limit interruption of service)
	P6.5.4 - Computation of new connectivity - Though this functionality the system is taking the decisions of the atomic re-configuration options, taking into account the options and the related costs
	P6.5.5 - Recovery ready reconfiguration - In a multi-operator environment where heterogeneous networks operate in a cooperative manner, different protocols are applied in the network layer. Nowadays different networks use widely spread network architectures based on different protocols (i.e. MPLS). These protocols have some inherent disadvantages that affect their performance. For this reason, Traffic Engineering (TE) is applied in a way to optimize several parameters of the protocols. Since networks evolve in such a way, the overall system performance and stability becomes more and more indispensable and hard to preserve (i.e. end-to-end QoS). For this reason new TE optimization techniques should be devised. In this area we will investigate algorithms and techniques to optimize the parameters of legacy protocols. Possible approaches include simple optimization techniques or more sophisticated i.e. evolutionary algorithms.
P6.6 - Achievement of coherence	The problem here is to resolve possible incompatibilities between the offered QoS from RANs and backhaul/core segments, respectively. For that reason, some sort of negotiation and cooperation between segments is needed that will be used to fine-tune the resulting offers from the underlying segments, in order to achieve coherence. As of today, interaction of administrative/management domains (physical interaction between administrators) is definitely needed for achieving coherence between the offers from both RAN and backhaul/access networks, however such

	synchronization of all the participating segments takes too much time.
P6.7 - Configuration	**P6.7.1** - Actual configuration of the RAN and Core Network nodes according to the configuration determined - Currently the network configuration requires human intervention for the setting of every configuration parameter. This implies that the configuration may result to be very time and resource consuming.
	P6.7.2 - Setting reconfiguration commands - This functionality will apply the reconfiguration decision. First, it's necessary to identify concerned equipment and request each of them to perform the appropriate reconfiguration actions. Then, each of the targeted equipment has to translate and enforce the decision while taking into consideration the appropriate protocol and configuration parameters. Regarding the implementation of this functional block, there are several options that has to be discussed and assessed (centralized, distributed, hierarchical, etc.).
P6.8 - Assurance	Having configured the network, continuous monitoring is needed for collecting measurements (i.e. Performance Measurements and/or UE measurements) in order to ensure that the desired QoS level is guaranteed during the operational phase of the service. Actions can be triggered in order to adjust the network configuration parameters following the traffic and network conditions. Currently the performance analysis is done by periodically activating collection of measurements. The results are elaborated offline and manual changes of the network configuration are performed. In addition, the assurance processes tackled within this problem provide feedback to the H2N governance GUI and not to the NMS, which is the typical situation today.

Table 3-5 : UC6 Problems

A QFD analysis of these problems revealed the results of Figure 3-10. According to this analysis, when the "Overall Value" is considered, the 3 problems that are likely to contribute most in UC6 in descending order are P6.1, P6.5 and P6.4.

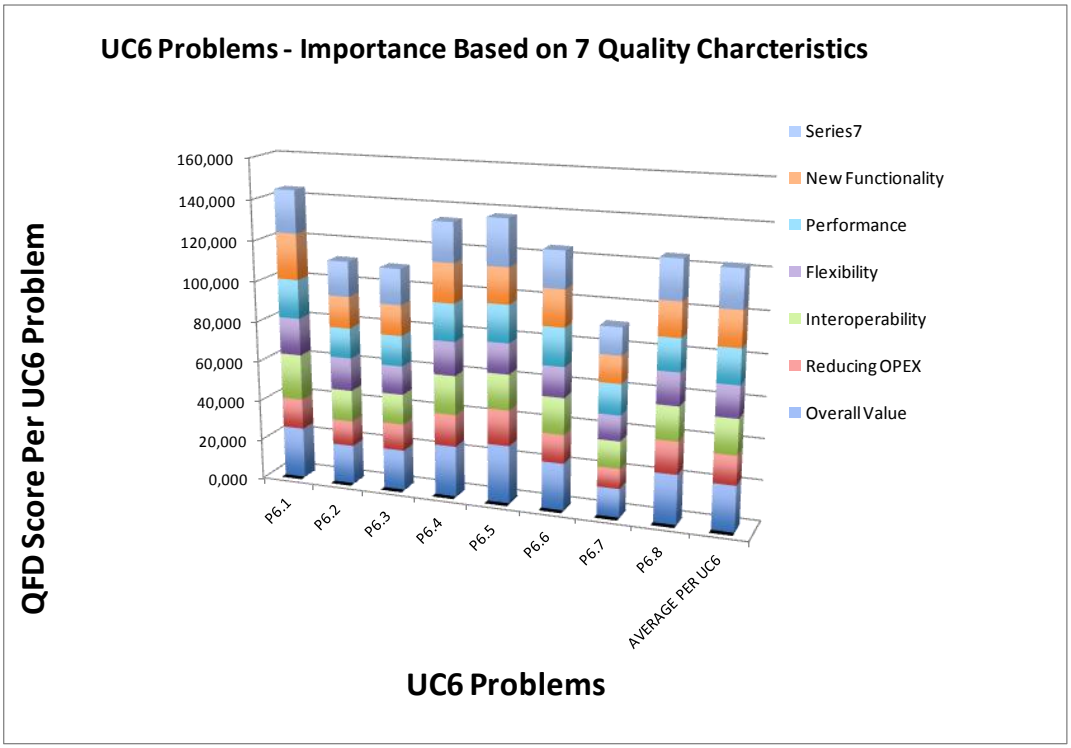

Figure 3-10: QFD Analysis per UC6 problem

3.1.5.2 UC6 Requirements contribution

The results of the QFD analysis of UC6 for each functional requirement can be found in Figure 3-11. According to this analysis, when the "Overall Value" is considered, the requirements that are likely to contribute most in solving UC6 problems are:

- FR6.16: UMF compliant systems should provide building and UMF should provide storage/retrieval/dissemination of information and knowledge on SLAs, Applications, User classes, RAN (network, resources and configuration), Backhaul/Core (traffic measurements, bandwidth estimations, network configurations), Traffic mobility requirements, and Traffic demand descriptions.
- FR6.15: UMF should provide policy models (network policies, routing update policies …).

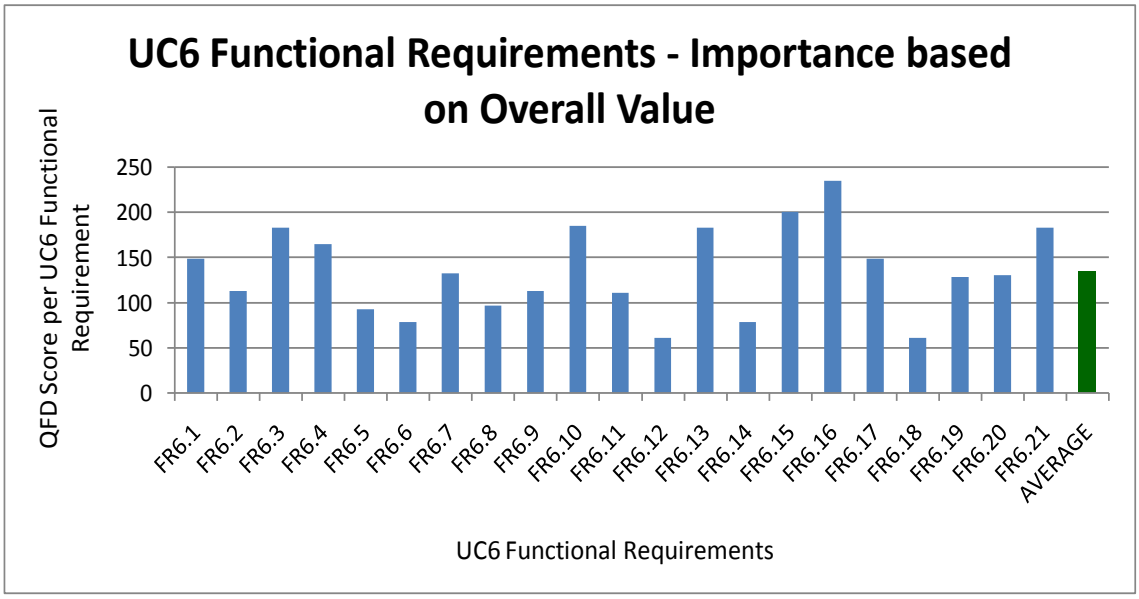

Figure 3-11: QFD Analysis of UC6

3.1.6 Use Case 7

This section summarizes the main results of the QFD analysis for the Network and Service Governance Use Case. Use case 7 has been refined into the following problems:

PROBLEMS/NEEDS	**PROBLEMS/NEEDS (lower level of details)**
Problem UC7-1	To provide the human operator with a mechanism for expressing the network management objectives in a high-level business language, without the need of highly specialized knowledge. Derivation of network policies from the business goals through the use of semantic techniques.
Problem UC7-2	Evaluation of the network governance tool, in terms of examining whether the generated policy rules and the applied configuration actions meet the initial business requirements. This evaluation will be realized through a feedback loop procedure that will realize the following actions: a) evaluation of the applied configuration actions in the infrastructure part and generated policy rules, and b) evaluation of the business requirements through examining how well the specific goal is met.
Problem UC7-3	Implementation of algorithms so that the network elements in FTTH environments can self-discover their context, through the use of network protocols.
Problem UC7-4	Implementation of self-monitoring algorithms in network elements in FTTH environments.
Problem UC7-5	Probabilistic self-Diagnosis functions should be implemented in the network elements, based on their own state and their operational context.
Problem UC7-6	Decision making processed based on semantic models and inference engines must be supported for self-healing purposes.

Table 3-6 : UC6 Problems

3.1.6.1 Importance of UC7 problems

Two problems are emerging as very important:

- Problem P7.1: To provide the human operator with mechanisms for expressing the network management objectives in a high-level business language, without the need of highly specialized

knowledge. Derivation of network policies from the business goals through the use of semantic techniques.

Problem P7.2: Evaluation of the network governance tool, in terms of examining whether the generated policy rules and the applied configuration actions meet the initial business requirements. This evaluation will be realized through a feedback loop procedure that will realize the following actions: a) evaluation of the applied configuration actions in the infrastructure part and generated policy rules and b) evaluation of the business requirements through examining how well the specific goal is met

The impact of the problems in the accommodation of the functional requirements is shown in the following figure:

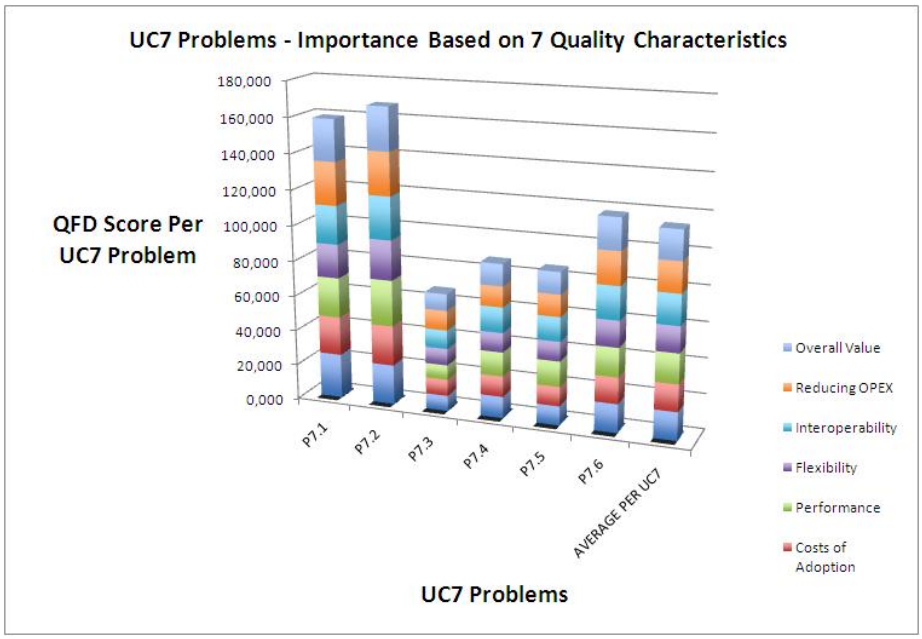

Figure 3-12: QFD score per UC7 problem

3.1.6.2 UC7 Requirements contribution

The analysis shows that FR7.7, FR7.8 and FR7.19 requirements contribute most in solving UC7 problems if ''Overall Value'' is considered, while FR7.9 and FR7.18 contribute least in solving UC7 problems if ''Overall Value'' is considered. The most impactful requirements are summarized here:

FR7.7: UMF must incorporate a policy language to provide information and allow communication between the autonomic entities.

FR7.8: UMF must define interfaces between governance tools and autonomic entities, to allow the insertion, modification, and dissemination of policies.

FR7.19: UMF must define interfaces between autonomic entities and governance tools to allow the communication of collected monitoring, information, notification of results of diagnosis processes, notification of self-healing actions and notification of alarms.

The following figure presents the overall importance of the functional requirements:

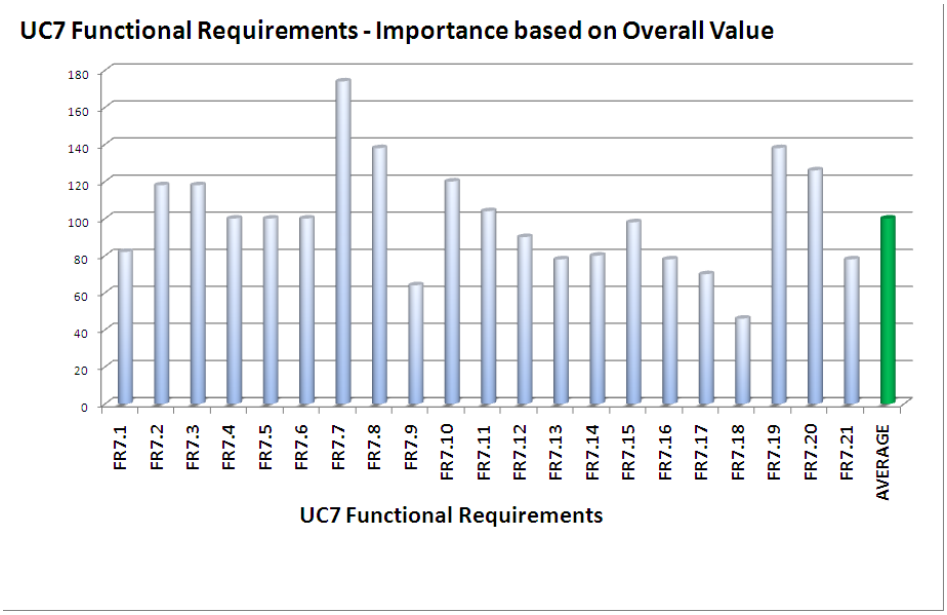

Figure 3-13: QFD score per UC7 Functional Requirement

In conclusion, priority is to first focus on the policy framework that enables the translation from business to infrastructure objectives. The QFD requirement analysis shows that communication between governance and autonomic entities supporting self-management needs to be considered also in a first stage, supporting the policy-related requirements.

4 Exploitation

4.1 Project Internal Use

4.1.1 Towards WP2

D4.1 provided the first report on the synthesis of the functional, non-functional and business requirements derived from the project use cases. This deliverable consolidates these requirements and provides the refined and the additional requirements that have emerged from the analysis of the 2nd burst of the use cases. Continuing the work started in D2.1 [16], WP2 will need to enrich the design of UMF by addressing these additional requirements.

Moreover, as reported in this deliverable, a QFD analysis has been performed for each use case in order to prioritize the use case problems and requirements. The prioritization dictated by the QFD analysis should be considered in the first complete specifications in the upcoming D2.2 i.e. ensuring that the respective prioritized requirements are addressed in the UMF design. More specifically, after having a thorough look at the QFD analysis above, in the majority (if not all) of the use cases, the results recommend to assign priority in requirements related to policy (governance), orchestration/coordination and knowledge. This is to be taken into account in the next specification of UMF, particularly in the detailed design of the mostly impacted Functional Blocks (FBs) among the ones identified in D2.1 [16], namely Governance FB and Policy Derivation & Management FB, Cooperation FB, and Information & Knowledge Building FB.

In addition, a set of non-functional requirements is also put under scrutiny in this deliverable (see also section 2.2.2) including scalability, trustworthiness, stability and convergence. The work on WP2 and in particular on the design of UMF is very important here. More specifically, the project develops solutions aimed at solving the distinct problems in each of the use cases, the so-called Network Empowerment Mechanisms (NEMs). NEMs are designed and deployed by targeting a network segment or service infrastructure and with the specific purpose to solve an operational problem and to achieve a performance objective. Then, it is the role of UMF and generally of work in WP2 to enable the integration and interworking of NEMs within the operator's management ecosystem and ensure at the same time that a set of non-functional requirements that are stated in this deliverable and are key to the project, and its stakeholders, are fulfilled.

4.1.2 Towards WP3

WP3 approaches the use cases listed in this deliverable from a methods perspective, i.e. WP3 tries to provide the right tools to meet the functional and non-functional requirements. A detailed mapping between the use cases and the WP3 task forces has already been provided in D4.1. In-depth results of methods that are suitable for the use cases can be found in deliverables D3.1-D3.4. Here, non-functional requirements have also been put in the focus, e.g. the convergence time and the stability of SON functions (use case 4). These non-functional requirements can at times be competing – for instance, the convergence time can be shortened at the cost of stability and vice versa. Eventually, finding the right balance between competing objectives requires some subjective judgment by operators, which makes this topic strongly related to trust management (treated in D4.3). Furthermore, this deliverable (D4.2) already sets the use case requirements in the context of the UMF. From a WP3 perspective, the classification into functional and non-functional requirements will later be complemented by a classification into methods that are necessary for the UMF functional blocks, namely governance, coordination, and knowledge, and methods that are independent from the functioning of the UMF.

4.1.3 Towards WP4 – Task 4.3

The introduction of the QFD methodology in the project inspired the idea of exploring the possibilities for application of the method in relation to the business modelling work to be executed in task 4.3. The idea of scoring requirements and thereby prioritising them seems promising to apply to business criteria as well, with the objective of analysing what the business impact of the system will be. In particular, we will link the requirements with the twelve parameters from the business-modelling matrix by [11]. This section gives an introduction to how this method is applied within T4.3 and what the expected results and next steps are. This will be detailed in Deliverable 4.7.

4.1.3.1 Background

In the QFD analysis performed by the consortium, for each use case, the problems were matched with the functional and non-functional requirements, giving them scores ranging from 1 (low or no impact) to 5 (high impact). This way, the importance, and thus the priority, of the individual problems and requirements are quantified so they can be compared and prioritised.

This has sprung an interest from task 4.3, *Impact Analysis*, since if such a methodology would be applicable to business requirements and standard business (model) choices, it could prove to be very useful in determining which problems and requirements would be the most essential from a business perspective and on which business aspects the system will have an impact given the current set of problems and requirements. In other words, it could show where the business strengths of the system are, e.g. proving unique selling points or solving 'gaps' or inconsistencies in the current business model, given the currently formulated set of requirements.

In particular, we were interested in linking the QFD methodology to a fixed set of business modelling choices for future-internet systems, namely the twelve parameters provided in the business model configuration matrix by [11]. These business-model design parameters encapsulate the dimensions of value creation on the one hand (which relates to aspects such as the value proposition and the financial model), and the dimension of control on the other hand (relating to the outset of the value network and the functional architecture). The parameters are presented in Table 4-1.

Control parameters		**Value parameters**	
A. Value network parameters	**B. Functional architecture parameters**	**C. Financial model parameters**	**D. Value proposition parameters**
A1. Combination of assets	B1. Modularity	C1. Cost (sharing) model	D1. Positioning
A2. Vertical integration	B2. Distribution of intelligence	C2. Revenue model	D2. User involvement
A3. Customer ownership	B3. Interoperability	C3. Revenue sharing model	D3. Intended value

Table 4-1: The business model design parameters by [9]

4.1.3.2 Application

We created a matrix for scoring the functional and non-functional requirements to the business model design parameters. Since the requirements per use case vary, this exercise will have to be performed for every use case separately. The scoring entails that a point is given in case the requirement impacts the said design parameter. When counting the points per design parameter, we can gain some insights into the importance of that parameter given the requirements of a specific use case.

In the figures below, we applied this to use case 6 (*Operator-governed, end-to-end, autonomic, joint network and service management*) and use case 7 (*Network and Service Governance*). These two use cases have been selected as the first candidates for our analysis, as at the time of work all use cases were still under development and these two seemed the most stable in an early stage. Eventually, the analysis will be applied to all use cases, which is planned for Deliverable 4.7.

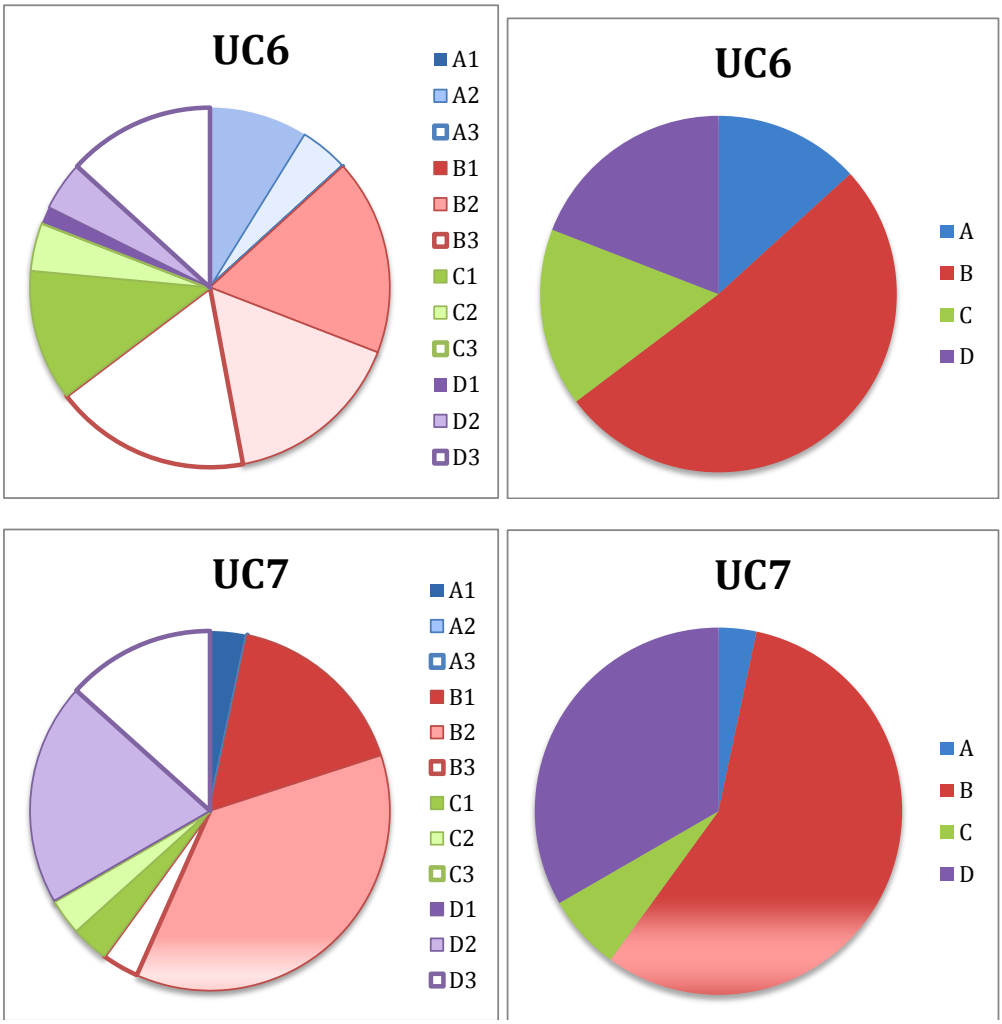

Figure 4-1: Scoring of the use-case requirements on the business-model design parameters

The left pie charts indicate the scores, and thus the impacts of the requirements on the different design parameters. The parameters are shown by their abbreviations, which correspond to the parameters in Table 4-2

Control parameters		**Value parameters**	
A. Value network parameters	**B. Functional architecture parameters**	**C. Financial model parameters**	**D. Value proposition parameters**
A1. Combination of assets	B1. Modularity	C1. Cost (sharing) model	D1. Positioning
A2. Vertical integration	B2. Distribution of intelligence	C2. Revenue model	D2. User involvement
A3. Customer ownership	B3. Interoperability	C3. Revenue sharing model	D3. Intended value

Table 4-2 - Parameters reported (with abbreviations) in figure 4.1

The right pie charts are aggregations, which show the scores of the design-parameter categories rather than the individual parameters.

What becomes clear here is that for these two use cases the impacts on the business model design parameters differs. Use case 6 has a high impact on the functional-architecture side of the business model, where use case 7 also has a strong impact on the value proposition, mainly caused by the impact on *User Involvement*. The impact on the value network and the financial parameters are much smaller in use case 7.

4.1.3.3 Next steps

This analysis represents a first attempt at applying the QFD methodology in the context of business impacts. Future work will include (1) analysing how to fit the business requirements in this method — whether they are requirements and thus inputs, or rather additional and case-specific business impacts that get scored by the inputs, (2) applying the method to the other use case as well and updating the new requirements for these two use cases, and (3) distilling a thorough analysis from this exercise. This is work that will be presented in the deliverable for task 4.3, Deliverable 4.7, *Analysis of the impact of deployment of autonomic networking functionalities*.

4.2 External Exploitation

The functional, non-functional and business requirements have been identified and reported in the D4.1 deliverable issued in July 2011. Deliverable D4.2 presents the final overall synthesis that consisted of the refinement, the prioritization and, in some case, the extension/amendments of the initial requirements. One of the main targets of this analysis is to gain relevant design information to feed the further project activities. Indeed, the primary purpose of the bottom-up analysis and refinement of use case requirements is to enrich the UMF design process and focus the technical reference problems to be addressed by the network empowerment mechanisms.

Besides this, another important exploitation of the identified use case requirements is their use to feed the project standardization activities. The scope of this exploitation in standards, which concerns the requirements themselves (and not the expected project outcome resulting from these requirements), is twofold: (1) driving the work of selected standardization groups with relevant requirements in order to strengthen the project impact; and (2) juxtaposing the project and external requirements for completion purposes.

The first targeted group is the ETSI AFI which is an Industry Specification Group aiming to develop pre-standard specifications for Autonomic Network Engineering for the Self-Managing Future Internet [12]. The first work item (WI#1), currently active within this group, targets the description of scenarios, use cases, and definition of requirements for the Autonomic/Self-Managing Future Internet [13]. Scenarios and use cases are intended to reflect real-world problems, which can benefit from the application of Autonomic/Self-Management principles. This purpose is in line with the use case definition, requirement identification, refinement and the overall synthesis that have been achieved within the UniverSelf project and reported in this document. The current release of WI#1 provides a rather consolidated set of scenarios and requirements that have been identified by ETSI AFI stakeholders. It covers auto-configuration, fault management, monitoring, coordination of multiple self-* mechanisms, among other functions; and it addresses legacy and emerging technologies. In this context, the plan for the exploitation of the UniverSelf use case requirements towards ETSI AFI consists of two steps that shall be taken:

- First, the refined requirements extracted from UniverSelf use cases will be juxtaposed and thoroughly compared with the current AFI list of requirements in order to identify potential gaps.
- Then, based on this analysis, a contribution will be prepared and provided in order to enrich WI#1 and, by the same, to participate in driving the second AFI work item (WI#2 - Generic Autonomic Network Architecture), and the technology-specific work items (WI#3 documents) on applicability of the generic architecture to specific environments.

Besides ETSI AFI, and as part of the overall project dissemination plan, potential contributions towards other relevant standardization groups are continuously investigated and discussed. These initiatives include:

- **NMRG Group at IRTF:** As stated in its charter, the Network Management Research Group (NMRG) provides a forum for researchers to explore new technologies for the management of the [14]. Besides, NMRG has recently updated its charter, which offers an opportunity to influence shaping (self-) management solutions for future networks on topics such as safe configurations, stability or trust with appropriate problem statement and requirements drafts.
- **3GPP:** The requirements addressing novel SON functionalities, SON coordination and related policy management, which have been identified within UniverSelf use cases (e.g. mainly UC4), are in the scope of future 3GPP releases.
- **ITU-T SG 13**: It is the lead study group for future networks and NGN within ITU-T [15]. Therefore, the requirements addressing novel 'Future Networks' (FN) functions and related management operations, which have been identified within UniverSelf use cases (e.g. mainly UC2 and UC3), are in the scope of future releases of FN recommendations by this study group.

5 Conclusion

Deliverable D4.2 provided a synthesis of the Use Case requirements derivation and analysis. In particular, starting from the list of requirements initially reported in the report D4.1, the analysis has been extended and refined; moreover a prioritization of Use Case problems and related requirements has been made by using the QFD methodology. D4.2 reports the lessons learnt (in requirements derivation and analysis) and the related internal and external exploitations.

Main lessons learnt have been realizing the importance of pursuing a clear distinction between Industry's needs/requests (with the related priorities) and functions/features needed to satisfy the needs/requests, both in the short-medium and long term. In this sense the adopted approach, using use cases as descriptors of a set of precise problems to be solved, led to a reasoned prioritized list of problems and related requirements reflecting Network Operators needs (e.g. reducing network OPEX, exploiting new revenue streams and improving the return on investment for network equipment and infrastructures).

Concerning internal exploitations, WP2 is already using deliverable D4.2 results to enrich the design of UMF, by addressing the requirements enchantments. Prioritization dictated by the QFD analysis will be considered in the first complete specifications in the upcoming D2.2 i.e. ensuring that the respective prioritized requirements are addressed in the UMF design. For example the majority of use cases recommend assigning priority in requirements related to policy (governance), orchestration/coordination and knowledge. This will to be taken into account in the next specification of UMF.

Regarding WP3, internal exploitations have the goal to provide the right tools to meet the deliverable D4.2 functional and non-functional requirements. A detailed mapping between the use cases and the WP3 task forces has already been provided in D4.1. In-depth results of methods that are suitable for the use cases can be found in deliverables D3.1-D3.4. Moreover, from a WP3 perspective, the classification into functional and non-functional requirements will later be complemented by a classification into methods that are necessary for the UMF functional blocks, namely governance, coordination, and knowledge, and methods that are independent from the functioning of the UMF.

Concerning external exploitations, D4.2 will be used to feed the project standardization activities. The scope of this exploitation in standards, which concerns the requirements themselves (and not the expected project outcome resulting from these requirements), is twofold: (1) driving the work of selected standardization groups with relevant requirements in order to strengthen the project impact; and (2) juxtaposing the project and external requirements for completion purposes. The first targeted group is the ETSI AFI which is an Industry Specification Group aiming to develop pre-standard specifications for Autonomic Network Engineering for the Self-Managing Future Internet. Besides ETSI AFI, and as part of the overall project dissemination plan, potential contributions towards other relevant standardization groups (e.g. NMRG Group at IRTF, 3GPP and ITU-T SG13) are continuously investigated and discussed.

QFD analysis has also attracted the interest of WP4-Task 4.3 (Impact Analysis). Applicability of such methodology to business requirements and standard business (model) choices could be very useful in determining which problems and requirements would be the most essential from a business perspective and on which business aspects the system will have an impact given the current set of problems and requirements. Specifically Task 4.3 linked QFD methodology to a fixed set of business modelling choices for future-internet systems, namely the twelve parameters provided in the business model configuration matrix by [11]. Results of this work will be presented in Deliverable 4.7 (*Analysis of the impact of deployment of autonomic networking functionalities)*.

References

[1] UniverSelf D4.1 deliverable “Synthesis of Use Case Requirements 1st release”

[2] Akao, Y., ed. (1990). Quality Function Deployment, Productivity Press, Cambridge MA.

[3] Becker Associates Inc, http://www.becker-associates.com/thehouse.HTM and http://www.becker-associates.com/qfdwhatis.htm

[4] Hauser, J. R. and D. Clausing (1988). "The House of Quality," The Harvard Business Review, May-June, No. 3, pp. 63-73

[5] Lowe, A.J. & Ridgway, K. Quality Function Deployment, University of Sheffield, http://www.shef.ac.uk/~ibberson/qfd.html, 2001

[6] Mizuno, S. and Y. Akao, ed. (1994). QFD: The Customer-Driven Approach to Quality Planning and Development, Asian Productivity Organization, Tokyo, Japan, available from Quality Resources, One Water Street, White Plains NY.

[7] Rosenthal, Stephen R, Effective product design and development, How to cut lead time and increase customer satisfaction, Business One Irwin, Homewood, Illinois 60430, 1992

[8] Reilly, Norman B, The Team based product development guidebook, ASQ Quality Press, Milwaukee Wisconsin, 1999

[9] Sullivan, L.P., 1986, "Quality Function Deployment", Quality Progress, June, pp 39-50.

[10] IETF RFC2119 Key words for use in RFCs to Indicate Requirement Levels, http://tools.ietf.org/html/rfc2119.

[11] Ballon, Pieter (2007) Business modelling revisited: the configuration of control and value. Info 9(5), pp. 6-19.

[12] ETSI AFI Group, Website: http://portal.etsi.org/portal/server.pt/community/afi

[13] ETSI GS AFI 001 V1.1.1 (2011-06), Group Specification, Autonomic network engineering for the self-managing Future Internet (AFI); Scenarios, Use Cases and Requirements for Autonomic/Self-Managing Future Internet

[14] Network Management Research Group (NMRG), Website: http://irtf.org/nmrg

[15] ITU-T Study Group 13 (SG 13), Website: http://www.itu.int/ITU-T/studygroups/com13/index.asp

[16] UniverSelf Project, Deliverable D2.1 UMF Specifications Release 1, July 2011

Abbreviations

3GPP	3rd Generation Partnership Project
3GPP LTE	3GPP Long Term Evolution
3GPP SAE	3GPP Service Architecture Evolution
AFI	Autonomic network engineering for the self-managing Future Internet
AP	Access Point
API	Application Programming Interface
BoF	Birds-of-a-Feather
BSS	Business Support System
CAPEX	Capital Expenditures
DiffServ	Differentiated services
DoW	Description of Work
E2E	End-to-End
EMS	Element Management System
eNodeB	Evolved NodeB
ETSI	European Telecommunications Standards Institute
FG-FN	Focus Group – Future Networks
FMC	Fix Mobile Convergence
FTTH	Fiber To The Home
GUI	Graphical User Interface
GW	Gateway
H2N	Human-to-Network
ICT	Information and Communication Technologies
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
IRTF	Internet Research Task Force
IMS	IP Multimedia Subsystem
IP	Internet Protocol
IRTF	Internet Research Task Force
IS	Information System
IT	Information Technology
ITU	International Telecommunication Union
ITU-T	International Telecommunication Union – Telecommunications standardization sector
KPI	Key Performance Indicator
LCCN	Learning-Capable Communication Networks
LE	Large Enterprises
LSP	Label Switched Path
LTE	Long Term Evolution
LTE-A	LTE – Advanced
MPLS	Multi- Protocol Label Switching
NaaS	Network as a Service
NMRG	Network Management Research Group
NMS	Network Management System
OAM	Operations Administration and Maintenance
OFDM	Orthogonal Frequency-division Multiplexing
OFDMA	Orthogonal Frequency-Division Multiple Access

OPEX	Operational Expenditures
OSS	Operations Support System
PDN-GW	Packet Data Network Gateway
QFD	Quality Function Deployment
QoE	Quality of Experience
QoS	Quality of Service
ROI	Return of Investment
RAN	Radio Access Network
RRM	Radio Resource Management
SGW	Serving Gateway
SME	Small and Medium Enterprises
SLA	Service Level Agreement
SON	Self Organised Networks
TCO	Total Cost of Ownership
TMF	TeleManagement Forum
UC	Use-Case
UMF	Unified Management Framework
VoIP	VoIP - Voice over IP
VPN	Virtual Private Network

Definitions

Functional Requirement – it is a description of what a system is supposed to do and it defines of a function, or a feature of a system, or its components, capable of solving a certain problem or replying to a certain need/request. The set of functional requirements present a *complete* description of how a specific system will function, capturing every aspect of how it should work before it is built, including information handling, computation handling, storage handling and connectivity handling.

System Design - a plan for implementing functional requirements.

Non-functional requirement – it is a specification criteria that can be used to judge the operation of a system, rather than specific behaviours; it is a description of how well a system performs its functions; it represents an attribute that a specific system must have. The non-functional requirements are controlled by other aspects of the system.

Business requirements – it is a description in business terms of what must be delivered or accomplished to provide value.

Business Opportunities - main opportunities and relevance of introducing a product/service in the market.

Business Bottlenecks – any bottlenecks that may impact the business adoption of a new product/system/service by the market.

System boundaries / limits define the constraints and freedoms in controlling the system. Limits can be determined by analysing how the behaviour of the system depends on the parameters that drive the system. Some limits would lead to unexpected and significant behaviour changes of the system, for example the unpredictable boundaries or changes in the scale of magnitude. Some other limits are determined by non-common behaviour interactions between the components of a system.

System Architecture - a plan for implementing non-functional requirements within the system limits/boundaries. It is conceptual model that defines the structure, behaviour, and a number views of a system within the system limits

Quality Function Deployment (QFD) –*a method for developing a design quality translating the consumer's demand into design targets and major quality assurance points to be used throughout the production phase.*

Use Case (UC) – it is a descriptor of a set of precise problems to be solved. It describes steps and actions between stakeholders and/or actors and a system, which leads the user towards a value added or a useful goal. A UC describes what the system shall do for the actor and/or stakeholder to achieve a particular goal. Use-cases are a system modelling technique that helps developers determine which features to implement and how to gracefully resolve errors.

Network Governance – a framework, which enables operators to describe their goals and objectives, through high-level means and govern their network. Includes the derivation of network policies from the business goals through the use of semantic techniques.

Stakeholder - a person, group or organization with an interest in something.

Viewpoint - It is a representation of a whole system from the perspective of a related set of concerns.

Accessibility - *the degree to which a system, device, service, or environment is available to as many people as possible. Accessibility can be viewed as the "ability to access" and benefit from some system or entity.*

Availability - the degree to which a system is in a specified operable and committable state at the start of a task. It is the proportion of time a system is in a functioning condition.

Certification – it refers to the confirmation of certain characteristics of an object, element of system. This confirmation is often, but not always, provided by some form of external review, assessment, or audit.

Configuration – It is a function establishing and maintaining consistency of a system and/or its performance. It is changing system's functional and physical attributes with its non-functional requirements, design, and operational information throughout its life.

Compliance - the conformance to a rule, such as a specification, policy, standard or regulation.

Extensibility - the ability to extend a system and the level of effort and complexity required to realize an extension. Extensions can be through the addition of new functionality, new characteristics or through modification of existing functionality/characteristics, while minimizing impact to existing system functions.

Interoperability - the ability of diverse systems and subsystems to work together (inter-operate)

Interoperability is a characteristic of a system, whose interfaces are completely understood, to work with other systems, present or future, without any restricted access or implementation.

Maintainability is a characteristic of design and installation, expressed as the probability that an element of a system will be retained in or restored to a specified condition within a given period of time, when the maintenance is performed in accordance with prescribed procedures and resources.

Operability - the ability to keep a system in a safe and reliable functioning condition, according to pre-defined operational requirements.

Performance - it describes the degree of performance of a system (according to certain predefined metrics, e.g. convergence time)

Privacy - the ability of system or actor to seclude itself or information about itself and thereby reveal itself selectively.

Resilience - the ability to provide and maintain an acceptable level of service in the face of faults and challenges to normal operations.

Reliability - the degree to which a system must work. Specifications for reliability typically refer to stability, availability, accuracy, and maximum acceptable/tolerable bugs.

Robustness is the ability of a system to cope with errors during execution or the ability of a system to continue to operate despite abnormalities in input or in environment context.

Safety - being protected against different types and consequences of failure, error harm or any other event, which could be considered non-desirable.

Serviceability refers to the process and ability to install, configure, and monitor systems, identify exceptions or faults, debug or isolate faults to root cause analysis, and provide hardware or software maintenance in pursuit of solving a problem and restoring the system into service.

Scalability - the ability of a system to handle growing amounts of work or usage in a graceful manner and its ability to be enlarged to accommodate that growth.

Supportability - a system's ability to be easily modified or maintained to accommodate usage in typical situations and changing scenarios. For instance, how easy should it be to add new blocks and/or subsystems to the support framework.

Security - the ability to prevent and/or forbid access to a system by unauthorized parties.

Usability - the ease with which a system performing certain functions or features can be adopted and used.