# Deliverable D4.6

# Synthesis of deployment results

| | | |
|---|---|---|
| **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 July 2012 | |
| **Actual delivery date** | 17 October 2012 | |
| **Lead beneficiary** | FT | Christian Destré, christian.destre@orange.com |

| | |
|---|---|
| **Authors** | ALBLF: Leila Bennacer, Benoit Ronot, Laurent Ciavaglia<br>FT: Zwi Altman, Christian Destré<br>INRIA: Remi Badonnel, Martin Barrere, Olivier Festor<br>NEC: Johannes Lessmann, Zarrar Yousaf, Paulo Loureiro<br>NKUA: Eleni Patouni, Vangelis Kosmatos, George Katsikas, Roi Arapoglou, Kostas Chatzikokolakis, Alex Apostolidis, Nancy Alonistioti<br>TI: Antonio Manzalini<br>TID: Beatriz Fuentes<br>UNIS: Majid Ghader<br>UPRC: Panagiotis Demestichas, Kostas Tsagkaris, Giorgios Poulios, Vasilis Foteinos, Aimilia Bantouna, Panagiotis Vlacheas, Vera Stavroulaki, Yiouli Kritikou, Dimitris Kelaidonis, Marios Logothetis, Dimitris Karvounas, Andreas Georgakopoulos, Louisa Papadopoulou, Assimina Sarli, Evangelia Tzifa<br>UT: Ramin Sadre<br>VTT: Teemu Rautio, Jukka Mäkelä, Petteri Mannersalo |

# Executive summary

This deliverable presents the synthesis of intermediate deployment results.

Making the deployment assessment of our solutions is not obvious. Lot of facets need to be considered, including implementation maturity, performance or assessing how our solutions could be deployed in (quasi-) real networks. Thus, this deliverable is describing our understanding on how to realise the deployment assessment of the project developments. We identify a first list of criteria based on ISO/IEC 9126-1 in order to structure and help to guide this assessment.

The implementation of UniverSelf solutions is based the use case lifecycle which follows the development of a solution from its conception up to its final and complete validation. Then, the solutions have different levels of implementation maturity depending on their relative progress, complexity and understanding. Some solutions are implemented at the level of a mechanism (e.g. the methods developed and evaluated in WP3), based on simulations and thus not yet fully integrated in a UMF-compliant management system. Other solutions are implemented according to the UMF release 2 specifications, referring to solution composed of UMF core functions and NEMs (e.g. in the WP2 and WP3), based on emulation and/or test-bed. Results are then described for each use case depending on their implementation level. They are based on available, implemented and tested solutions.

For each use case, solutions are presented including the solved problems and the related objectives. An evaluation is fulfilled to show what the solution benefits are, focusing on performance, novelty and/or added-value. Then a first (or intermediate) deployment assessment is done according to the identified list of criteria.

In the last section, we start to discuss the deployment assessment of UMF as a framework. An illustration of how to map the ensemble of UMF core functions and NEMs over a 3GPP-LTE system is described (i.e. a technology-specific instantiation exercise).

Lot of results are already part of this deliverable: they cover various problems and network technologies. As part of the project progress, higher implementation level will be available in the next months, beginning with the Year 2 review demonstrations (November 2012). The consolidation of the deployment results will be done in the next release of the deliverable, D4.12 (June 2013).

# Table of Content

# Foreword

This document provides a first synthesis of simulation, prototyping, experimentation and demonstration results (outcome of task 4.2). Deliverable D4.6 represents a first report on the implementation of the key UniverSelf solutions, mechanisms and algorithms: proof-of-concept environment are described, including the constituent simulation modules and software and hardware components. Results are also compared with vendors' and operator's performance metrics and/or a global autonomic metric.

As a synthesis of the first phase of activities of task 4.2, this report has a number of relationships with other project activities covering:

- The WP2 UMF design and specifications highlighting the link with deliverable D22 - UMF Specifications, release 2 and the ongoing investigations on the UMF deployment scenarios;
- The development and evaluation of methods in the WP3 highlighting the link with deliverables D3.5, D3.6/D3.7 and D3.8 which report on the innovative methods, their individual evaluation and their relationships with UMF. D46 discuss on the deployability of the same solutions (NEMs) from a different, complementary angle aiming at providing the first elements and guidelines for the deployment of the solutions developed by the project;
- The other tasks of the WP4 addressing the impact and migration of the solutions such as task 4.3 and the link with the deliverable D4.7 and the trust and certification aspects studied in task 4.4 highlighting the link with deliverable D43 where trust and certification appears as the first step towards adoption and deployment of autonomic systems and networks.
- The report contains also links with other activities done in the scope of task 4.2 such as the deliverables D44/D45 and D48/D49, respectively the first and second use case prototypes (and their associated leaflet).

This first release of the report aims at an intermediate deployment assessment of the solutions developed by the project. Therefore it is normal that the assessments reported here present different level of maturity and completeness. A second release of the report (deliverable D4.12, June 2013) will provide a complete and comprehensive analysis of the project solutions feasibility and readiness for deployment.

# 1 Introduction to UNIVERSELF deployment results

## 1.1 General introduction

This deliverable is part of the UniverSelf Work Package 4 (WP4) activities. WP4 is responsible for implementing and developing UniverSelf solutions complying with UMF specifications, supporting methods and algorithms developed in WP3 (NEMs) and solving the identified problems. As part of these experimentation and validation works, this deliverable focuses on assessing the solutions deployment i.e. how the solutions we are developing and implementing are feasible and deployable from a system point of view. It corresponds to identify the gap from our research project solutions to industrial solutions and how our solutions are contributing to the project goals. Many facets need to be considered, including:

- Assessing how far our solutions could be deployed in real networks. UniverSelf solutions, as they exist when this deliverable is edited, are mostly software-based and it is important to identify all the elements that are necessary for supporting real network elements and/or existing legacy management tools.
- Assessing industrial impact of our solutions. Our solutions need to be part of a migration path from legacy management system(s) to future network management system(s), contributing to the business drivers.
- Assessing the implementation state of our solutions. By implementation state, we refer to implementation maturity. It is related to functionality compliance and technology/problem coverage. Dedicated implementation criteria (e.g. open and extensible code) need to be considered. Trust and solution certification need also to be considered in this context.
- Assessing solution performance. Solution performance is related to the way and efficiency problems are solved. It is also related to how the solutions behave in a non-optimal or difficult situation.
- Assessing autonomic novelty. Deployment also consists in answering to autonomic challenges, managing stability, scalability, levels of automation and performance.

This deliverable is the first release of the deployment results. In the next sections, we present the methodology and the criteria we chose to structure the assessment and then, first results are presented based on the project use cases (as described in deliverables D41 and D42).

## 1.2 Making deployment assessment in UniverSelf

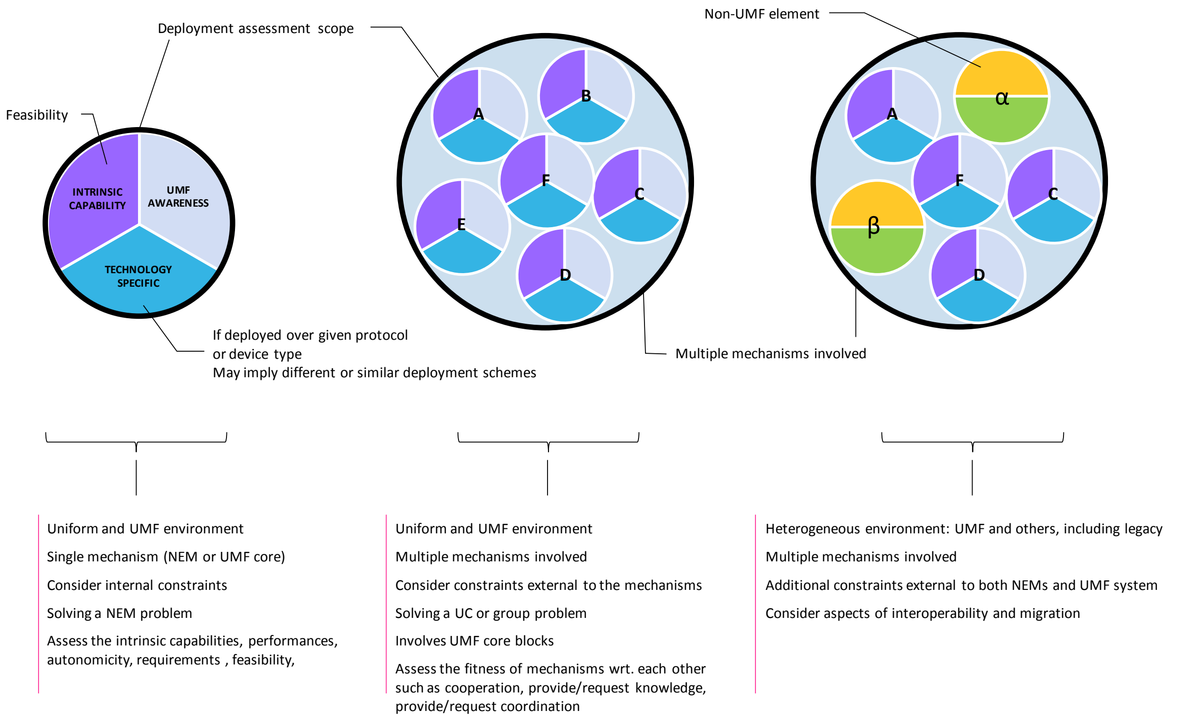
The implementation of UniverSelf solutions is following the use case life cycle. Several levels of implementations exist in the project (see Figure 1):

- Mechanism implementation level. As part of WP3, methods and algorithms are developed and implemented to solve set of problems coming from use cases. Depending of the nature of the methods and problems, it corresponds to a single Network Empowerment Mechanism (NEM[1]) or a single UMF Core mechanism (e.g. coordination mechanism). At this level of implementation we can only make individual deployment assessment, which relies mainly on simulations. The assessment focuses on performance, functionality coverage as there is no consideration of (external) management system.
- UMF core and NEM implementation level. It corresponds to an UMF system composed of the core blocks (Governance, Coordination, and Knowledge) and a set of NEMs. Deployment assessment can be done at both NEM and management system levels. At this level of implementation, prototypes built on emulation and testbeds are available. Those prototypes allow assessing the deployment of multiple NEMs and core mechanisms together and simultaneously.

[1] A NEM is defined as: a functional grouping of objective(s), context and method(s) where "method" is a general procedure for solving a problem. A NEM is (a priori) implemented as a piece of software that can be deployed in a network to enhance or simplify its control and management (e.g. take over some operations). An intrinsic capability of a NEM is to be deployable and interoperable in a UMF context (in a UMF-compliant network).

- Management system implementation level. It corresponds to a full management infrastructure that may consist in a UMF system and additional non-UMF elements: legacy management system, management tools, non-UMF supported network elements etc...

First results in this deliverable are related to the first two levels of implementations. The third level is not yet covered and we plan to consider it in the next release of deliverable with an analysis.


Deployment assessment scope
Non-UMF element
Feasibility
INTRINSIC CAPABILITY
UMF AWARENESS
TECHNOLOGY SPECIFIC
A
B
F
C
E
D
α
A
F
C
β
D
If deployed over given protocol
or device type
May imply different or similar deployment schemes
Multiple mechanisms involved

Uniform and UMF environment

Single mechanism (NEM or UMF core)

Consider internal constraints

Solving a NEM problem

Assess the intrinsic capabilities, performances, autonomicity, requirements , feasibility,

Uniform and UMF environment

Multiple mechanisms involved

Consider constraints external to the mechanisms

Solving a UC or group problem

Involves UMF core blocks

Assess the fitness of mechanisms wrt. each other such as cooperation, provide/request knowledge, provide/request coordination

Heterogeneous environment: UMF and others, including legacy

Multiple mechanisms involved

Additional constraints external to both NEMs and UMF system

Consider aspects of interoperability and migration

**Figure 1: Deployment assessment levels and scopes**

## 1.3 List of criteria for deployment assessment

In order to make our deployment assessment, we identify the need to have common criteria. Their choice is complex and we identify several options. A first option is to re-use the existing non-functional requirements as defined in deliverables D4.1 and D4.2 to assess how far our solutions are compliant with them. The main issue is that non-functional requirements are tightly related to use case problems and they do not clearly identify how far our solutions are deployable or the properties that such solutions should verify. They are also, for some, very specific to our project preventing a simple external assessment or comparison.

The next option is related to the RFC 5218 - What makes for a successful protocol [17]. Several factors contributing to or hinder a protocol's success are identified. For example, some of the success factors are:

- Incremental deployability.
- Open code availability.
- Freedom from usage restrictions.
- Open specification availability.
- Open maintenance process.
- Extensible.
- No Hard scalability bound.

Clearly these factors are closer to our deployment assessment target but these factors are defined in a protocol context and the gap or differences with our service and network self-management solution is to be further investigated.

This leads us to our third option coping with the previous mentioned limitation: the ISO/IEC 9126-1 internal/external quality model [23].

It is one of the most widespread standards for software product quality and evaluation. The goal of this standard is the definition of a multi-level hierarchy quality model. Such model is based upon generic software characteristics that are further decomposed into sub-characteristics. The latter are then analysed in software attributes, which can be quantified using specific metrics. It should be noted that the ISO/IEC 9126-1 standard distinguishes two different aspects of the quality model: a) the internal and external quality and b) the quality in use. The first aspect of the model focuses on the definition of six software characteristics and their high level decomposition to the sub-characteristics level. The internal quality is defined using the set of attributes that can be measured during the development process whereas the external aspect is measured using the testing process. The quality in-use aspect is measured using the quality-in-use attributes; the latter reflect the user view on the quality level. Table 1 illustrates the ISO/IEC 9126-1 internal/external quality model.

| *Characteristics* | *Subcharacteristics* |
|---|---|
| Functionality | Suitability |
| | Accuracy |
| | Interoperability |
| | Security |
| | Functionality Compliance |
| Reliability | Maturity |
| | Fault Tolerance |
| | Recoverability |
| | Reliability Compliance |
| Usability | Understandability |
| | Learnability |
| | Operability |
| | Attractiveness |
| | Usability Compliance |
| Efficiency | Time Behaviour |
| | Resource Utilisation |
| | Efficiency Compliance |
| Maintainability | Analysability |
| | Changeability |
| | Stability |
| | Testability |
| | Maintainability Compliance |
| Portability | Adaptability |
| | Installability |
| | Co-existence |
| | Replaceability |
| | Portability Compliance |

**Table 1: The ISO/IEC 9126-1 internal/external quality model**

This model was extended to support autonomic computing characteristic [1]. The quality model for autonomic computing assessment is shown Figure 2, Table 2 and Table 3.

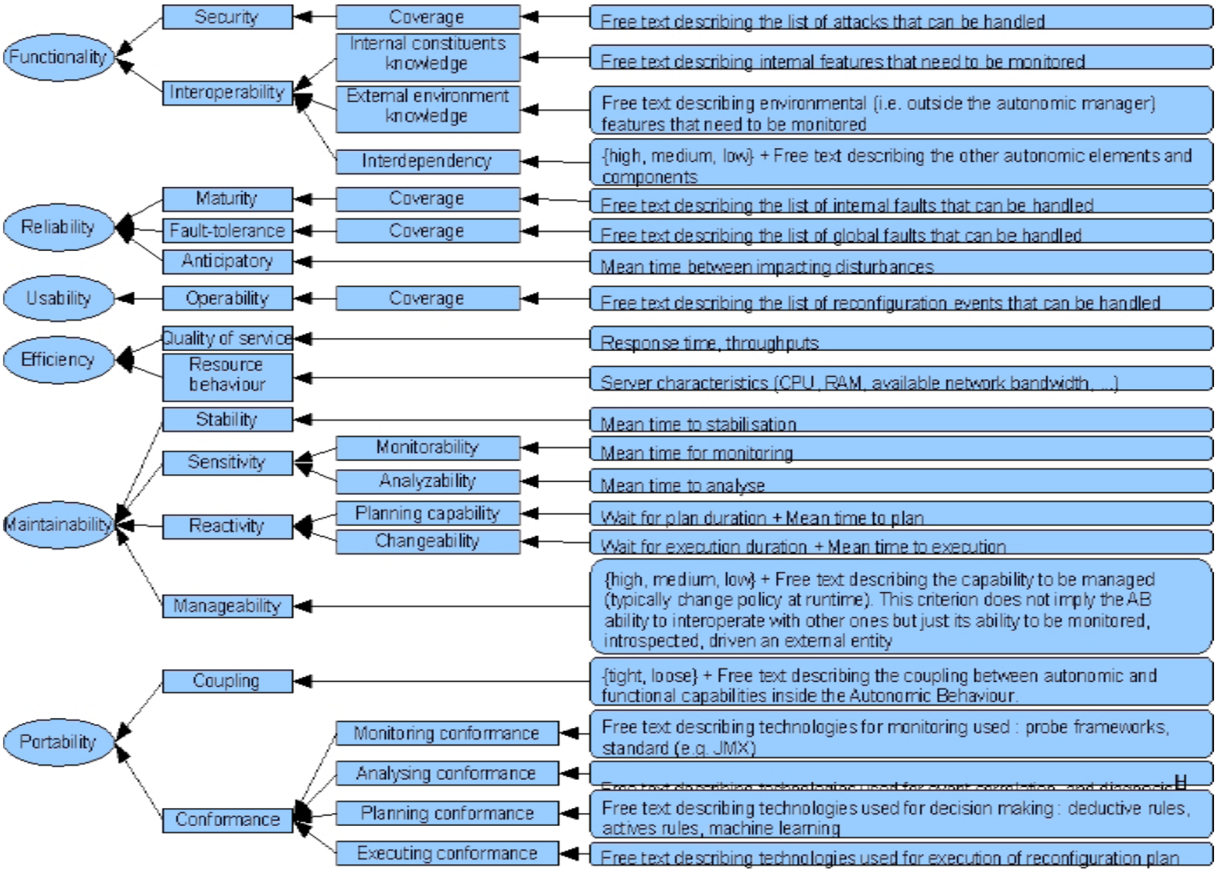

Functionality
Security
Coverage
Free text describing the list of attacks that can be handled
Internal constituents knowledge
Free text describing internal features that need to be monitored
Interoperability
External environment knowledge
Free text describing environmental (i.e. outside the autonomic manager) features that need to be monitored
Interdependency
{high, medium, low} + Free text describing the other autonomic elements and components
Reliability
Maturity
Coverage
Free text describing the list of internal faults that can be handled
Fault-tolerance
Coverage
Free text describing the list of global faults that can be handled
Anticipatory
Mean time between impacting disturbances
Usability
Operability
Coverage
Free text describing the list of reconfiguration events that can be handled
Efficiency
Quality of service
Response time, throughputs
Resource behaviour
Server characteristics (CPU, RAM, available network bandwidth, ...)
Maintainability
Stability
Mean time to stabilisation
Sensitivity
Monitorability
Mean time for monitoring
Analyzability
Mean time to analyse
Reactivity
Planning capability
Wait for plan duration + Mean time to plan
Changeability
Wait for execution duration + Mean time to execution
Manageability
{high, medium, low} + Free text describing the capability to be managed (typically change policy at runtime). This criterion does not imply the AB ability to interoperate with other ones but just its ability to be monitored, introspected, driven an external entity
Portability
Coupling
{tight, loose} + Free text describing the coupling between autonomic and functional capabilities inside the Autonomic Behaviour.
Conformance
Monitoring conformance
Free text describing technologies for monitoring used : probe frameworks, standard (e.g. JMX)
Analysing conformance
Free text describing technologies used for event correlation and diagnosis
Planning conformance
Free text describing technologies used for decision making : deductive rules, actives rules, machine learning
Executing conformance
Free text describing technologies used for execution of reconfiguration plan

**Figure 2: Quality model for autonomic computing assessment**

| *Criterion* | *Sub-criterion* | *Metric* |
|---|---|---|
| Sensitivity | Monitorability | Mean time for monitoring |
| | Analyzability | Mean time to analyze |
| Reactivity | Planning capability | Wait time for plan duration + Mean time to plan |
| | Changeability | Wait time for execution duration + Mean time to execution |
| Anticipation | | Mean time between impacting disturbances |
| Stability | | Mean time to stabilization |

**Table 2: Metrics for quantitative assessment of Autonomic behaviour**

| *Criterion* | *Criterion type* |
|---|---|
| Description of Autonomic Behaviour (AB) | Free text. Short description of the AB. |
| Related to self-x characteristic | {self-configuration, self-healing, self-optimization, self-protection} |
| Coverage | Free text describing the list of disturbances the AB is able to deal with |
| Interdependency | High, medium, low + Free text. Description of the other ABs and components the current AB is depending on. |
| Internal constituents knowledge | Internal features that need to be monitored for this AB |
| External environment knowledge | Environmental (i.e. outside the AB) features that need to be monitored for this AB |
| Level of automation | {-, M, MA, MAP, MAPE} |
| Monitoring compliance | Free text. Technologies used for monitoring: probe frameworks, standard (e.g. JMX) |
| Analyzing compliance | Free text. Technologies used for event correlation and Diagnosis |
| Planning compliance | Free text. Technologies used for decision making: deductive rules, actives rules, machine learning |
| Executing compliance | Free text. Technologies used for execution of reconfiguration plan |
| Coupling | Tight, loose + Free text. Description of coupling between autonomic and functional capabilities inside the AB. |
| Manageability | High, medium, low + Free text. Capability to be managed (typically change policy at runtime). This criterion does not imply the AB ability to interoperate with other ones but just its ability to be monitored, introspected, driven an external entity. |

**Table 3: Qualitative criteria for Autonomic behaviour assessment**

The evaluation of these model/metric/criteria is not available at the time of the deliverable edition. It will be considered in the next deliverable related to the deployment assessment.

The next section will present the description and high level assessment of our available solutions.

# 2 Deployment of key UniverSelf solutions

## 2.1 Introduction to key UniverSelf solutions

Since the beginning of project, we implemented and tested a lot of elements. This deliverable is focusing on the available results coming from implemented and tested solutions. Future results based on the on-going implementation works (and additional solutions) will be part of the next release of this deliverable (D4.12, June 2013). As discussed in section 1.2, available implementations are mainly focused on the mechanisms and UMF and NEM levels. The next sections are presenting the results of "key UniverSelf solutions" defined as available, implemented and tested simulations, emulations, or prototypes that are solving UniverSelf problems (as identified in deliverable D4.2). Results are presented according to the use cases and to three main subsections:

- The key solution description is describing the mechanisms/NEM and the related problems solved.
- The key solution evaluation is describing both results and how they are obtained (testbed or prototype description). Results are also discussed to demonstrate their added-value and how they are contributing to UniverSelf goals.
- Key solution deployment assessment is discussing the criteria as defined in section 1.3 in order to qualify the gap between the provided solution and its “real” deployment by an operator, in other words, making the assessment on how far the solution could be adopted by industry as it exists today, or considering the requirements, constraints and open issues faced by the solutions to be deployed, what is achieved by the solutions and what is necessary to be done to fulfil them.

## 2.2 Use Case 1

### 2.2.1 Use Case 1 key solution

Use case 1 is focusing on self-diagnosis and self-healing mechanisms. The available main results are related to the mechanisms for building knowledge and make reactive and proactive diagnosis. They correspond to the following NEMs:

- NEM2: Self-Diagnosis based on Bayesian Networks (BN) and Case Base Reasoning (CBR)
- NEM13.2: Anomaly detection.
- NEM14: Proactive Diagnosis of Congestion.
- NEM20: Self-Diagnosis based on network and service data.
- NEM21: Optimization of context acquisition and dissemination.
- NEM33: Proactive diagnosis based on pattern recognition.

They are mainly related to the first set of problems related to Self-Proactive/Reactive for network and services (see Table 4).

| *UC 1 problems* | | *Sub-problems/objectives* |
|---|---|---|
| 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 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-diagnosis/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 4: UC1 problems**

#### 2.2.1.1 NEM2: Self-Diagnosis based on Bayesian Networks and Case Base Reasoning

We propose a new hybrid approach combining Case-Based Reasoning (CBR) and Bayesian Networks (BN). According to literature, Bayesian Networks are currently the most powerful and popular diagnosis method. However, the complexity of inference in BN increases exponentially with the number of nodes. Hence, this technique is not suitable for large scale systems including a large number of components such as current and future networks with hundreds or thousands of elements. To overcome this limitation, we propose a combined case-based and Bayesian reasoning approach to improve the BN inference while keeping the advantages of BN technique, the resulting solution improves the degree of automation of the diagnosis process and requires less intervention of human expertise.

In order to optimize the diagnosis process, we consider only a subset of the BN structure, and, inside this subset, only the nodes where variations of the monitored parameters have been observed. This subset is identified thanks to Case-Based Reasoning. CBR is also used for the learning phase of our approach. CBR learning allows improving the process efficiency over time by accelerating the identification and resolution of previously encountered pathological cases. It addresses the P1.1.1, P1.1.3, and P1.1.5 problems.

#### 2.2.1.2 NEM20: Self-Diagnosis based on network and service data - NEM21: Optimization of context acquisition and dissemination

Self-diagnosis based on network and service data NEM is used in conjunction with Context acquisition and dissemination NEM in order to provide a reactive self-diagnosis mechanism (i.e. addressing Problem 1.1.3) to be applied to the core network. The former NEM exploits fuzzy logic and machine learning techniques while the latter uses data pre-processing and data mining ones, to achieve an adaptive and autonomic decision making framework for identifying QoS degradation on the core network. The two NEMs communicate with each other by exploiting UMF functionality for inter-NEM communication through the COORDINATION block which assures the non-conflicting operation of the mechanism when the operator's policies are met. The main contribution of the inter-NEM interaction is related to the high level of autonomicity that introduces, without human intervention needed, thus reducing OPEX and latency. Furthermore, since the deployment of the mechanism can be either centralised or distributed, there are no severe scalability issues since the proposed data clustering scheme uses a simple, thus effective classification scheme based on the Euclidean distance. Finally, another important aspect of this framework's operation is related to its adaptability; the feedback between the two NEMs strengthens the ability of each NE to adapt its state according to the sensed parameters.

#### 2.2.1.3 NEM13.2: Anomaly Detection

Anomaly-based intrusion detection systems aim to classify an activity as benign (normal) or malicious (anomalous) by comparing it with a model of normality. The sensitivity of such systems to anomalies controls which instances of traffic are flagged as anomalous. The sensitivity level usually depends on a set of system parameters. Such parameters play a central role, since there exists a natural trade-off between detecting all anomalies (at the expense of raising alarms too often), and missing anomalies (but not issuing any false alarms). In traditional and existing solutions, the task of tuning the system is carried out by the system manager or by expert IT personnel and it requires detailed knowledge of both the detection system as well as the network to be protected.

The core of the NEM is an autonomic approach that optimizes the parameters of the detection system toward high-level policies provided by the management environment, in this way replacing the manual tuning of the system parameters by the system manager or by expert IT personnel. In the studied example (see section 2.2.2.4), we allow the system manager to set the high level policy by defining the relative importance of correctly detected anomalies over false alerts. Mathematically, this can be described as an optimization problem over the true negative rate and true positive rate of the detection system and the system parameters.

#### 2.2.1.4 NEM14: Proactive Diagnosis/ Prediction of Congestion

The problem addressed by this mechanism is part of P1.1.2 and refers to congestion. More precisely the mechanism aims to empower the system to combine monitored context in order to predict a possible congestion on a specific link. Variables that were examined towards this direction are related either to network traffic, in terms of incoming traffic on the link in Bytes and/or packets and their respective trends, or to network and link capabilities such as buffer size of the node that releases traffic in the link, the queue of the same node, the link capability and the packet drops.

The core of the mechanism is a knowledge-based approach during which the mechanism identifies and learns the relations between the parameters that are used and a potential congestion. The knowledge is built using the unsupervised technique of Self-Organizing Maps (SOMs) and according to this knowledge the mechanism is then capable of proactively diagnosing/ predicting how close the link is to congestion. As the mechanism focuses on a specific link, no scalability issues are expected. Moreover, the autonomicity and the proactivity that the mechanism introduces are envisaged to reduce Operational Expenditure (OPEX), since less human intervention will be required in the network. Finally, the mechanism is also expected to enhance Quality of Service (QoS) and Quality of Experience (QoE) given the fact that network managers or an autonomic control loop of P1.2.1 will treat the predicted congestions before network degradation occurs and users experience it.

#### 2.2.1.5 NEM33: Proactive diagnosis based on pattern recognition

The NEM33 "Proactive diagnosis based on pattern recognition" aim is to predict the events that will occurs by the recognition of a small part of the pattern and estimate the future occurrence of events in time and with a statistical confidence. It addresses the P1.1.2 problematic.

The patterns can be of various kinds like network elements alarms, KPI, status metrics, etc..., from one kind to multiple combinations. The Artificial Neural Network (ANN) monitors the networks for each element of the patterns it is supposed to detect. If events describing a pattern are occurring the ANN will inform registered processes (NEMs through COORDINATION, GOVERNANCE, etc...), of the future occurrence of this event the confidence of the prediction, the next steps of the pattern and the time before occurrence.

The core of the NEM is an ANN that is designed to monitor one or several patterns. This design is specified in the pattern description as the data needed for its training. During the initialization step the training is performed. This setup step is necessary and will determine the ability of the ANN to correctly detect the pattern. The ANN, once trained with the whole pattern, is adjusted to react to the parameters which the pattern begins.

This ANN will then be connected in the network to the elements that contains the parameters of the pattern. The 'how to connect' is described in the pattern and each monitored parameters will be tested. If more than one parameter is not available the NEM will not instanced.

Once up and running the ANN will continuously monitor the network and keep informed of the evolutions of its predictions the other NEMS/Governance through the UMF mechanisms of its event predictions.

## 2.2.2 UC1 Key solution evaluation

### 2.2.2.1 NEM2: Self-Diagnosis based on Bayesian Networks and Case Base Reasoning

As described in deliverable D3.6, the evaluation of the self-diagnosis mechanism based on Bayesian Networks and Case Base Reasoning is reported below.

To create and simulate a network topology, a set of nodes is generated using the BNJ (Bayesian Network Tools in Java), an open-source suite software for BN. Each node is characterized by a marginal probability table, calculated from statistics collected from the observed network.

We evaluate and compare the performances of the combined CBR-BN approach to those of the classic BN technique. The evaluation results are organized according to a set of metrics, namely: Accuracy, Reliability and Speed.

**Accuracy**: This figure shows that, whatever the size of the original network, our NEM can precisely identify the root cause with greater accuracy than the reference approach of fault diagnosis.

| Number of nodes in BN | BN approach | **CBR-BN approach** |
|---|---|---|
| 40 nodes | 3 to 4 | **1** |
| 60 nodes | 3 to 4 | **1** |
| 80 nodes | 3 to 4 | **1** |
| 100 nodes | 4 to 5 | **1** |
| 200 nodes | 4 to 6 | **1** |
| 500 nodes | 6 to 8 | **1** |
| 1000 nodes | 15 to 21 | **1** |
| 2000 nodes | 25 to 40 | **1** |

**Table 5: Test of accuracy**

**Reliability**: According to the dataset, the confidence interval, in 95% of cases, is approximately (96.5, 98.7).

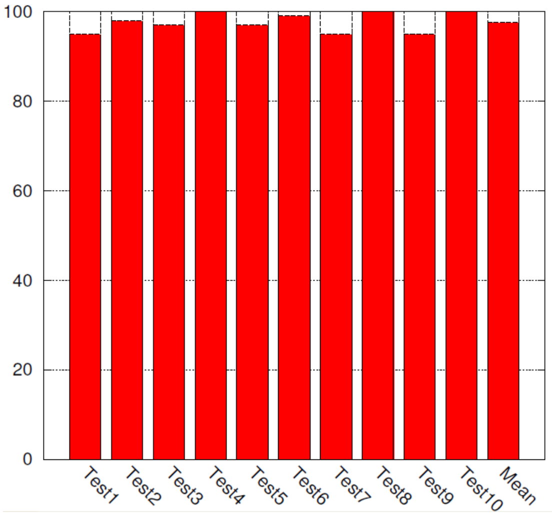


**Figure 3: Test of reliability**

**Speed**: The increase in the original network size has less impact on the detection time for the CBR-BN approach than for the BN approach (diverging curves).

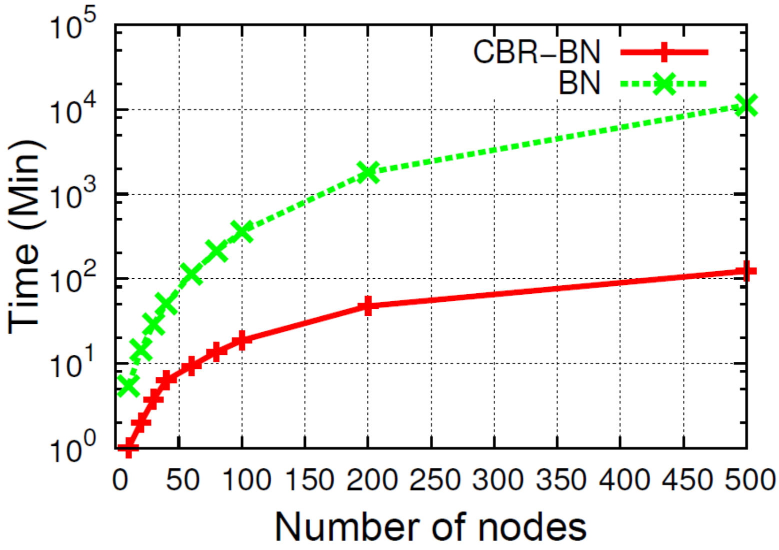


**Figure 4: Test of detection time**

The added-value of our NEM is essentially the reduction of the complexity which appears in a significant reduction in the number of nodes involved in the diagnosis process.

In order to illustrate this phenomenon, the following table details the number of nodes involved in the diagnosis process for the BN and CBR-BN cases. For networks greater than 40 nodes, the CBR-BN solution enables on average a reduction of the resulting network to a mere 1/10th of the original network size.

| Number of node in BN | Size of the case for 9 tests | | | | | | | | | **Median** |
|---|---|---|---|---|---|---|---|---|---|---|
| 10 | 2 | 2 | 3 | 2 | 3 | 3 | 3 | 3 | 2 | **3** |
| 20 | 2 | 3 | 4 | 4 | 4 | 3 | 5 | 3 | 4 | **4** |
| 30 | 4 | 4 | 3 | 2 | 3 | 4 | 4 | 4 | 5 | **4** |
| 40 | 4 | 5 | 6 | 3 | 4 | 5 | 3 | 5 | 5 | **5** |
| 50 | 5 | 5 | 4 | 6 | 5 | 5 | 5 | 4 | 6 | **5** |
| 80 | 7 | 8 | 5 | 7 | 5 | 6 | 5 | 7 | 7 | **7** |
| 100 | 10 | 9 | 10 | 11 | 10 | 11 | 11 | 10 | 10 | **10** |
| 200 | 23 | 21 | 21 | 18 | 19 | 16 | 21 | 20 | 22 | **21** |
| 500 | 46 | 54 | 49 | 53 | 59 | 42 | 54 | 49 | 62 | **53** |
| 2000 | 126 | 269 | 420 | 74 | 139 | 305 | 126 | 275 | 431 | **269** |

**Table 6: Reduction of network size using CBR-BN**

#### 2.2.2.2 NEM20: Self-Diagnosis based on network and service data

Starting with NEM20 "Self-Diagnosis based on network and service data", the goal is to allow the management system, using inputs from network elements (network and service data), to proceed in identification of QoS degradation events in IP networks. For the case study under consideration and for a VoIP service, delay, jitter and packet loss are identified as playing a significant role in QoS degradation, thus, using fuzzy logic, the event identifier evaluates the aforementioned inputs for every active session.

Initially, we configure the network elements with a generic membership function configuration and we compare with the developed ground truth. For the extraction of the ground truth, we use fuzzy reasoners that are built specifically for the environment of the data that we generated; thus we consider that when the data set is evaluated by them, the decisions will be correct. Using a generic configuration the self-diagnosis scheme achieves a success rate of 64%. Given the fact that such self-diagnosis scheme is built to operate adequately in all environments we consider this success rate as acceptable. Then, we apply the clustering adaptive algorithm in this self-diagnosis scheme and have a success rate of 70.01% (amelioration of 8.6%). Finally, by incorporating the statistical adaptive mechanism, we modify the input membership function. Therefore, the input states are being captured by new membership functions, which are being described by Gaussian distributions, with higher overlap areas. The success rate of the adapted scheme reaches 84% compared to the ground truth (amelioration 23.8%).

The main contribution of this process lies in the fact that the operator has the ability to automate the network management procedure and thus reduce the OPEX and improve the overall network performance by applying specific remedy actions after the QoS degradation identification.

#### 2.2.2.3 NEM21: Optimization of context acquisition and dissemination

Regarding NEM21 "Optimization of context acquisition and dissemination", it is based on Data Mining methods for processing, classifying and labelling a set of network parameters (e.g. Delay, Jitter, Packet Loss) as monitored by a network element. In particular, the NEM takes as input a large set of network measurements and produces a classification scheme according to the geometric characteristics of the input data. Furthermore, a QoS label is added to each measurement in order to address the network element’s state. The produced results can be exploited by the network in order to reduce the amount of the exchanged information among the devices, thus preserving resources.

The extracted information is significantly reduced compared to the initial data volume since the involved network elements will only exchange 12 tuples instead of 50093. Specifically, 4 tuples contain the cluster centroids and radius, while eight tuples are used for the cluster bounds representation (2 bounds per cluster). Considering that tuple size is 101.01 bytes, only 1212.12 bytes will be exchanged instead of 5.06MB, therefore the derived context reaches (0.0024%) of the initial volume.

Following this approach, the extracted context which is related to the geometric characteristics of each cluster, is very accurate since the successfully classified instances ratio is 100%. Instead of communicating the whole dataset, Normalization techniques’ contribution produces a very descriptive set of data that each NE can use/disseminate to describe its QoS state.

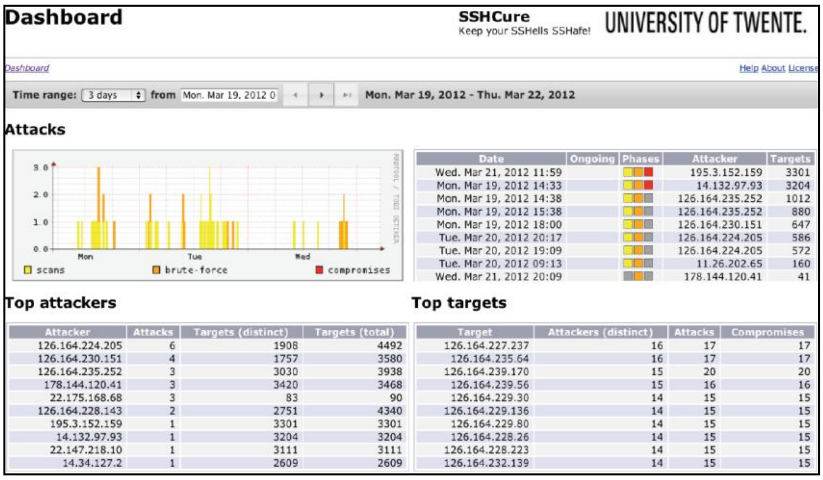
The results highlight the need of introducing novel data mining concepts in order to infuse cognitive capabilities to network elements. The dissemination of the aggregated context by each device in the network forms a resource-demanding process in terms of CPU, memory and high bandwidth for the communication among the network elements. For this reason, by applying the proposed Distributed Data Mining framework, the bulk network data can be replaced by a minimal thus meaningful context. The latter is able to adequately describe the whole dataset and preserve the resources required for exchanging all the aggregated measurements. Finally, if this mechanism is enhanced with learning capabilities (integration with NEM20) then important contribution towards the OPEX reduction will be also met.

#### 2.2.2.4 NEM13.2: Anomaly Detection

The evaluation has been performed in two parts. We have validated the optimization procedure for tuning the detection parameters by applying it to a simple Hidden-Markov-model-based intrusion detection system. The optimization procedure allows explicitly relating the system parameters and the performance measures in the form of an optimization problem. In our experimental system, the optimization problem consisted in maximizing the number of correctly classified observation sequences and the policy expressed the relative importance of detecting all the attacks versus keeping the false alarm rate low. Our validation showed that, by varying the relative importance, we are able to fine-tune the system to favour either the detection of all the anomalies or the detection of attacks only when they are certain. However, it should be taken into account that such a policy affects the system performance in multiple ways, that is, how timely the system is able to detect an attack or recover from it.

The choice of which would be a suitable policy is left to the system manager who has to select the policy according to their requirements, SLAs, etc. By embedding the NEM and its functionality into the UMF, the policy could not only be changed by the personnel, e.g., through a user interface, but also by other NEMs deployed in the network. Other NEMs would be able to change the policy based on their own observations and actions: For example, a NEM handling reconfiguration actions for the network could select a more anomaly-sensitive policy in order to better protect the network while the reconfiguration takes place, accepting the higher probability for false positives.

In the second part of the evaluation, we have aimed at replacing the simple detection model described above by a more complex model that is not only able to detect an attack but also to give more details on the attack, that is, in our experiments, the current phase of an SSH brute force attack or whether it was able to compromise a host. A first prototype implementation run on empirical data sets has shown that attacks and compromised hosts are identified accurately. The additional details on the attack provided by the system allow a system manager to better judge the consequences and nature of the attack (see screenshot in Figure 5). However, due to the complexity of the model, its integration into the above optimization procedure has not been done yet and requires further work.



Dashboard

SSHCure
Keep your SSHells SSHafe!

UNIVERSITY OF TWENTE.

Dashboard

Help About License

Time range: 3 days from Mon. Mar 19, 2012 0 Mon. Mar 19, 2012 - Thu. Mar 22, 2012

Attacks

| Date | Ongoing | Phases | Attacker | Targets |
|---|---|---|---|---|
| Wed. Mar 21, 2012 11:59 | | 🟨🟧🟥 | 195.3.152.159 | 3301 |
| Mon. Mar 19, 2012 14:33 | | 🟨🟧🟥 | 14.132.97.93 | 3204 |
| Mon. Mar 19, 2012 14:38 | | 🟨🟧▨ | 126.164.235.252 | 1012 |
| Mon. Mar 19, 2012 15:38 | | 🟨🟧▨ | 126.164.235.252 | 880 |
| Mon. Mar 19, 2012 18:00 | | 🟨🟧▨ | 126.164.230.151 | 647 |
| Tue. Mar 20, 2012 20:17 | | 🟨🟧▨ | 126.164.224.205 | 586 |
| Tue. Mar 20, 2012 19:09 | | 🟨🟧▨ | 126.164.224.205 | 572 |
| Tue. Mar 20, 2012 09:13 | | 🟨🟧▨ | 11.26.202.65 | 160 |
| Wed. Mar 21, 2012 20:09 | | ▨🟧▨ | 178.144.120.41 | 41 |

Top attackers

| Attacker | Attacks | Targets (distinct) | Targets (total) |
|---|---|---|---|
| 126.164.224.205 | 6 | 1908 | 4492 |
| 126.164.230.151 | 4 | 1757 | 3580 |
| 126.164.235.252 | 3 | 3030 | 3938 |
| 178.144.120.41 | 3 | 3420 | 3468 |
| 22.175.168.68 | 3 | 83 | 90 |
| 126.164.228.143 | 2 | 2751 | 4340 |
| 195.3.152.159 | 1 | 3301 | 3301 |
| 14.132.97.93 | 1 | 3204 | 3204 |
| 22.147.218.10 | 1 | 3111 | 3111 |
| 14.34.127.2 | 1 | 2609 | 2609 |

Top targets

| Target | Attackers (distinct) | Attacks | Compromises |
|---|---|---|---|
| 126.164.227.237 | 16 | 17 | 17 |
| 126.164.235.64 | 16 | 17 | 17 |
| 126.164.239.170 | 15 | 20 | 20 |
| 126.164.239.56 | 15 | 16 | 16 |
| 126.164.229.30 | 14 | 15 | 15 |
| 126.164.229.136 | 14 | 15 | 15 |
| 126.164.229.80 | 14 | 15 | 15 |
| 126.164.228.26 | 14 | 15 | 15 |
| 126.164.228.223 | 14 | 15 | 15 |
| 126.164.232.139 | 14 | 15 | 15 |

**Figure 5: Screenshot of the detection system's report screen**

### 2.2.2.5 NEM14: Proactive Diagnosis/ Prediction of Congestion

As already explained, the mechanism is based on knowledge building. The knowledge is built using SOMs and of course some data were also used for obtaining the knowledge, i.e. for training the relations of the data to the mechanism. The evaluation of the mechanism was performed using unseen data, i.e. data that had not been used during the training of the mechanism. In particular, those unseen data were used as "triggers" for which the mechanism had to predict if congestion of the link is to be experienced or not. As soon as the predictions were received from the mechanism, they were compared to the ground truth, i.e. the real state to which the link was entering under the circumstances described by the data. The percent of correct predictions was the metric used for evaluation.

In the absence of real data, the data that were used (for both the training of the mechanism and its evaluation) were obtained by simulating a network topology on the Network Simulator 2 (NS-2). The main traffic of that topology came from VoIP services and TCP and UDP packets. After multiple tests, the best combination of parameters for predicting congestion of a specific link proved to be the combination of a) the Incoming Bytes, b) the trend of the Incoming Bytes, c) the Link Capability and d) the Buffer Size. When this combination was used the percent of correct predictions that was received was equal to 86.6%.

#### 2.2.2.6 NEM33: Proactive diagnosis based on pattern recognition

As described before this NEM is based on the automation of 1) creating a context-specific core ANN, 2) training and adjust its detection parameters (thresholds), 3) connect to the network elements for real-time monitoring. These 3 steps have been tested and evaluated on a virtual network with a 2 parameter pattern based on the CPU temperature and usage. These tests successfully demonstrate the feasibility and efficiency of such technology. However they do not constitute a rigorous scientific proof: a concrete use case with complex pattern and its data set for design/training/evaluate was missing.

An application on alarm prediction is actually under development to demonstrate and validate this NEM.

### 2.2.3 UC1 Key solution deployment assessment

#### 2.2.3.1 NEM2: Self-Diagnosis based on Bayesian Networks and Case Base Reasoning

The feasibility and deployability of the NEM Self-Diagnosis based on Bayesian Networks and Case Base Reasoning (CBR-BN) is assessed with respect to the following criteria:

**Configurability**: the Alpha parameter.

The solution proposed presents an interesting and convenient characteristic regarding configuration parameters: there is only one single parameter that determines the performance threshold of the solution. This is definitely an advantage as once it is ready to be deployed; the operator only has to set-up one parameter. In the present context, this parameter is called 'alpha'.

In the diagnosis inference process, when we perform a statistical test, we refer to the probability of "mistakenly rejecting our hypothesis" as "alpha". The error rate increases with the increase of the value of alpha. However, a small alpha value (i.e. below 0.05) requires more measurements and the computation time grows respectively in an exponential way. The idea here is to find a compromise for the alpha value. The common value used for this parameter is equal to 0.05, which provides very good performance results in terms of speed, accuracy and reliability (see previous section on evaluation of the solution) without detrimental effects on the quality of the diagnosis or on the required computation resources or convergence time.

The alpha parameter is the single parameter to define. It is simple and easy to understand and set.

**Scalability**

The fault diagnosis approach must support networks composed of tens of thousands of interconnected elements. The CBR-BN solution is not dependent on the number of nodes contained in the network under study. Indeed, this solution can be applied on large networks without impacting negatively its overall efficiency. This characteristic is achieved thanks to the use of the case approach, where the problem (symptom) is first identified and the related problem-case is determined. The problem-case represents a sub-set of the initial Bayesian network under study. The reduction of complexity realized through this process is important (as highlighted in the results above) and allows scaling the approach to network topologies including several thousands of elements without affecting negatively the performance of the solution (the solution performance for speed scales linearly).

This is also a very important feature of the solution for its deployment as the constraints in terms of segmentation/partitioning of the network topology under the watch of the mechanism is non-impacting and the operator is free to deploy the solution based on its own internal constraints (e.g. administrative or technology domains). Another advantage is the capability to perform end-to-end diagnosis over large scale topologies providing additional benefit to the operator in the search of root causes which can be very remotely connected.

**Flexibility/dynamicity**

The removal or the addition of nodes in the network does not affect the performances of the CBR-BN approach. The latter ensures stable performances, despite network dynamicity in terms of topology changes.

The update of the model can be carried out either on a periodic basis or on an event basis when new equipment is added in the asset inventory. This strategy is left open to the operator to decide.

- Periodically: by analyzing periodically logs, new dependencies may occur. Model construction is thus done gradually.

- When a new device is added to an existing topology, the set of metrics associated with it are added to the Bayesian Network model. An observation is then performed on these metrics and the Chi-Square test is performed between these metrics and the others metrics to find the new nodes position with respect to the initial Bayesian Network model.

Thanks to this flexibility, the operator does not have to update manually and keep track of the changes in its network to be sure that the diagnosis will be accurate and updated automatically. Of course, a period of synchronisation/adaptation is needed during which the diagnosis process may return a root-cause that does not take into the recent addition of a network element or metric.

**Necessity of a case database**

The root causes identified are represented as “a solution-case” following the case-based reasoning approach, and are stored in a case database. These are reused if similar problems were coming to arise. A case database is thus required for the proper functioning of the solution.

A specific evaluation on the dimensioning and deployment is required by the operator in order to guarantee optimal performance of the solution operation. Issue of synchronization of multiple case databases are also part of the deployment issues to be addressed, but are not specific/different to normal operator process.

**Adaptability/Genericity**

The use of self-diagnosis NEM requires the prior specification of the model (technology, topology) and metrics to be monitored at each device. Once these specifications are identified, the operator will have nothing to more to configure for the functioning of the NEM. In other words, this solution requires only a small adaptation effort to be applicable to different network technologies.

The solution is also generic in the sense that it can be instantiated or applied to different technologies without requiring re-engineering or advanced configuration aspects.

**Computation and storage requirement**

Thanks to the combination of CBR and BN, whatever the size of the physical network, the number of BN nodes will never have negative effect on network performance. Indeed, the Chi-Square algorithm allows a dynamic modelling of nodes added to the network.

In addition, it is possible to organize the case database, so as to save the solutions according to “cache replacement strategies”. Thus, no limit will be imposed on the case database size.

The requirement in terms of computation and storage resource can thus be easily addressed by current, conventional technologies.

**Coordination/Cooperation with other NEMs**

The Self-Diagnosis based on Bayesian Networks and Case Base Reasoning NEM is essentially a knowledge-building NEM which main purpose is to compute and provide the most accurate root cause to the demanding system. Therefore, the functioning of the NEM does not require particular coordination with other NEMs, except from a process-oriented (or orchestration) view defining the systems authorized/capable of demanding/triggering a root-cause analysis. This results essentially in the operator setting up default sequence of actions/triggers among the different NEMs involved in the fault management process (monitoring, fault detection, fault notification, fault diagnosis, fault recovery, etc...)

#### 2.2.3.2 NEM20: Self-Diagnosis based on network and service data - NEM21: Optimization of context acquisition and dissemination

The proposed NEMs can be applied distributed inside the network therefore the hardware capabilities required for their smooth deployments depend on the network domain they will be applied in. For access networks, where the aggregated measurements are not highly bulky, typical low-end server capabilities are sufficient to manage the load of the NEMs. For the core network the requirements are higher, however parallel computing and virtualization also help towards this direction, in case that the operator needs to reduce the cost.

After deploying the NEMs, a communication way with UMF core is needed so as to enable the orchestration of the NEM functionality; thus GOVERNANCE, COORDINATION and KNOWLEDGE interfaces should be also deployed. The trigger for the instantiation of each NEM can be received by the GOVERNANCE interface while the inputs (i.e. network measurements such as delay, jitter, PL) required for the NEM functionality can be gathered from the KNOWLEDGE interface. COORDINATION is used in case that synchronization or interaction between NEMs is required. Technically speaking, the abovementioned interfaces are agreed to be RESTful web services (client/server approach) that communicate using HTTP/TCP/IP protocol stack, however, since NEM20 and NEM21 are simulations, the afore communication is not available yet.

Regarding the criteria for assessing the deployment aspects, the following criteria are achieved from NEM20 and NEM21:

- Accuracy in terms of functionality: Clustering algorithm and fuzzy logic provide accuracy rates for revealing the quality of the derived classification.
- Adaptability in terms of portability: NEM20 and NEM21 are feedback based approaches that adapt the QoS labelling procedure according to the environment changes (inputs).
- Scalability: There are no restrictions on the amount of input data and the mechanisms can be deployed at every network domain (i.e. access, core).
- Interoperability: NEM21 interacts with NEM20 for achieving self-Diagnosis.

#### 2.2.3.3 NEM13.2: Anomaly Detection

The maintainability is driving motivation behind this NEM. The proposed solution aims to hide the low-level aspects of the detection process from the network manager, that is, the parameter tuning. Efficiency is another important goal of the NEM. The evaluated algorithms and prototypes for this NEM are all flow-based, i.e., they operate on aggregated traffic information instead of using traditional packet-inspection based approach. In this way, scalability and low resource usage are achieved, provided that flow-enabled monitoring equipment is deployed in the target network. Security aspects, such as the resistance of the NEM against Denial-Of-Service attacks, will also depend on such equipment. Guaranteeing and evaluating the interoperability with other NEMs deployed in the network is in progress. As described in 2.2.2.4, it is planned that other NEMs change the detection policy based on their own observations and actions.

#### 2.2.3.4 NEM14: Proactive Diagnosis/ Prediction of Congestion

This NEM has not yet been incorporated in any demonstration thus its development assessment is not available yet. However, the way it should be deployed has already been identified. The NEM is a knowledge-based NEM which means that its main interactions (according to the current specifications of UMF) will be with KNOWLEDGE UMF core block. The NEM will interact with KNOWLEDGE UMF core block for providing its knowledge and predictions so as the latter to be available to NEMs that will be capable of exploiting such information. On the other hand, KNOWLEDGE UMF core block will collect, store or even process this information and disseminate it to the NEMs that will request it, e.g. NEMs that address P1.2.1. As a result, the NEM should be capable of sending and receiving messages to KNOWLEDGE UMF core block. Moreover, the NEM will also need to interact with GOVERNANCE UMF core blocks for its registration and management. Thus, interface between the NEM and GOVERNANCE UMF core block should also be available.

#### 2.2.3.5 NEM33: Proactive diagnosis based on pattern recognition

This NEM has not yet been incorporated in any demonstration thus its development assessment is not available yet. However the proposed NEM can be equally applied in a distributed or centralized process.

The hardware capabilities required for its deployment depend on the network domain they will be applied. For access networks, where the aggregated measurements are not highly bulky, typical low-end server centralized process capabilities are sufficient to manage the load of the NEMs. For the core network the requirements are higher, however parallel computing and virtualization also help towards this direction, in case that the operator needs to reduce the cost. The main factor of performance of the NEM is the complexity of the pattern that directly impact the core ANN design and its monitoring connectivity to the network.

After deploying the NEMs, a communication way with UMF core is needed so as to enable the orchestration of the NEM functionality; thus GOVERNANCE, COORDINATION and KNOWLEDGE interfaces should be deployed. The trigger for the instantiation of the NEM is received by the GOVERNANCE interface while the internal architecture and the monitoring connectivity descriptions required for the NEM functionality can be gathered from the KNOWLEDGE interface. COORDINATION is used in case that synchronization or interaction with other NEMs (typically, self-diagnostic and self-healing NEMs). Technically speaking, the abovementioned interfaces are client/server approach services that communicate using HTTP/TCP/IP protocol stack. Implementation of these processes will be incorporated once technical choice for UMF implementation will be fully available.

Regarding the criteria for assessing the deployment aspects, the following criteria are achieved from NEM33:

- Accuracy in terms of functionality: the core ANN provide sufficient accuracy rates for predict the next step of monitored patterns, the non-stop monitoring and detection process continuously increase these rates.
- Adaptability in terms of portability: NEM33 is a feedback based approach that adapts the detection thresholds according to the environment changes.
- Scalability: There are no restrictions on the amount of input data and the mechanisms can be deployed at every network domain (i.e. access, core).

Interoperability: NEM33 is designed to continuously interact with other NEMs for achieving self-Diagnosis and Self Healing.

## 2.3 Use Case 2

Today networks are becoming more and more ubiquitous and dynamic. In the near future they will be able to hook together a sheer number of heterogeneous (processing, communication and storage) real/virtual resources. One of the major challenges of will be managing such large amount of resources in a cost effective and timely way in order to meet technical and business requirements, which, in turn, may change dynamically. Introducing self-organization capabilities in network management is seen as a potential answer to face these challenges whilst ensuring network adaptability to changing conditions. In essence, these self-organisation capabilities can be practically achieved through the exploitations of “constrained optimizations” (CO) solvers (i.e. through protocols, control loops, algorithms…spread across network layers). These solvers have to be properly orchestrated to avoid instabilities due to unwanted couplings or interactions. Actually, it should be noted that the potential occurrence of network instability may have primary effects, such as jeopardizing dramatically the performance and compromising an optimized use of resources. Ensuring stability of a strategic asset like a network of ICT resources is of paramount importance for society. Main goals of UC2 are to develop and demonstrate methodologies and practical approaches to detect and control the potential occurrence of instabilities in diverse network contexts. Specifically, the actor is the Network Operator. Network Operator should have the possibility to monitor and control network stability. This implies, in turn, the possibility of enforcing high level policies for self-stabilization or even the possibility of de-activating autonomic methods or control-loops (whose unwanted coupling may have caused instability).

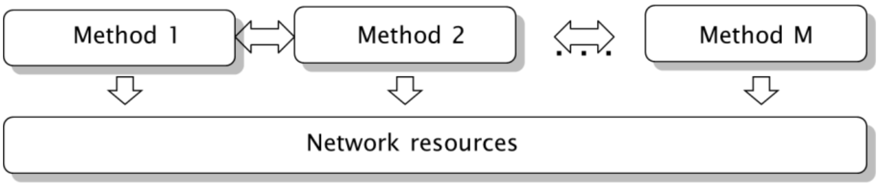


**Figure 6: Unintended coupling or interactions of multiple methods *may create instabilities***

### 2.3.1 Use Case 2 key solution

UC2 key solutions are based on two main complementary approaches, which are:

- Achieving stability through Constrained Optimization.
- Achieving stability by detection of Vulnerable Configurations.
- Self-Organizing Maps in Support of TCP Vegas.

#### 2.3.1.1 Achieving Stability through Constrained Optimization

Self-organization in future networks will be achieved through exploitations of “constrained optimizations” (i.e. through protocols, control loops, methods, algorithms…spread across network layers). It should be noted that even today Internet protocols can be reverse engineered as “constrained optimization” solvers.

Given the growing complexity of networks, in the future said “constrained optimizations” should be made more and more automatically (self-*). In order to maximize the benefits from “Constraints Self-Optimizations”, (exploited through control-loops…) it is necessary to orchestrate their actuations in order to avoid potential unwanted coupling or even competition causing instabilities, that can jeopardize network performance.

In [8] and [9], Kelly et al. presented the innovative approach of formulating a network constrained optimization problem in terms of maximizing an utility function (e.g. for TPC the variables are the source rates constrained by link capacities; the objective function is an arctang(.) capturing the protocol design goals). Since then many research activities have been carried out on Network Utility Maximization (NUM). Even also cross-layer interactions can be characterized by viewing the process of “layering as decomposition of a given NUM problem into many network sub-problems (solvable with control loops). These sub-problems can be “combined together” by certain utility functions. An example of a framework of “layering as optimization decomposition” is well described in [10].

UC2 is dealing with network stability by progressing above approach. Scope is defining a practical framework for setting-up, configuring and tearing down sets of methods (e.g. control loops, algorithms, etc) in a network in order to achieve optimization and stability at the same time.

#### 2.3.1.2 Detection of Vulnerable Configurations

From a security perspective, the stability of a network also depends on its capability to prevent vulnerable configurations. Operations performed during self-management tasks may make the network converging to vulnerable states and expose it to multiple security threats. Standardization efforts have contributed to the normalization of languages for enumerating and describing these vulnerabilities, such as the Open Vulnerability and Assessment Language (OVAL) [11], have been done for enumerating and describing these vulnerabilities. In that context, an approach has been introduced in [12] [13] for integrating these descriptions in the autonomic management plane.

This solution corresponds to NEM 13.1 dedicated to vulnerability management, whose the transversal goal is to dynamically detect configuration vulnerabilities while other maintenance operations are done. The objective is to enable autonomic networks to exploit the knowledge represented by vulnerability descriptions, in order to prevent vulnerable states. More precisely, the approach consists in translating OVAL descriptions into policy rules so that these policies can be then interpreted and executed by the Cfengine autonomic configuration system [14] to support vulnerability detection. We remind that an OVAL description (or OVAL definition) specifies a criterion that logically combines a set of OVAL tests. Each OVAL test in turn defines the process by which a specific configuration condition or property is assessed on the target device. Each OVAL test analyzes an OVAL object looking for a specific state, thus an OVAL test will be true only if the referred OVAL object matches the specified OVAL state. For instance, an OVAL test can check the version number of a given library. The overall result for the criterion specified in the OVAL definition is built using the results of each referenced OVAL test. We have formalized these vulnerability descriptions using first-order logic and developed a translation algorithm for inferring Cfengine policies. This algorithm has been implemented within a prototype, called Ovalyzer, and has been experimented in the context of IOS-related OVAL vulnerability descriptions.

#### 2.3.1.3 Self-Organizing Maps in Support of TCP Vegas

TCP Vegas is a congestion control mechanism that is capable of acting proactively for minimizing the dropped packets. However, it is not yet widely applied as it has a very important drawback. In case of dynamic network conditions it may misinterpret a network reconfiguration, e.g. reroutes in the forward or the backward path of the TCP flow, for potential congestion and react by decreasing the congestion window so as to avoid the congestion. On the other hand this instability of TCP Vegas can cause further instabilities in the network as the unneeded decrease of a congestion window also implies lower rate of the TCP flow and lower utilization of the link.

This mechanism targets at enhancing TCP Vegas' performance even in dynamic network conditions by supporting it with the knowledge on if the reason that caused an increase of the Round Trip Time (RTT) was a reconfiguration or a potential congestion. This will remove TCP Vegas drawback and thus the instabilities that this causes. The core idea of the mechanism is to use SOM learning technique for building the necessary knowledge. This knowledge can then be used as a feedback to the congestion control mechanism for guiding its decisions of decreasing or not the congestion window.

## 2.3.2 UC2 Key solution evaluation

### 2.3.2.1 Stability through Constrained Optimization

The Stability through Constrained Optimization (corresponding to NEM 47) has been evaluated through simulations. NEM-47 harmonizes the configurations of methods (intended as network CO solvers) in order to avoid conflicts and instabilities jeopardizing the network performance. Figure 7 is showing the adopted solution.

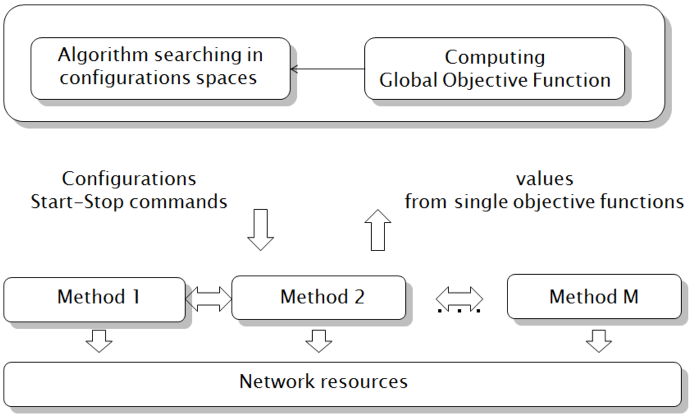

Algorithm searching in configurations spaces
Computing Global Objective Function
Configurations Start–Stop commands
values from single objective functions
Method 1
Method 2
Method M
Network resources

**Figure 7: NEM 47, Stability through Constrained Optimization**

Each method tries maximising its associated objective function. At the same time, NEM-47 algorithm searches, in the space of methods configurations (using, for example, a beam-search algorithm), those configurations allowing to maximise the global objective function (which is a combination of the single utility functions). This is done at regular intervals, upon reaching a trigger or in reaction to changes in the global objective function.

### 2.3.2.2 Detection of Vulnerable Configurations

The detection of vulnerable configurations (corresponding to NEM 13.1) has been evaluated through the Ovalyzer proof-of-concept prototype [15]. This prototype written in Java 1.6 over Fedora Core is an OVAL to Cfengine translator implementing our algorithm for converting the content of one or several OVAL descriptions into Cfengine policy rules. Its main components and their high-level interactions are described in the functional architecture depicted in Figure 8.

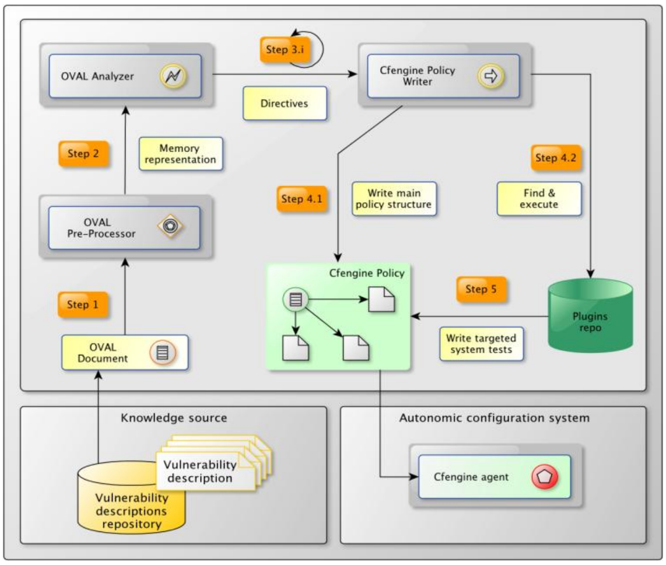


**Figure 8: Functional Architecture of Ovalyzer**

Ovalyzer first takes as input a vulnerability description coming from an official OVAL repository (step 1). An OVAL pre-processor is in charge of parsing the content of this OVAL description, adjusting configuration aspects and feeding the OVAL analyzer module with a memory representation of the specified input (step 2). The OVAL analyzer module is then in charge of orchestrating the translation flow and providing the required directives for generating Cfengine code (step 3.i). It may perform several calls to the Cfengine policy writer depending on the content of the OVAL description. The Cfengine policy writer is responsible for generating the main Cfengine policy entries (step 4.1) and delegating specific platform rules to plugins specifically designed for generating this type of Cfengine code (step 4.2). Plugins produce the required Cfengine code is included inside the generated Cfengine policy files (step 5). The translator core is therefore responsible for managing every high-level aspect of the OVAL descriptions it processes, while available plugins support the required functionality for generating the appropriate Cfengine code. The generated policies can then be consumed by a Cfengine running instance for detecting configuration vulnerabilities.

Since such translations are made in an automatic manner, several tests for evaluating Ovalyzer's performance have been done. We have particularly focused on the time required for generating Cfengine policy files over different sets of IOS vulnerability definitions. The experiments have consisted in executing Ovalyzer with a set of only one definition and measure the generation time, then with a set of two definitions and measure the generation time, and so on, until 134 definitions. Intuitively, one might expect a monotonically growth with the number of definitions to translate, however, the obtained results are quite far from what expected. Within some executions for translating more than 100 definitions, the processing time is near from those executions translating less than 20 definitions. On the other hand, executions with a high translation time can be observed on a regular basis during the experiments that may be due to the scheduling strategies of the operating system, not only with memory processes but also with I/O resources. We believe that such behaviour is interesting for two reasons. First, involved equipment within autonomic networks may present similar scheduling issues; second, it gives a realistic overview of the expected behaviour so autonomic strategies can take such conduct into account. In our experiments, we have measured an average and median time for the executions performed, of respectively 9.5 and 6.1 seconds, which are relatively low values in comparison to the time scale of vulnerability publication.

### 2.3.2.3 Self-Organizing Maps in Support of TCP Vegas

In order to evaluate this key solution, simulations using Network Simulator 2.35 were performed. The network topology simulated is depicted in Figure 9. In this topology, nodes 3, 4 and 5 are routers and nodes 1, 2, 6 and 7 are end-hosts. In particular, node 1 is an FTP server, node 6 is an FTP client connected to the server and continuously downloading from it, while node 2 runs a UDP source application sending data to node 7 at specific time period.

Although the network topology is simplified, it is adequate enough in reproducing and investigating the problem, since the number of hops in a path is not directly affecting Vegas' underestimations of RTT. Our key factor here is the total RTT measured by the end-to-end hosts, and thus, there is no loss of generality even in this simplified topology.

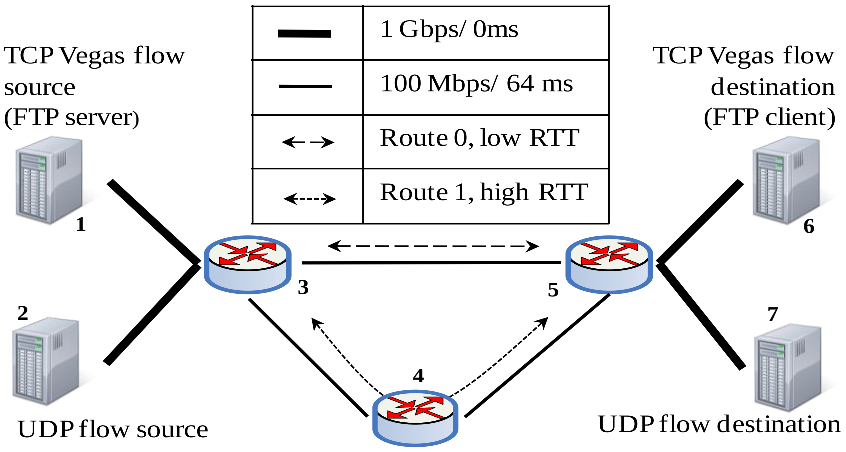


Figure 9: Network configuration for the simulations

Six test cases were examined. However, due to limited space they were merged and presented in two simulation scenarios. The first simulation scenario targets at showcasing three basic test cases in which TCP Vegas misinterprets the monitored increase of the RTT for congestion. This results in TCP Vegas decreasing its congestion window instead of maintaining or increasing its size. In particular, the default route of the scenario for both the forward and backward path for the TCP-FTP flow between the nodes 1 and 6 is route 0. The TCP flow starts simultaneously (t0=0 sec) with the simulation. Enough time is given to the TCP sender so as to increase its congestion window to its maximum and stabilize. At time t1=200 sec the forward path of the TCP-FTP flow is rerouted through node 4 (route 1), which has higher minimum RTT than route 0 (first test case). However, Vegas has no indication of this change and since the minimum RTT has higher value than the previous one, Vegas fails to identify this fact and update it. Thus, the only information that Vegas has is that the RTTs have increased, which for Vegas designates a congested link. So the congestion window is decreased, instead of the opposite, so as to avoid the congestion. At t2=450 sec the path is reverted back to route 0. At time t3=650 sec, a UDP flow starts between end points 2 and 7. The sending rate (from node 2 to node 7) is initially set to 1 Mbps and is then increased by 0.5 Mbps per second for a time period of 200 seconds. In this case as well, Vegas detects an increase to the RTTs as a signal that there is a traffic increase and decreases the congestion window. Although this decision is correct, this will eventually end up to UDP flow dominating the link between nodes 3 and 5. At time t4=850 sec, link 3-5 becomes congested and the TCP-FTP flow is rerouted through node 4 (route 1) (second test case) while at the same time the UDP flow stops. In this case as well, just like the first test case, Vegas failed to understand that the route has changed and that the minimum RTT needs to be updated ending up in no reaction regarding the congestion window size (grey line of Figure 10). At t5=1100 sec, the TCP-FTP flow gets back to route 0, TCP Vegas realizes the decrease of the RTTs and starts increasing again the congestion window. On the contrary, at t6=1350 sec, when the backward path is rerouted through node 4 (third test case), Vegas does not identify the cause of this change. Once more, it misinterprets its observation for congestion and thus decreases again the congestion window that in this case should have increased. The new reroute of the forward path of the TCP-FTP flow through node 4 at t7=1550 sec, as expected from our previous observation, results in no increase of the congestion window. Contrarily, it has reached its lowest value equal to 0. Finally, the simulation ends at t8=1850 sec. The corresponding fluctuation of the observed congestion window in the above scenario is also depicted in Figure 10.

In the same figure (Figure 10) is also depicted the same scenario supported by knowledge. This knowledge enhances the decisions of Vegas when a large RTT is observed. In particular, SOM encourages or discourages Vegas to reset according to the past experience of the network and its current context. As can be seen in Figure 10 knowledge has intervened in Vegas decision at t1=207 sec, t2=857 sec, t3=1357 sec and t4=1558 sec, i.e. when the increase of the RTTs were not caused by congestion but by other circumstances. The comparison of the two diagrams in Figure 10 reveals that knowledge can indeed eliminate the faulty assumptions of Vegas improving its performance in terms of increasing and maintaining the size of the congestion window instead of decreasing it.

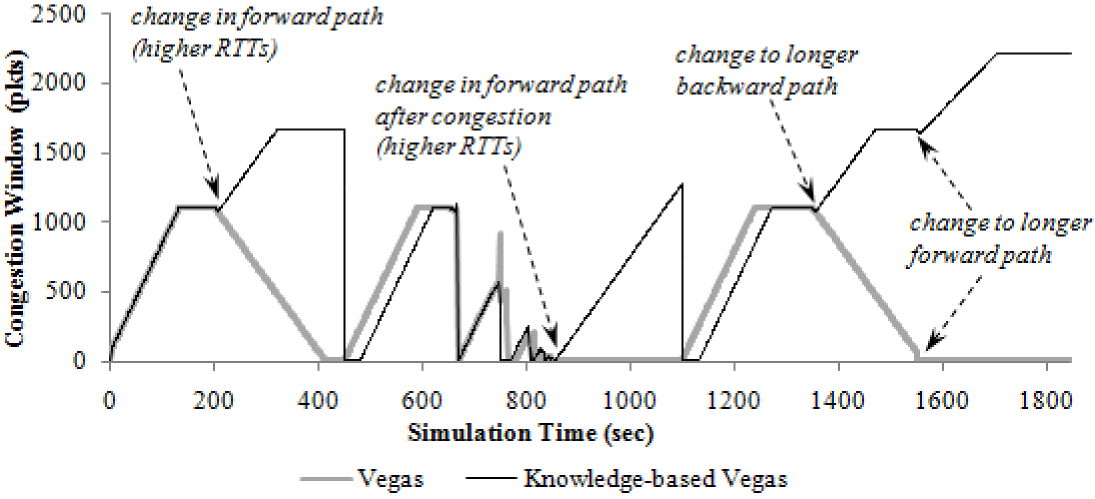


**Figure 10: Fluctuation of the congestion window when using TCP Vegas with and without the support of SOM (Scenario 1)**

Accordingly to congestion window size, Figure 11 depicts the fluctuation of the rate of the TCP flow both with and without support. As the capacity of the path is restrained by the links with the minimum capacity regardless the selected route, the maximum rate that can be achieved is 100 Mbps. Moreover, as can be seen in the diagrams, faulty assumptions of TCP Vegas result in not exploiting in all cases the maximum capacity of the link. On the contrary, when knowledge is used in support of Vegas, maximum capacity of the link is used more often and for longer periods of times. Thus, support of Vegas by knowledge provides a higher and more stable rate for the TCP flow.

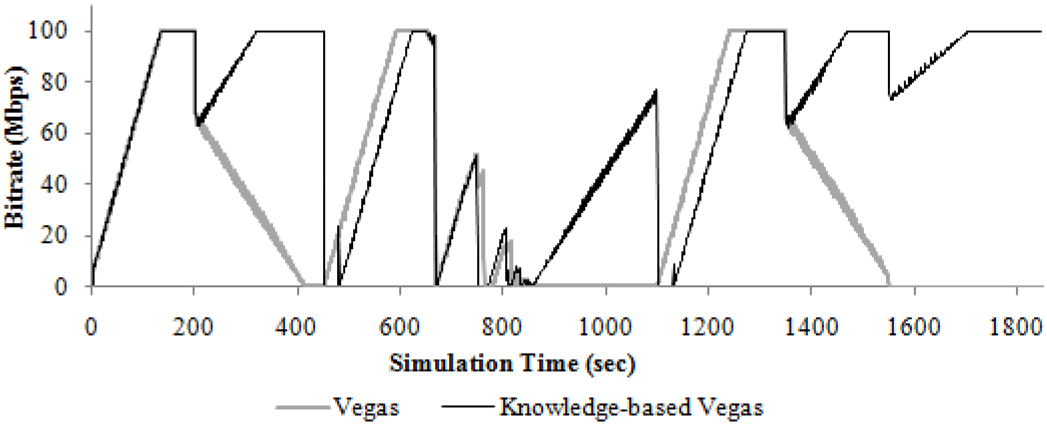


**Figure 11: Fluctuation of the bit-rate of the TCP flow when using TCP Vegas with and without the support of SOM (Scenario 1)**

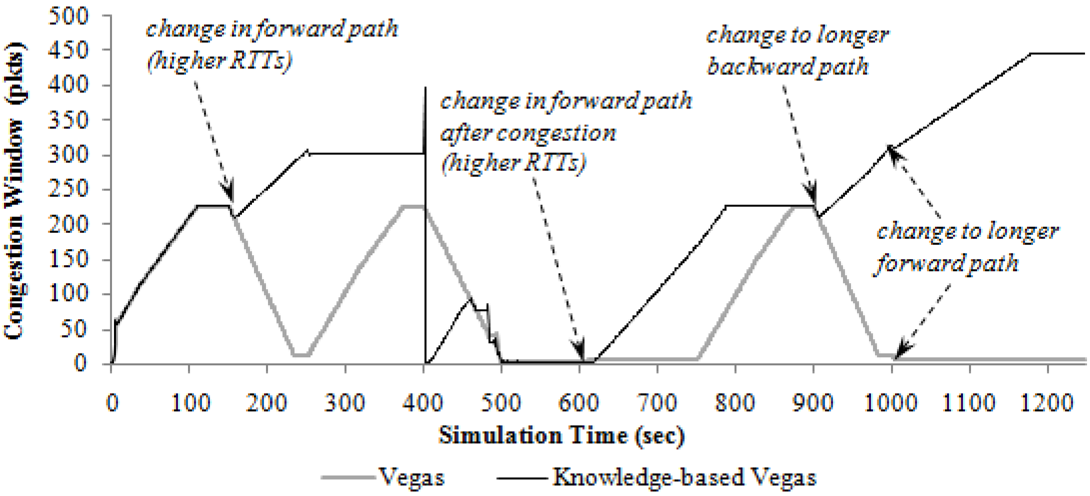


**Figure 12: Fluctuation of the congestion window when using TCP Vegas with and without the support of SOM (Scenario 2)**

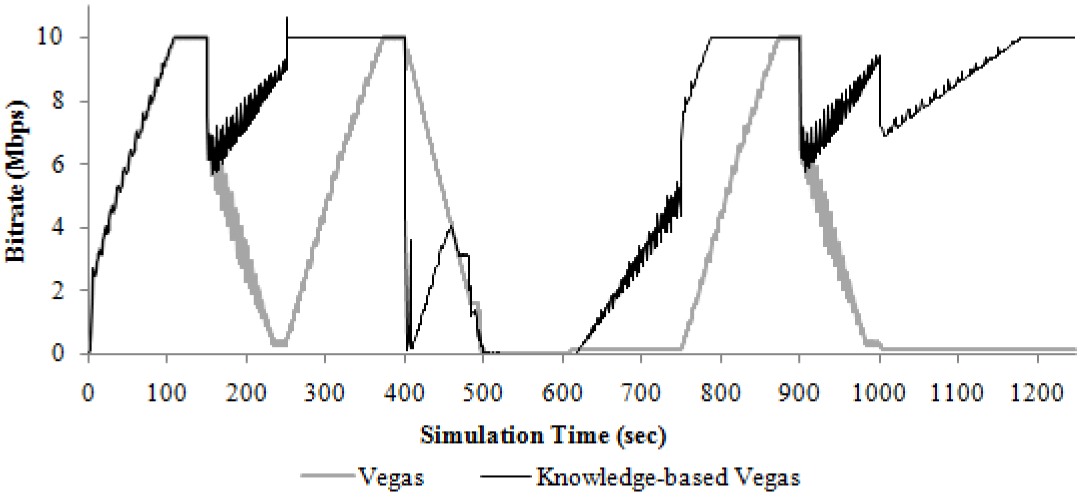


**Figure 13: Fluctuation of the bit-rate of the TCP flow when using TCP Vegas with and without the support of SOM (Scenario 2)**

Moving to the second simulation scenario, the following changes were done with respect to the first one. The acknowledgements are sent following the delayed acknowledgement technique with a delay of 200ms. Moreover, the propagation delay of the links between the nodes 3, 4 and 5 was increased to 128 ms while the capacity of the same links was decreased to 10 Mbps. Accordingly, the UDP flow rate was initially set to 0.2 Mbps while the rate with which it was increasing was 0.1 Mbps per 100 sec. The behaviour of Vegas for this scenario in terms of congestion window size and achieved rate of the TCP flow are depicted in Figure 12 and Figure 13, respectively. Similarly to the conclusions in which we were led by the 1st scenario, this scenario showcased that the performance of TCP Vegas is significantly improved both in terms of congestion window size and achieved rate of the TCP flow when the mechanism is supported by knowledge.

### 2.3.3 UC2 Key solution deployment assessment

#### 2.3.3.1 Stability through Constrained Optimization

NEM-47 has been tested in the case of a highly dynamic network where it is possible moving functionality and virtual resources across a physical infrastructure of combined IT resources (for virtual computing and storage) and network resources (virtual routers). Specifically Virtual Machine (VM) and Virtual Router (VR) can be moved from one physical node to another (the physical node merely serves as the carrier substrate on which the actual virtual node operate). Dynamic provisioning of virtual resources (VMs and VRs) can allow load and traffic engineering in order to improve performance (e.g. limiting hotspots in the IT resources) and to reduce power consumption in the routers network. In other words, the size of the physical network can expand and contract according to load and traffic demand, by idling or powering down node not needed. For example, in case of hotspots in the IT resources, operators can change the allocation or migrate VMs to improve performance (e.g. response time). At the same time, as the network traffic volume decreases, operators can migrate VR to a smaller set of physical routers and shutdown or hibernate unneeded physical routers to save power. When the traffic starts to increase, physical routers can be brought up again and virtual routers can be migrated back accordingly).

In summary, in this use case we can see the interaction of two main coordinated control loops: the former is in charge of allocating VM across multiple networks for performance optimization; the latter is in charge of migrating VR a smaller set of physical routers for saving power (by shutting down or hibernating unneeded physical routers). It should be noted that although both control loops would be stable if operating alone, without a proper coordination, the combination of the two control loops may risk a positive feedback loop.

Let's see how a critical situation can occur. Suppose during one minute a hot spot in a node 1 happens. VM control loop notices this and shifts some VM away from node 1 to a node 2 in the next minute, while at the same time VR control loop notices traffic surge in node 1 and moves more VR to that node and reduce the number of VR in node 2 (saving energy in node 2).  While either of these actions alone would lead toward convergence, the two in combination cause overcompensation. The simulation results of this critical situation are shown in Figure 16.

The utility function of the control-loop in charge of migrating VR a smaller set of physical routers can be based on saving electrical power. The utility function of the control-loop in charge of allocating or migrating VM across multiple networks is based on optimizing certain performance parameter (or set of parameters); for example, we may consider a function that indicates a decreasing utility as the response time increases (but any other functions could be considered depending on the required metrics). The two utility functions associated to the two control loops should not be maximized independently otherwise instabilities may occur. A global utility function should be addressed which is a combination of the utility functions of the two control loops.

Figure 15 and Figure 16 show an example of simulation results for the dynamics of Users' requests which is shown in Figure 14.

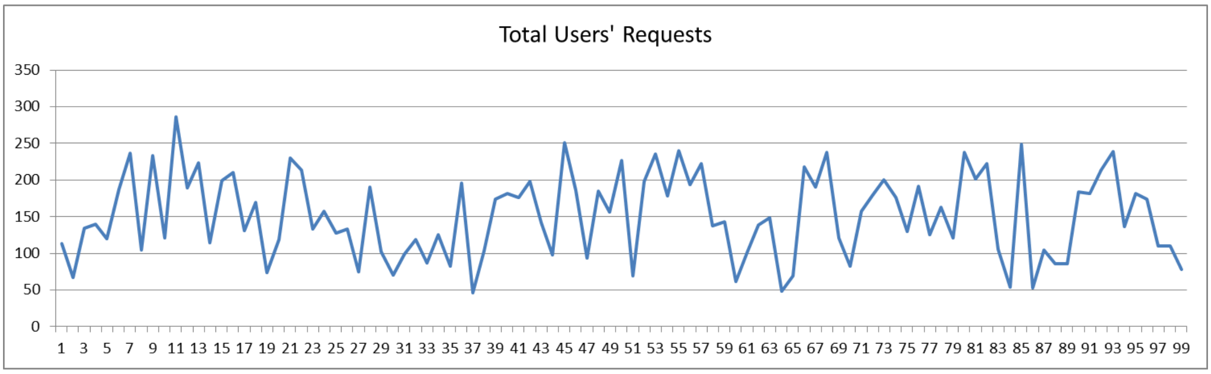


**Figure 14: Total number of Users' requests (as a function of time)**

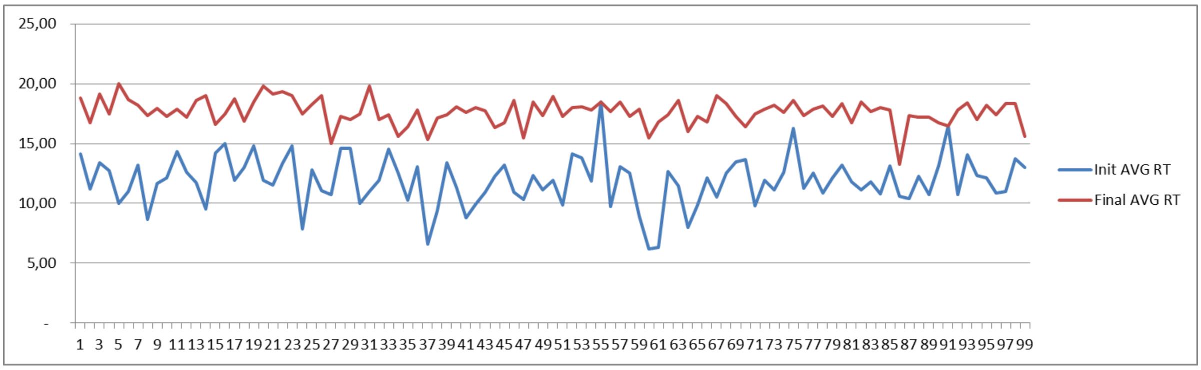


**Figure 15: Average response time of IT servers: without VM migration (blue line); with VM migration (red line)**

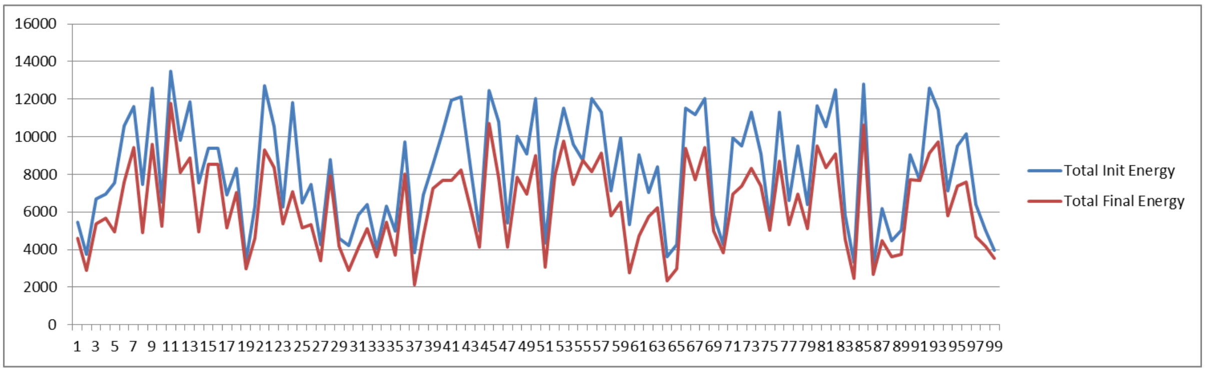


**Figure 16: Total energy consumed by routers: without VR migration (blue line); with VR migration (red line)**

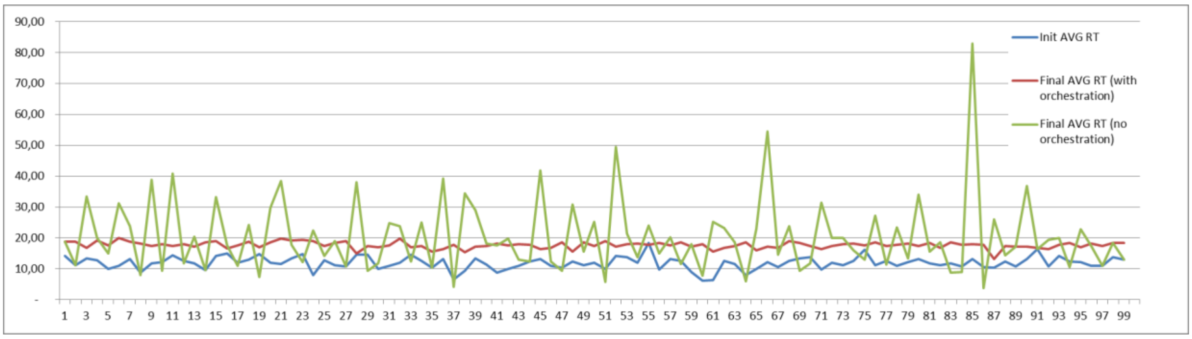


**Figure 16: Comparisons of average response times of IT servers with and without NEM47 (respectively red and green line). Green line indicates the occurrence of critical situations (unstable values of average response times of IT servers).**

Regarding the criteria for assessing the deployment aspects, the following criteria are achieved from NEM47:

- Accuracy in terms of functionality: the adopted CO algorithm ensures the required accuracy rates.
- Adaptability in terms of portability: NEM47 is feedback based approach that orchestrates other NEMs.
- Scalability: it is assured by the overall approach which is a hierarchical decomposition of network problems in sub-problems, for which NEMs are acting as CO solvers (with associated utility functions).
- Interoperability: NEM47 can interacts with any NEM implementing CO solvers through an interface exchanging start-stop commands, configuration parameters and values of related utility functions.

#### 2.3.3.2 Detection of Vulnerable Configurations

The Ovalyzer prototype has been selected and demonstrated during the Demo Contest of the IEEE/IFIP Network Operations and Management Symposium that took place in April 2012 [16]. It has been deployed on an emulated environment, routers being emulated using Dynamips / Dynagen running the operating system IOS version 12.4 [8]. This prototype clearly contributes to the security criterion of the ISO/IEC 9126-1 quality model. In particular, we have quantified the coverage of Ovalyzer with respect to the vulnerabilities (and therefore the potential attacks) that can be handled. The official OVAL repository has 134 vulnerability definitions for the IOS platform. These definitions are based on three types of test, namely, line test, version55 test, and version test. One plugin per each type of test is needed in order to provide the required translation capabilities. In our case study (Cisco IOS), three plugins have been written, namely, CfengineIosLine.jar, CfengineIosVersion.jar and CfengineIosVersion55.jar. During the experiments, we have observed that each plugin does not provide a large coverage by itself. For instance, line test only covers 1.49% of the available IOS definitions. This is because typically vulnerability definitions use more than one test for specifying the required conditions to be met on the target system. When combined, plugins shall cover a wider range of OVAL definition. Different platforms may require a larger family of components to analyse, thus requiring more type of tests and hence, more plugins. For the IOS platform, only these three plugins were required for allowing a maximal coverage, i.e. translating the 100% of available definitions in the OVAL repository. Moreover, the plugin-based architecture of Ovalyzer supports functional extensibility, while its data model supports declarative extensibility using the JAXB library for managing XML documents. Such feature provides to Ovalyzer the ability to dynamically evolve with new OVAL versions and to cover new vulnerability descriptions.

#### 2.3.3.3 Self-Organizing Maps in Support of TCP Vegas

Although this NEM has not yet been incorporated in any demonstration, the way it should be deployed has already been identified. The NEM is a knowledge-based NEM which means that its main interactions (according to the current specifications of UMF) will be with KNOWLEDGE UMF core block. The NEM will interact with KNOWLEDGE UMF core block for providing its knowledge to TCP Vegas in order to enhance and stabilize its performance. As a result, the NEM must be capable of sending and receiving messages to KNOWLEDGE UMF core block. Moreover, the NEM will also need to interact with GOVERNANCE UMF core blocks for its registration and management. Thus, interface between the NEM and GOVERNANCE UMF core block must also be supported.

Moreover, the results from the evaluation of the solution also provide some insights with respect to the gain of using this knowledge-based approach of TCP Vegas instead of the original version of TCP Vegas. For example, in Figure 15 the green areas show the gain that is obtained in terms of utilization of the network for scenario 1 (which was explained in section 2.3.2.3). Similar gains are expected when the mechanism is deployed.

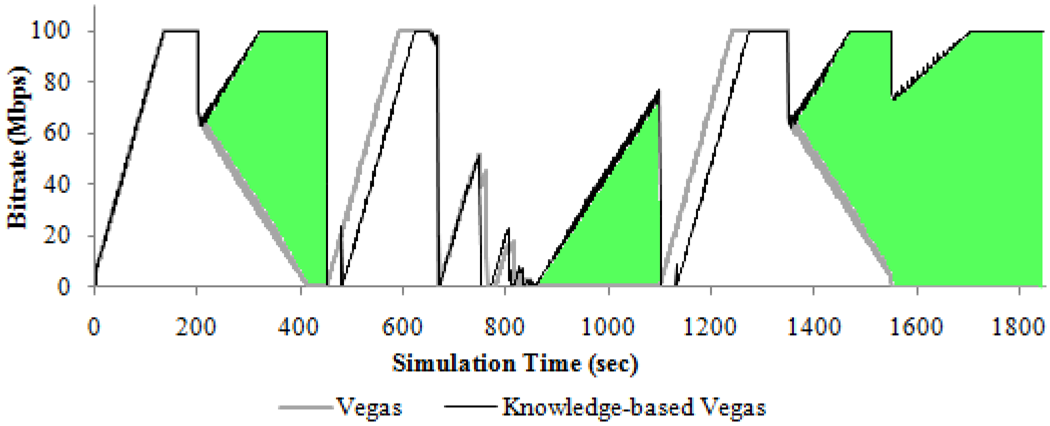


**Figure 18: Gain in terms of utilization when knowledge-based TCP Vegas approach is followed**

## 2.4 Use Case 3

### 2.4.1 Use Case 3 key solution

The use of time sensitive and bandwidth intensive applications e.g., Mobile video traffic (streaming and broadcast), is becoming an increasingly pervasive application and the QoS/QoE demands for the provision of such application ubiquitously by mobile users is putting a lot of pressure on mobile network operators (MNO) and their respective infrastructures (especially the core and backhaul). To effectively meet the customer demands requires increasing network complexity resulting in increased CAPEX/OPEX.

At present, most of the network/content/application service hosting and management is being concentrated at the core. As a result all the user traffic has to go through the core over the backbone/backhaul and hence a lot of resources (bandwidth and processing wise) are consumed. This centralized organization of networks imposes serious operational and performance demerits, especially in case of mobile users using bandwidth intensive real time applications (such as streaming video content etc.). Due to mobility, such a centralized system poses several issues due to the constant re-routing of user data to its new point of attachment and this would imply dynamic management of bi-directional tunnels maintained between the UE and the Core (more specifically the PDN-GW). Such a scenario thus imposes extra burden on the backbone and the access links and can significantly impact QoS delivery to mobile users. Also such a centralized architecture makes it difficult for new service introduction and incurs high management costs. Most importantly, the concentration of service/content management introduces a single point of failure, and can potentially increase end-to-end delay for mobile users using real-time applications.

In view of this, UC3 focuses on the *dynamic virtualization and migration of data/content and network entities* (gateways and servers) nearer to users. The motivation is to provide/resolve the most frequently used resources/content/functions nearer to the mobile user in order to release the resource (e.g., bandwidth, processing, tunnelling, routing) consumption from the core and distribute them autonomously, intelligently and dynamically (on a use-case basis) towards the edge of the network (i.e., access network). This approach is expected to enhance the overall QoS/QoE for the users with efficient resource utilization. It also focuses on context-aware methods for delivering progressive video streaming services to multiple mobile users with user expected QoE over resource constrained wireless links.

### 2.4.2 UC3 Key solution evaluation

In order to achieve the UC3 objectives several NEMs have been designed, implemented and are being implemented and tested in experimental test-beds/simulation set-ups. Table 7lists the UC3-specific NEMs.

| NEM | NEM Details | Problem Solved |
|---|---|---|
| 7 | ContextDiscoveryfromraw data | To discoverknowledgefromrawcontext data and derivereasoningthatisuseful for the process of autonomicmgmt. |
| 42 | Dynamuc Resource Management and Stability | Address the issue of configuration instabilitiesintroduced by the interaction of multiple control loopswithpossiblyconflicting objectives |
| 43 | Context Management | Optimization of monitoring information flow based on changing conditions (e.g. topological changes, consumer location) |
| 44 | Management of Virtualized Infrastructure | Managingvirtualized network and service elements |
| 21 | Optimization of context acquisition and dissemination | Extractcontextfrom massive network data by applying data pre-processing and data-mining techniques |
| 22 | Wireless accessloadbalancing | Multi-interface devices able to connectnumerousAPswithvarying network load and service availabilities. |
| 23 | Service migration | Localization of services based on the demand and revenue |

| | | |
|---|---|---|
| 10 | Load-aware EPC instantiation | Instead of static EPC instances thatrun on dedicated hardware, soft EPC willbedynamicallydeployed on a network of general-purposenodessuch as to optimizeoverallload in the network. |
| 45 | Codec Selection and FairScheduling (CSFS) | For achievingfairness in delivering "OTT video" content to mobile users and ensuring optimum QoE delivery. |
| 48 | Migration of contents | To define the most appropriate node to which data will be stored in order to be offered to the user without the need to access the core network. |

**Table 7: UC3 Specific NEMs**

The above listed NEMs are being implemented in experimental testbeds and simulation environments. The summary of the different evaluation frameworks are described below:

Wireless access load balancing and service migration NEMs are an end-to-end load balancing solution based on cascade fuzzy classification. Load balancing and data offloading to complementary networks as a way to deal with temporal congestion in a cellular network have become possible due to an increasing number of devices supporting more than one radio interface. In order to support end-to-end as well as QoE metrics, load-balancing mechanisms should utilize information from user end devices, networks and services. The cascade approach allows splitting the classification problem into smaller pieces where the (fuzzy) rules are easier to define. Moreover, this supports a distributed framework where different decision mechanisms can select suitable combination of pre-classified context information and combine those according to their needs. Compared to our earlier work based on self-organizing maps (SOM), reported in a similar setting in D3.1, the cascade fuzzy solution seems more suitable for load balancing in highly dynamic wireless accesses.

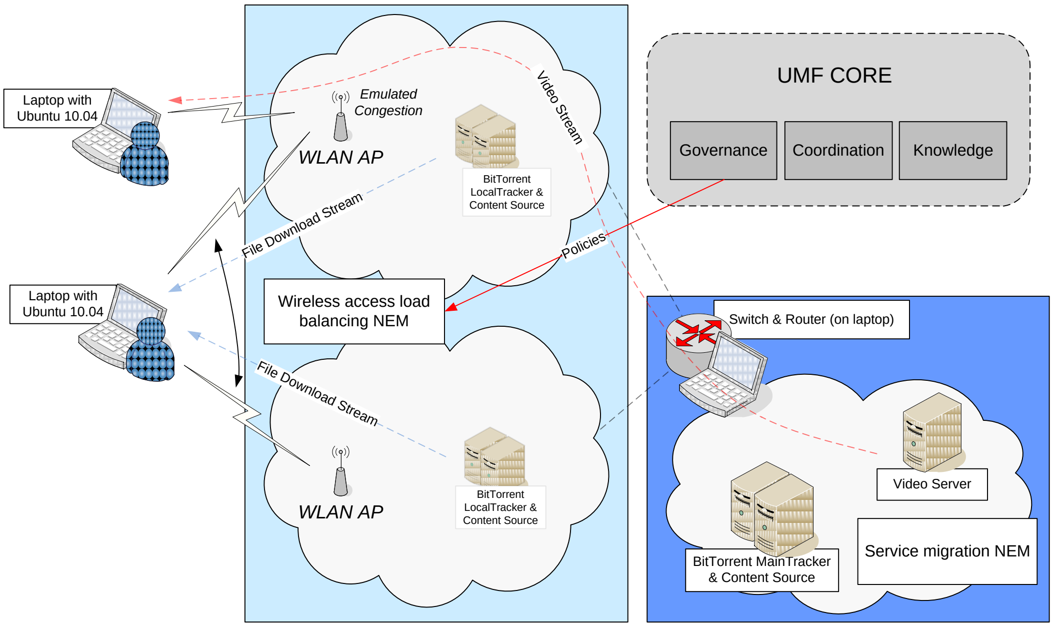


**Figure 19 : VTT'swireless testbed and NEMs.**

Figure 19 represents VTT's wireless testbed located in Converging Network Laboratory (CNL, http://www.cnl.fi), and the implemented NEMs (Wireless load balancing and Service migration). Evaluation framework is based on real devices such as laptops (running Ubuntu operating systems), mobile phone(s), access points (IEEE 802.11a/g) and a switch. Implemented NEMs, running on the testbed, are monitoring information from the network, services as well as end-user devices, and uses this information as an input for fuzzy-based classification system, which ranks available access points for certain user equipment and for particular service. At the same time, the system maintain operator control through UMF's Governance block by providing means to specify, which users performs actions based on classification results (e.g. different user classes). With implemented NEMs on this testbed, the most critical traffic peaks in the access points are minimized and the QoE for the most critical customers can be guaranteed. More detailed view of the system, used algorithms and preliminary evaluation results can be found in D3.5 [5].

#### 2.4.2.1 Optimization of context acquisition and dissemination NEM 21

Context acquisition and dissemination NEM is based on data mining tools (MATLAB simulation) for automatically extracting knowledge from network elements (NEs). Sensing the environment of each NE and aggregating the network measurements (i.e. Delay, Jitter, Packet Loss), one can decide about the state (i.e. Load, QoS) of this specific device by applying data pre-processing and data clustering algorithms to these datasets thus producing a respective label. This algorithm can be either applied in a distributed way (i.e. each NE identifies its own state) or a centralised one (i.e. measurements from several NEs of a network domain are aggregated through KNOWLEDGE), depending on the operator's needs. The application of this mechanism should be done periodically; the frequency may be decided by the operator according to its policies so as to trigger a remedy action (e.g. Self-diagnosis, Self-healing). Context acquisition and dissemination NEM moves towards this direction in order to automate such operations (i.e. reduce OPEX) as well as to trigger several machine learning algorithms for autonomic decision making. The latter is part of the inter-NEM communication through the COORDINATION block of UMF which enhances the autonomicity levels of the network in a safe and conflict-free way, respecting also the given policies.

#### 2.4.2.2 Migration of contents NEM 48

Migration of contents NEM is a bio-inspired solution and is based on Ant Colony Optimization (ACO). Specifically, within the network nodes exist that are able to store data. The target of this NEM is to define the node to which data should be stored in order to be accessed by the users. Therefore the operator can save valuable resources, e.g. the core network, due to the fact that users can access data from the aforementioned nodes without the need to utilize the core network. In addition the proposed solution will increase the QoS (in terms of bitrate, delay, jitter, etc.) that users experience because content is migrated closer to the users. The ACO approach that is exploited has received a very positive feedback from the network community due to their proved ability to provide good solutions in a logical amount of time, distributed usage, as well as the ability to be applied in the most up-to-date network problems.

Figure 20 illustrates UPRC's simulation platform that was developed in Java. The implemented NEM exploits the ACO algorithm in order to determine the optimal data storage node to which contents will migrate to. The algorithm takes into account various aspects, e.g. the quality of the links that connect the nodes (in terms of bitrate, delay, etc.) and the capabilities of the nodes (e.g. available storage capacity, etc.).

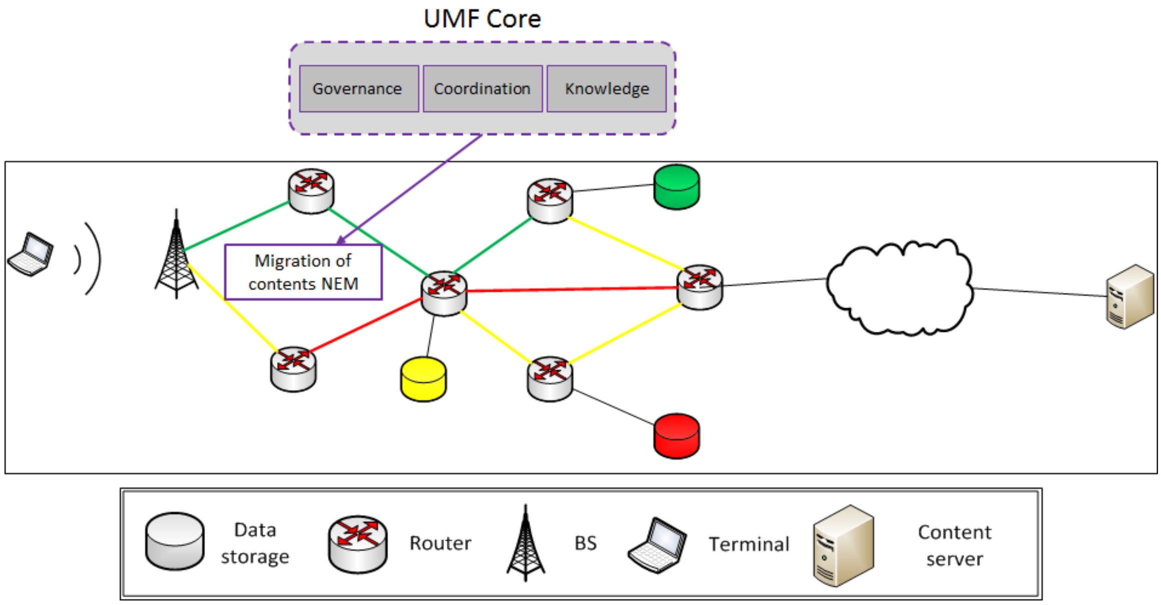


**Figure 20 : UPRC's simulation platform.**

### 2.4.2.3 Codec Selection and FairScheduling (CSFS) NEM 45

Codec Selection and Fair Scheduling (CSFS) NEM (NEM 45) has been proposed in order to select suitable codec profiles for mobile users that want to access YouTube-like services, and then ensure that the video content is being progressively delivered to multiple users, sharing the same channel or associated with the same Base Station (BS)/eNodeB (eNB) the user, in a fair manner. The Codec Selection algorithm will ensure that the codec selected for/by the user is appropriate to the user device capabilities and underlying network conditions. Once selected, the fairness aspect will ensure that each user, regardless of its relative position/distance from the BS/eNB is able to receive video content at rates that will ensure the user to view it with the expected QoE. In other words, fairness will ensure that users with favourable channel conditions do not hog the bandwidth from users with lesser channel conditions. This is typically the case with TCP RTT un-fairness. The fairness algorithm ensures that each user is able to receive data at rates that will ensure that the playout buffer does not get starved, which otherwise would lead to image stalling or jerkiness degrading the QoE.

However, the CSFS NEM is dependent heavily on the timely availability of good quality context information. Figure 21 shows the interaction overview of CSFS NEM with other relevant NEMs via the UMF. The CSFS NEM functionality and services are being developed in an OMNeT++ based simulation environment, and its details along with some initial results are reported in D3.7 [6].

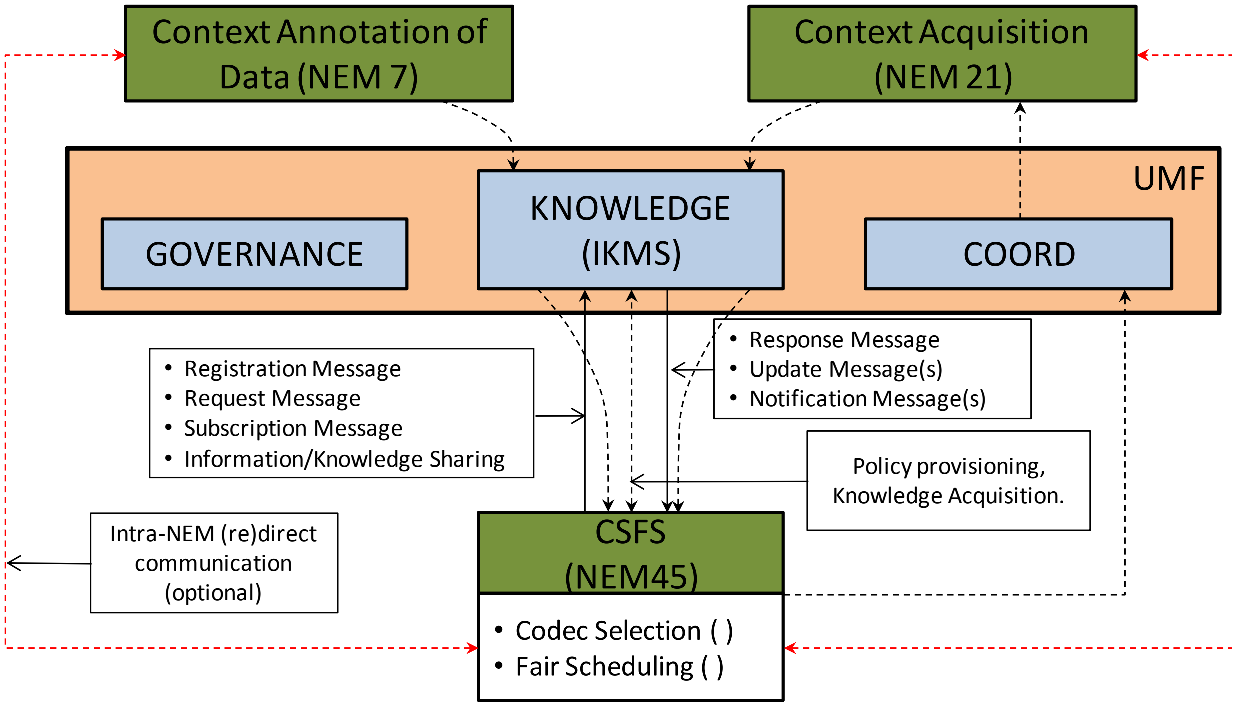

Context Annotation of Data (NEM 7)
Context Acquisition (NEM 21)
UMF
GOVERNANCE
KNOWLEDGE (IKMS)
COORD
• Registration Message
• Request Message
• Subscription Message
• Information/Knowledge Sharing
• Response Message
• Update Message(s)
• Notification Message(s)
Policy provisioning, Knowledge Acquisition.
Intra-NEM (re)direct communication (optional)
CSFS (NEM45)
• Codec Selection ( )
• Fair Scheduling ( )

**Figure 21: Conceptual overview of the interaction of NEM 45 with NEM 21 and NEM 7 via the UMF**

### 2.4.2.4 Load-aware EPC instantiation NEM 10

Currently, 3GPP analyses ways to decentralize the mobile core by introducing statically located Local Gateways near the access network and providing early breakout points for the user's traffic. The Soft-EPC NEM (NEM 10) has been proposed as a further step in decentralizing the mobile core by allowing the dynamic instantiation of the required functions/gateways on General Purpose Nodes (GPN) along the path a given service/ set of services requires and enabling a flat network architecture. Two enabling algorithms have been proposed, the details of which has been summarised in D3.5 [5]. As simulation framework has been developed to analyse the gain, and the initial results presented in D3.5 [5] show that, for a given set of services consuming resources on a network, the two proposed algorithms can outperform a centralized approach in terms of the metrics analyzed.

Furthermore, the concept of GPN allows the instantiation of multiple virtual function instances on the same physical node, which can be beneficial for use cases where multi-tenancy or multi-ownership is required. The reference simulation topology that is being used for analysis is depicted in Figure 22, which also shows the distributed instantiation of Evolved Packet core (EPC) functions across GPNs.

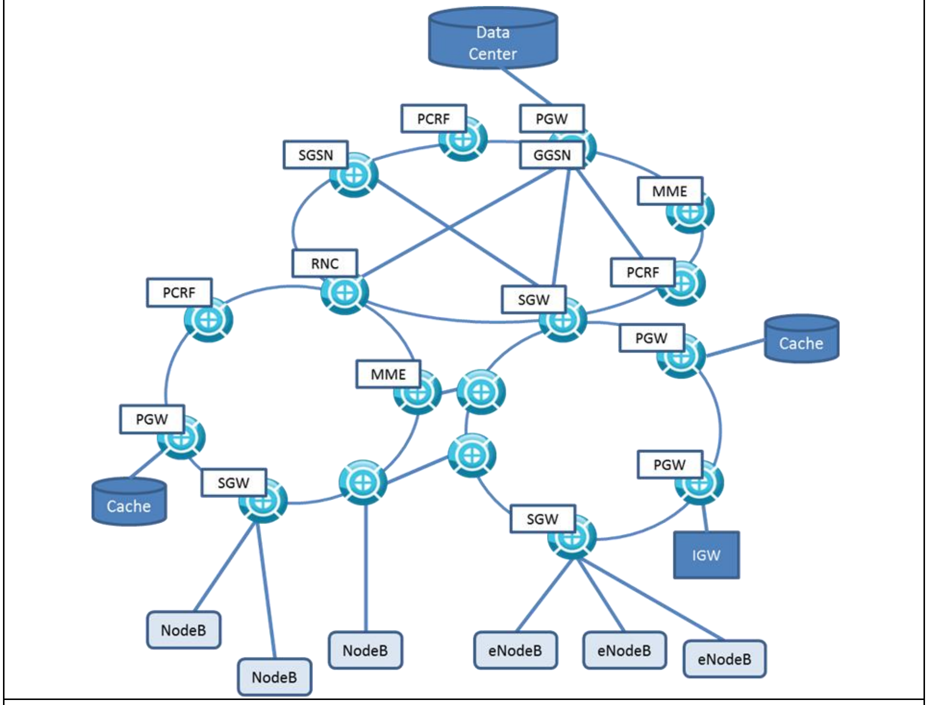


**Figure 22: Reference topology with distributed instantiation of EPC functions across GPNs**

### 2.4.2.5 ContextDiscoveryfromraw data NEM 7

Raw data obtained from different sources of information within the network does not currently properly processed in order to help operators for providing a fine-tuned service to the customers. An example of such a context is the user movement pattern visiting a specific venue. Annotating the data from the user's context (location taken from the GPS) and the information from the venue (size, number of annual visitors, etc.) can lead to reason user demands (e.g. access to particular web sites) to be locally cached in advance to a location closer to the venue. The results of such reasoning contribute in autonomic discovery of the network context to help business drivers for providing context aware services. Autonomicity provided here is supported by the combination of relying on external Linked data and reasoning techniques (e.g. rule based reasoning) where defeasible reasoning is required. UMF plays its role in managing the knowledge of the network (KNOWLEDGE block) fed by this NEM. The autonomicity resulted from maintaining context information helps in reducing OPEX costs where the operators are able to balance and optimise the load over different links, reduce the peak time traffic, and provide higher QoS to the customers. The context awareness provided here supports the operators to be confident over the autonomic network management by relying on context information.

The performance of this NEM is evaluated using the simulator environment relying on external traces and statistics where external sources of data get involved. Simulator environment is NS-2.. Figure 23 illustrates the evaluation process; refer to Deliverable D3.7 [4] for more details.

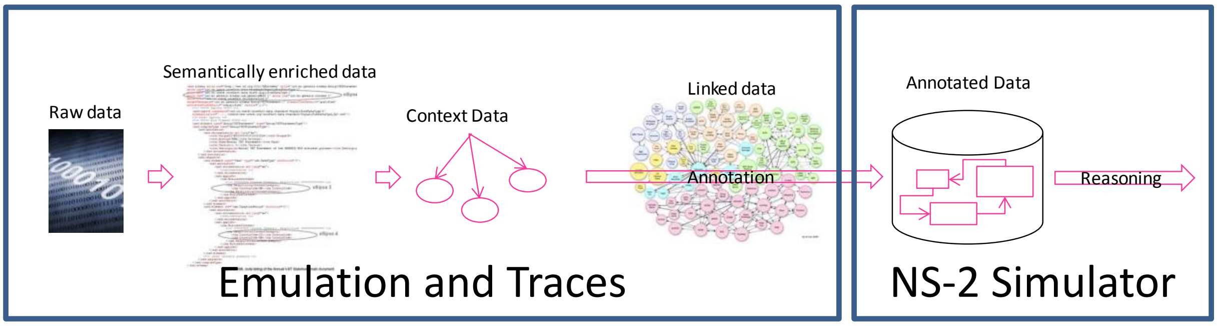


**Figure 23: Evaluation environment for Context Discovery NEM**

### 2.4.3 UC3 Key solution deployment assessment

The deployment requirements for the various NEMs, that constitute the core of the UC3 solution space for solving the problem space of the Dynamic Virtualization and Migration of Contents and Servers Use case, will be assessed in terms of usability, efficiency, extensibility and scalability.

#### 2.4.3.1 Wireless access load balancing (NEM 22) & Service migration (NEM 23)

Wireless access load balancing and service migration NEMs are based on distributed information collection, decision making and information delivery framework that does not require significant amount of computational resources in the network. However, if the information collection and refinement from the network side cannot be done in e.g. access points/base stations, it needs resources (i.e. server), which can be reached also from user equipment side. Information from the network is collected through standard interfaces and protocols, such as SNMP (device level information) and HTTP (service level information), so extra changes to the network side are not required. Additionally, user equipment must have support for providing required statics from the operating system, as well as must have support for selecting used interface, either in network, transport or application layer. GOVERNANCE block of the UMF core should be able to deliver information about which user equipment, for example, is moved to another network in possible congestion situations.

#### 2.4.3.2 Optimization of context acquisition and dissemination (NEM21)

The proposed NEM can be applied distributed inside the network therefore the hardware capabilities required for its smooth deployment depend on the network domain they will be applied. For access networks, where the aggregated measurements are not highly bulky, typical low-end server capabilities are sufficient to manage the load of the NEM. For the core network the requirements are higher, however parallel computing and virtualization also help towards this direction, in case that the operator needs to reduce the cost.

After deploying the NEM, a communication way with UMF core is needed so as to enable the orchestration of the NEM functionality; thus GOVERNANCE, COORDINATION and KNOWLEDGE interfaces should be also deployed. The trigger for the instantiation of each NEM can be received by the GOVERNANCE interface while the inputs (i.e. network measurements such as delay, jitter, PL) required for the NEM functionality can be gathered from the KNOWLEDGE interface. COORDINATION is used in case that synchronization or interaction between NEMs is required Technically speaking, the abovementioned interfaces are agreed to be RESTful web services (client/server approach) that communicate using HTTP/TCP/IP protocol stack, however, since NEM21 is a simulation, the afore communication is not available yet.

Regarding the criteria for assessing the deployment aspects, the following criteria are achieved from NEM21:

- Accuracy in terms of functionality: Clustering algorithm provides accuracy rates for revealing the quality of the context extraction.
- Adaptability in terms of portability: NEM21 is a feedback based approach that adapts the QoS labelling procedure according to the environment changes (inputs).
- Scalability: There are no restrictions on the amount of input data and the mechanism can be deployed at every network domain (i.e. access, core).

- Interoperability: NEM21 interacts with NEM20 for achieving self-Diagnosis.

### 2.4.3.3 Context Discovery (NEM 7)

Context Discovery NEM is distributed over the network and runs locally, around the user. No specific hardware is required for this specific NEM; it relies on any source of information that could be available through the sensor network around the user. By the enhancements made on the user terminals, deployment of the smartphones and different types of the sensors they provide, the sources of local information is being enriched every day. Access to the linked data from external resources is required for this NEM; therefore web service support is required for achieving its goals. The following criteria are addressed within NEM 7:

- Functionality: in terms of Suitability, this NEM is enhancing the network knowledge by providing the local context. This NEM contributes in service provisioning Accuracy, by supporting context awareness.
- Usability: this NEM improves Operability, and Attractiveness, by improving the QoS based on the local context of the user.
- Efficiency: this NEM contributes to efficient Resource Utilisation, by limiting unwanted service provisioning based on context.
- Scalability: the context discovery NEM is local and distributed, therefore by itself scales up easily. However the scalability of use of discovered context by the operators is subject to investigation.

### 2.4.3.4 Load-aware EPC instantiation (NEM 10)

Similar to NEM 21/22, this NEM is also distributed over the network and its actions are based on the distributed info collection with reference to user service/application requirement and based on that information will instantiate the relevant EPC functions/services at locations that will provide optimum service to the users. This complex action is designed to be coordinated via the UMF GOVERNANCE and COORDINATION blocks with other relevant NEMs.

Regarding the criteria for assessing the deployment aspects, the following criteria are fulfilled by NEM 10:

- Scalability: It offers a scalable solution as its outputs are not restricted to the number of users and/or the application/service demands.
- Accuracy in terms of functionality: The algorithm designed is such that it first computes the user application/service requirements, and then finding the best path for service access. The best path is calculated based on the shortest path that may have the resources available, and hence incurs the lowest blocking probability.
- Interoperability: NEM10 is designed to interact with other relevant NEMs (see deliverable D3.5) via the UMF interfaces.

### 2.4.3.5 Codec Selection and Fair Scheduling (NEM 45)

NEM 45 is located inside the content server, also an element under the migration control. Regarding the criteria for assessing the deployment aspects, the following criteria are fulfilled by NEM 45:

- Interoperability: NEM 45 is designed to interact with other relevant NEMs (see deliverable D3.6) via the UMF interfaces, more specifically the KNOWLEDGE block.
- Extensibility: This NEM renders itself naturally extensible as it is specified to reside in every content server instance that is present or instantiated in the network.
- Usability: Being an actor NEM, there is no usability issues expected with this NEM.
- Reliability: Although NEM 45 relies on other context management NEMs (see deliverable D3.6) via the UMF, however it is designed to provide the minimum capability in case of non-availability of required context information.

### 2.4.3.6 Migration of contents (NEM 48)

The Migration of contents NEM is based on Ant Colony Optimization. Therefore, it can be applied in a distributed manner, as well as in a centralized approach. As far as information collection is regarded, the user equipment should be able to provide its location (e.g. through GPS) in order to determine its location and hence use it as an input in order to define the most appropriate node that the data will be moved to. In addition, the KNOWLEDGE interface of the UMF core shall be utilized in order to store the locations of the users during day. Therefore, this information can be acquired later in order to predict the possible location of the user at each time. Furthermore, the GOVERNANCE interface of the UMF core shall be able to exchange information about which user should be able to use this functionality. However, due to the fact that NEM 48 is currently developed in a simulation platform the communication through the aforementioned interfaces via RESTful web services is not available yet.

## 2.5 Use Case 4

### 2.5.1 Use Case 4 key solution

*Context:* Use case 4 deals with the SON technology, with a particular focus on 3GPP networks (namely LTE and LTE-Advanced), but also considers an important Wi-Fi SON use case. The current world-wide deployment of the LTE technology which supports the SON technology makes this use case of particular interest. In the short term, operators will benefit from self-configuration functionalities which can facilitate the deployment phase and reduce the corresponding OPEX. In a second phase, self-optimization and self-healing will receive growing importance for managing resource allocation, mobility and interference in the network. Such self optimizing functionalities have been standardized in Releases 8 to 11 of the LTE and LTE-Advanced standard. To meet the expected traffic growth in the coming years, heterogeneous networks (with micro, picocells, femtocells and relay stations) can provide an efficient mean for densifying the network and adding additional capacity. In this context, large scale deployment of the SON technology will be crucial to allow managing the networks with thousands of nodes. The large scale deployment of SON at the network nodes with different possible functionalities per node will pose several critical problems:

- Govern the network to achieve specific goals of the operator
- Coordinate different SON functionalities to ensure scalability (for large scale deployment) while achieving the expected performance gain
- Resolve conflicts between SON functionalities when coordination fails
- Ensure stability of the network empowered by the SON technology

The problem of SON coordination has not yet been addressed by 3GPP although it is noted that a first use case of interworking of two specific SON functionalities: mobility robustness optimization (MRO) and mobility load balancing (MLB) has been considered in 3GPP Release 11 (2011-2012).

Figure 24 shows the parameter plane of two SON functionalities managing admission control threshold and the amount of allocated resources (bandwidth) for users. One can see that without coordination, certain regions in the parameter plane will lead to network instabilities.

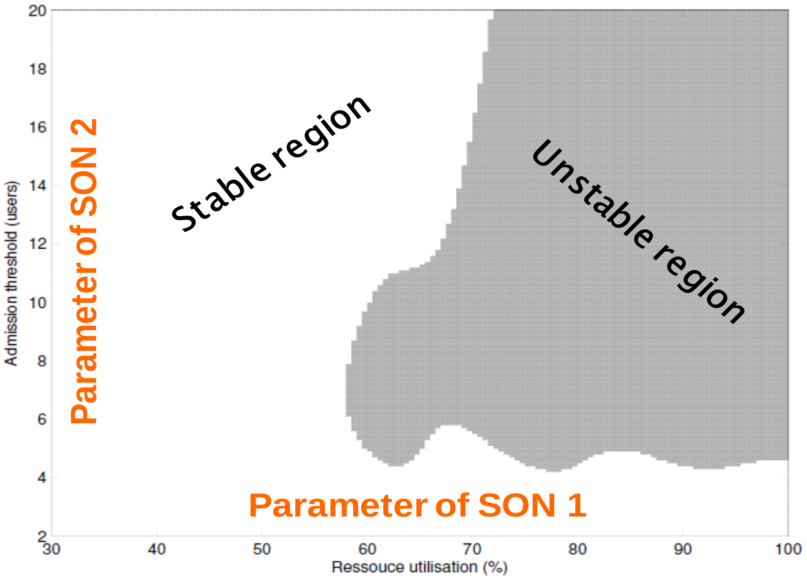


**Figure 24: Parameter plane of two SON functionalities managing admission control threshold and resource utilization**

*Solutions:* UC4 implements UMF core blocks which provide solutions to

- facilitate the introduction of new NEMs with SON functionalities
- provide the tools to govern the network empowered by SON functionalities, via the introduction of policies, utilities, targets and other elements guiding the operation of NEMs
- provide a framework for coordinating SON functionalities and for resolving conflicts. The coordination is among the key enabler for large scale deployment of the SON technology.

Since SON functionalities can differ in time scale, in the parameter they act upon and in the KPI they impact, different coordinating schemes are required for different use cases, rather than a "generic" scheme. Three solutions for SON coordination are developed and experimented in UC4:

(i) Off-line centralized coordination based on optimization

(ii) On-line distributed coordination based on control theory, and a

(iii) A policy based approach based on "policy conflict avoidance"

**Optimization approach solution**: The optimization solution targets centralized off-line implementation of the coordination process of the SON functionalities. Each SON is considered as one optimization process. The optimization is handed in a centralized manner, with less strict timing constraints depending on the SONs under consideration. We consider $I$ SON functionalities acting upon $I$ parameters, $\theta = (\theta_1,\cdots,\theta_I)$. In the optimization approach the objectives (utilities) of the different SON functionalities are integrated into one objective function $F(\theta) = [f_1(\theta), f_2(\theta),\dots,f_I(\theta)]^T$. In this way, the common objective function implicitly handles conflicts of the possibly competing objectives. Constraints related to KPIs or parameter values are typically added as part of the optimization problem. The operator can also include policies in the common objective function in terms of priorities (weights) and desired targets.

**Control approach solution:** Control theory approach considers SON functionalities as control loops, which operate in small time scale and are implemented in a distributed, on-line manner. The entire system including the control loops and the coordination mechanisms are described as an Ordinary Differential Equation (ODE). In the context of control theory the design of a coordination mechanism corresponds to the concepts of *controllability* and *state feedback synthesis*. We consider $I > 1$ control loops, each of which is in charge of controlling one parameter $\theta_i$, and the set of the SON parameters are denotes as $\theta = (\theta_1,\cdots,\theta_I)$. The ODE describing the system is modelled as: $\dot{\theta} = F(\theta)$. To simplify the design, the control model is linearized. Linearization is valid when small intervals of variation for the parameters $\theta_i$ are considered**.** The ODE is represented in the following form: $F(\theta) = A\theta + b$, with *A* being a matrix and $\theta$ and *b* - vectors. The coordination algorithm modifies the matrix *A* in a way to achieve stability. Validity of the solution due to noisy measurement of KPIs (due to propagation phenomena, signal sampling and discretization etc.) is analyzed using stochastic approximation.

**Conflict avoidance solution:** The conflict avoidance approach is based on the extended policy domain approach introduced in the literature. The coordination solution is based on separation in time based on utilities and performance objectives, to derive a triggering sequence for the SON functionalities. The process operates as follows: all SON functionalities exchange their expected utilities per time interval; expected utilities per time interval are ordered in decreasing order. The possible actions for a given parameter are {Decrease, Neutral, Increase} and are denoted as *predicate modalities*. In the next time interval the SON functionality to be triggered is selected following two conditions: predicate modalities for the present and for the previous time interval are the same AND the expected utility of an action chosen based on the previous time interval was the highest in the ordered list. This approach allows avoiding certain type of conflicts; the invocations of mechanisms are self-orchestrated based on relevant frequencies associate to each SON functionality.

**Other solution components:** UC4 proposes novel SON use cases, described as NEMs (listed below), and develop the GOVERNANCE UMF core block for specific use cases studied within UC4.

### 2.5.2 UC4 Key solution evaluation

UC4 objectives are achieved by the design of a set of NEMs for self-configuring, self-optimizing and self-healing the radio access network. Additionally, the coordination mechanisms explained above are listed together with the considered implementation. It is noted that within the prototypes, the UMF governance building blocks are also implemented, although these are discussed in more details in UC7. Both NEMs and coordination mechanisms are listed in the following Table 8 and Table 9.

| NEM Details | Problem Solved |
|---|---|
| Joint femtocell coverage optimization NEM | This NEM is used to dynamically adjust femtocell coverage in order to balance different conflicting objectives for load balancing, while minimizing coverage leakage and coverage gaps. |
| Self-healing mechanism for cell outage management NEM | This NEM addresses the actions needed in order to handle and recover a Base Station/Access Point (BS/AP) failure event in a wireless network. The NEM reconfigure the radio parameters of the network under certain limitations (i.e. SINR values, cell load, etc.). |
| Dynamic Inter-Cell Interference coordination in multihop cellular networks NEM | The NEM handles congestion due to the increase in traffic demand for high bandwidth applications and mitigate interference by dynamically coordinating radio resources and parameter configurations between neighbouring cells in a multihop network |
| Inter-Cell Interference coordination (ICIC) NEM | ICIC NEM finds the appropriate OFDM resource (subcarriers (SCs) or physical resource blocks (PRBs)) allocation in the target cell, to minimize the interference caused at the target cell's users, by taking into account the target cell context (load, radio conditions etc.), the amount of available resources and the context of the neighbouring cells in downlink LTE networks. |
| Coverage and Capacity Optimization (CCO) NEM | CCO NEM finds the appropriate OFDM resource and power allocation in the target cell, to maximize the throughput in the target cell (capacity optimization), while target cell's users experience acceptable channel quality (coverage optimization), by taking into account the target cell context (load, radio conditions etc.), the amount of available resources and the requested channel quality that the target cell's users should experience. |
| Load balancing/resource management of control plane data NEM | The dynamic management of Future Networks requires the modelling, incorporation and integration of suitable mechanisms and algorithms for the network decision making process. An important part of such procedure is the |

| | |
|---|---|
| | end-users' load-balancing which is related to their decision-making requests (control plane data). The key problem addressed is the dynamic computation of the system capacity in terms of computational resources as well as the management of the requests that exceed the system capacity and cannot be handled. |
| Prediction methods for load balancing | The NEM is used for proactive management of the system computational resources, triggering load balancing mechanisms. The possibility of predicting future loads is important to minimize the number of dropped user requests. |
| Intra LTE Load Balancing | The NEM considers performs the task of user association to LTE cell. The association is performed by self-adapting HO parameters |
| Tilt optimization | Coverage and capacity optimization for LTE by changing the vertical antenna tilt base station sectors |
| Model State Reduction (MSR) for coverage optimization | The NEM self-optimized coverage in heterogeneous wireless networks. By using the proposed approaches, the optimization of the small cells' coverage is ensured. The method is fully distributed and is not computationally complex. |

**Table 8: NEMs in UC4**

| | |
|---|---|
| ICIC and CCO optimization based coordination | Off-line coordination based on optimization solution described above. The coordination mechanism is applied to ICIC and CCO coordination in LTE network. The available resources are allocated to users providing either non-inferior solutions/Pareto Optima, i.e. in which an improvement in one objective requires a degradation of another in the case of multi-objective optimization, or satisfying more one objective than the other depending on the operator weights in the case of weighted multi-objective coordination, by taking into account the target cell context, the context of the neighbouring cells, the amount of available resources, the requested channel quality that the target cell's users should experience and the operator weights that correspond to the two objectives |
| Coordinating CCO and traffic balancing control based coordination | Online coordination based on control solution described above. The coordination is applied to CCO and traffic balancing to reduce load of congested macrocell and enhance throughput and QoS at cell edge. These goals are achieved by adapting coverage of relay stations and properly balancing traffic between backhaul links and direct (macro/relay - to mobiles) links. |
| Self-orchestration of SON LTE control loops (SOUP) | The extended policy domain approach described above is used for self-orchestration of SON LTE control loops. Safety and stability of concurrent operation of multiple SON LTE control loops in one LTE cell and across adjacent cells is considered |
| Interaction between Coverage and Capacity Optimization and Energy Saving in Self Organizing Networks | The optimization of the wireless network resources via dynamic AP switch On/Off and terminals load balancing, considering traffic levels and coverage requirements. The proposed algorithm leads to network performance improvement and avoidance of radio and energy resources waste. The interaction is between dynamic AP switch On/Off, which is considered as a Coverage and Capacity Optimization action and other identified metrics i.e. energy is studied. |

**Table 9: Coordination mechanisms in UC4**

UC4 prototypes are described below. The prototypes correspond to UC4 life cycle / phases:

In the Second phase, Prototype 1 has been implemented: a governance framework developed in UC7 is utilized in UC4 to govern both wireless and mobile network segments:

- Self configuration and self healing of APs in a Wi-Fi LAN, and
- SON coordination (LB and CCO) in a LTE-Advanced network with relay stations.

This prototype implements certain UMF functionalities, such as governing UC4 NEMs via a governance entity, including two coordinated NEMs. It is noted that at the second phase, the COORDINATION UMF functional block has not yet been fully specified. More details are described in the table above in “Self healing mechanism for Cell Outage Management NEM” and the “coordinating CCO and traffic balancing control based coordination” in the tables above (see Figure 25).

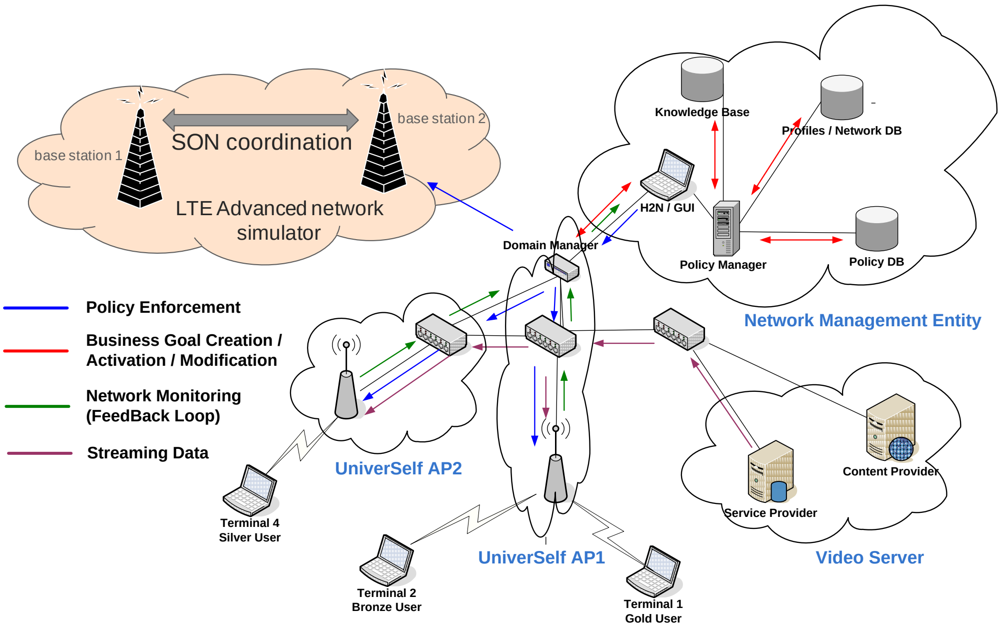


**Figure 25: UC4 prototype 1**

In the third phase, Prototype 2 is planned, (third year of the project) that will fully integrate all UMF components (GOVERNANCE, COORDINATION, KNOWLEDGE) in a single testbed. In particularly, Prototype 2 will implement coordination of two NEMs responsible for coverage capacity optimization (CCO) and interference coordination (ICIC) in LTE network. More details are described in the table above “ICIC and CCO optimization based coordination”, (see Figure 26).

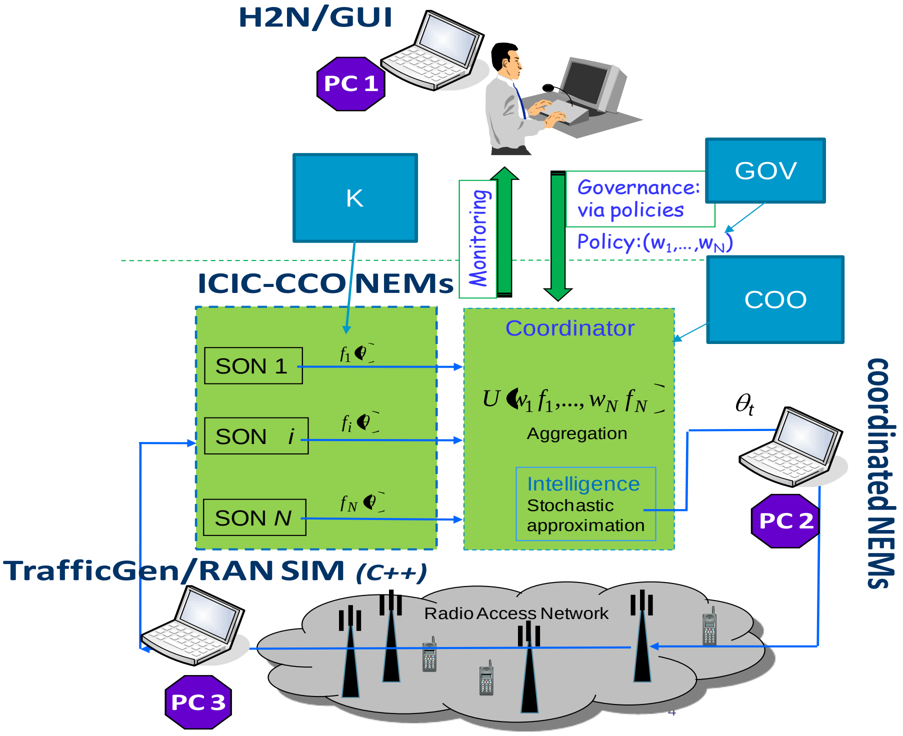


**Figure 26: UC4 prototype 2**

Prototype 3 is planned also in the third phase of UC4 lifecycle is based on the policy domain approach solution (described in the coordination table), and implement the coordination of two SON functionalities: mobility management and load balancing.

## 2.5.3 UC4 Key solution deployment assessment

Based on the prototypes of UC4 lifecycle, the evaluation of the gap between current technology and UMF will be carried out as part of the deployment assessment. The current technology of interest is LTE and LTE-Advanced. A particular focus will be given to the requirements for embedding the coordination building block of UMF implemented in the prototypes with respect to current architecture standardized.

Assessment will of course be also related to the success of the coordination functionality in terms of performance criteria indentified in the project, such as stability, and performance gain. These criteria have been addressed both in the first release UMF, and in the UC4 use case document.

Figure 27 shows a result produced by Prototype 1. The prototype implements a coordination scheme between two SON functionalities (NEMs), namely:

- Load balancing between backhaul link (macro to relay stations) and direct station (macro/relay) to mobiles links.
- Coverage Capacity Optimization (CCO) of relay stations.

The upper two figures present the network with SON coordination enabled. In the lower two figures the SON functionalities are disabled. On the right side one can see the link utilization which represents the links load. The worst backhaul and direct links are shown. The prototype shows that:

- The coordination mechanism succeeds to fully balance the cell loads.
- The network with coordinated SON have considerably lower loads and hence better capacity.
- The network remains stable during the SON process.
- The SON process is governed via the governance entity within the prototype / testbed.

These are examples of requirements and performance criteria that the prototype should meet. As stated in the previous section, prototype 2 aims at reaching even higher level integration of the UMF core blocks.

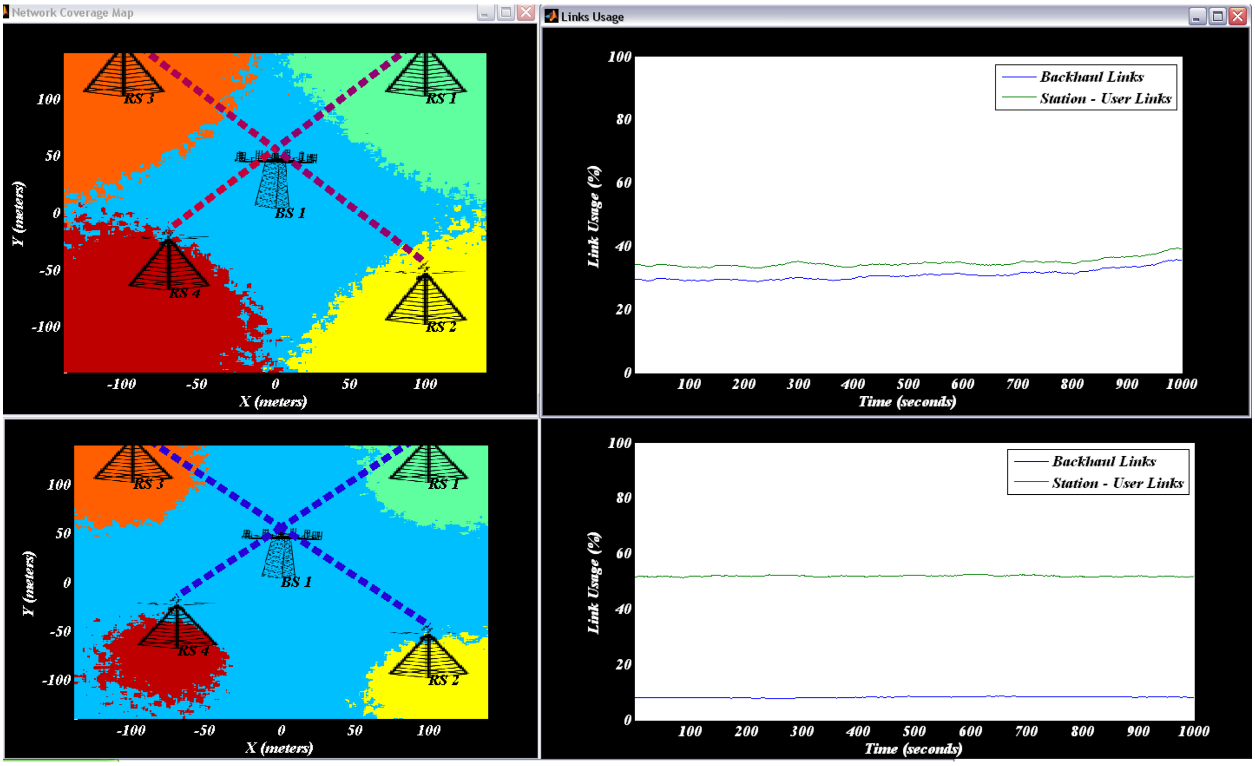


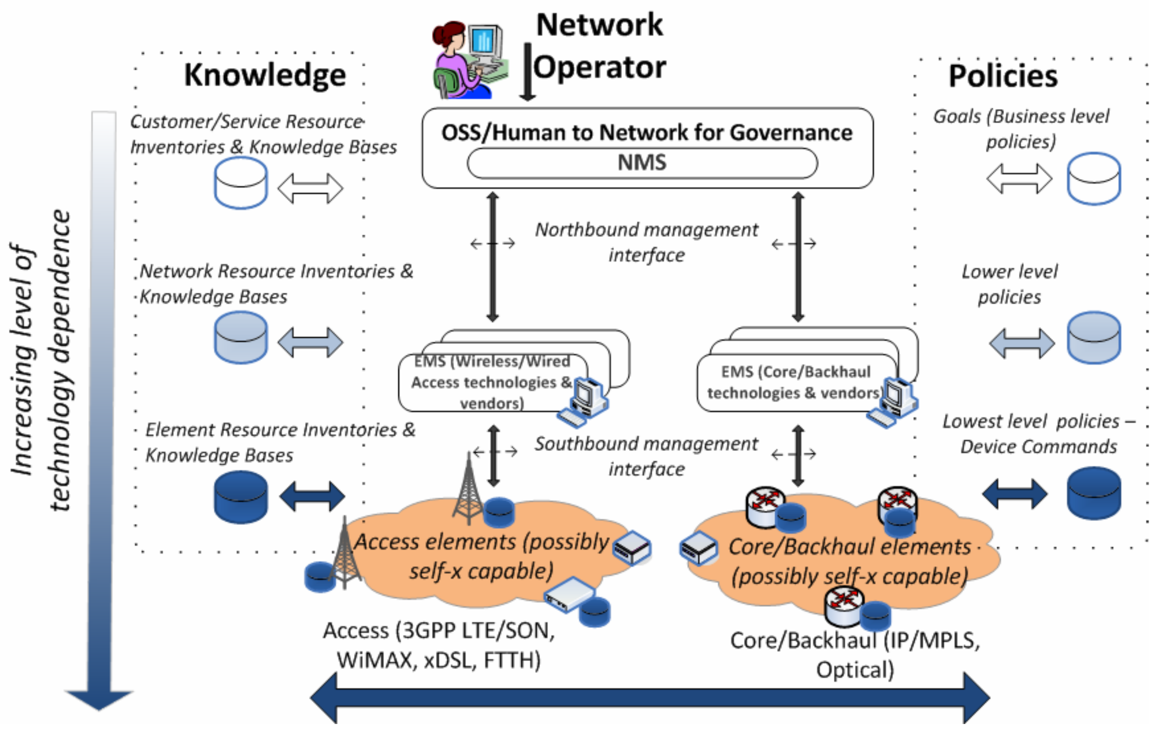
**Figure 27: Results coming from UC4 prototype 1**

## 2.6 Use Case 6

### 2.6.1 Use Case 6 key solution

Today's telecommunication operators worldwide need to confront changes in both their business and operational environment. Such changes significantly affect the operators' daily life and quick adaptation is key for them to keep following competition and increase profitability. Obviously, in the emerging and next generation market, operators that follow the trends and start addressing the challenges will have a competitive advantage over those who do not. But how can it be assured that a customer request for services and pertinent Quality of Service (QoS) will be satisfied in a fast, reliable and cost-efficient manner, exploiting technology advancements at the maximum levels? The solution can be provided by modernizing the supporting operations and management processes and preparing them to handle the foreseen dynamic service environment and autonomic capable network. The complete service and network management chain namely, Operations/Business Support Systems (OSS/BSS), Network Management System (NMS), Element Management System (EMS) and even the Network Element (NE) per se, needs to evolve from the cumbersome command and control mode to the so called "governance" mode. Governance aims at minimizing human intervention in the processes, at integrating and unifying disparate domains and technologies in the end-to-end service delivery path and at providing automated operations and decisions that are always aware of the dynamic context changes. Governance is materialized through a policy based framework [6] that provides the operator the means to set business level (technology agnostic) goals and objectives reflecting the highest level of a policies' hierarchy [7] and let them being automatically propagated to the network going through an arbitrary set of levels, where they are being transformed into lower level policies, until they finally reach the self-x capable element(s) in which they can be enforced in terms of lowest level, technology-specific commands.

In Use Case 6 it is assumed that a Mobile Network Operator (MNO) receives an urgent request for accommodating an additional traffic load, concerning a real time, video-based application (e.g. video-streaming of a programmed event), an associated set of user classes for the application, and a set of QoS levels for each user class of that application. In addition, the request can designate a specific location e.g. a conference centre of Piraeus region and a specific time period e.g. evening from 16:00 to 18:00, where the application will be delivered to an also (roughly) given number of conference attendants. The target is to ensure that the customer order will be satisfied in a fast, reliable and cost-efficient manner, by exploiting at the same time the technology advancement offered by network autonomicity at the maximum levels.


Network Operator
Knowledge
Customer/Service Resource Inventories & Knowledge Bases
Network Resource Inventories & Knowledge Bases
Element Resource Inventories & Knowledge Bases
OSS/Human to Network for Governance
NMS
Northbound management interface
EMS (Wireless/Wired Access technologies & vendors)
EMS (Core/Backhaul technologies & vendors)
Southbound management interface
Policies
Goals (Business level policies)
Lower level policies
Lowest level policies – Device Commands
Access elements (possibly self-x capable)
Core/Backhaul elements (possibly self-x capable)
Access (3GPP LTE/SON, WiMAX, xDSL, FTTH)
Core/Backhaul (IP/MPLS, Optical)
Increasing level of technology dependence

In order to accomplish this, several UMF Core Mechanisms and NEMs are cooperating. More specifically in showcasing and testing Use Case 6 the following UMF entities are involved and have been implemented: In the context of GOVERNANCE i) the Human to Network graphical user interface (H2N GUI) ii) the Situation Analysis and Diagnosis (SAD) UMF Core Mechanism iii) the Candidate Solution Computation (CSC) UMF Core Mechanism . In the context of COORDINATION iv) the coordination (COORDINATION) mechanism. And moreover three NEMs, v) the ICIC and CCO coordination (ICIC-CCO) NEM, vi) the Load Level Estimation (LLE) NEM and vii) the CORE Traffic Engineering NEM.

### 2.6.2 UC6 Key solution evaluation

The test-bed related to Use Case 6 has been developed as a Multi-Agent System (MAS) based on the JADE middleware platform and every UMF entity is implemented within one or more FIPA compliant intelligent agents that communicate with each other asynchronously, by sending and receiving ACL messages. A suitable ontology has been defined in order to allow the structured exchange of the necessary information.

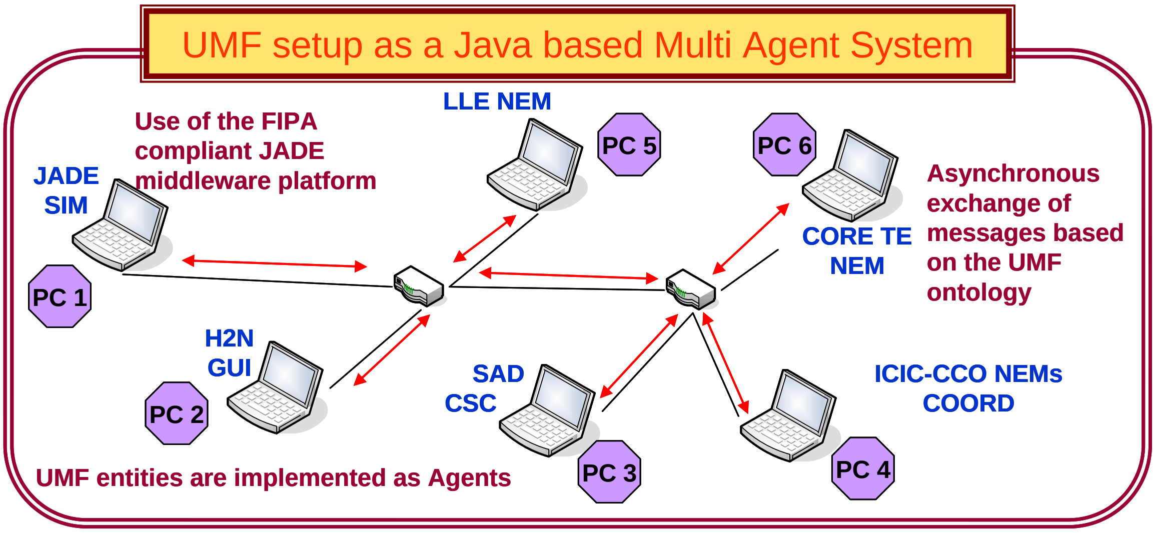

UMF setup as a Java based Multi Agent System
Use of the FIPA compliant JADE middleware platform
JADE SIM
PC 1
LLE NEM
PC 5
PC 6
CORE TE NEM
Asynchronous exchange of messages based on the UMF ontology
H2N GUI
PC 2
SAD CSC
PC 3
ICIC-CCO NEMs COORD
PC 4
UMF entities are implemented as Agents

**Figure 28: The setup of the UC6 prototype environment**

The load level estimation (LLE) mechanism addresses the problem of predicting long-term network traffic load for different areas or time-periods. The motivation behind it is the provisioning of network behaviour throughout time and space that might lead to improved allocation, management and utilization of the available resources. The concept is based on the observation that network traffic usually follows certain hourly-, daily-, and even weekly- patterns that derive from different characteristics of user behaviour and network usage habits, which in turn may vary from one area to another.

For the online building of such knowledge, a machine learning algorithm has been employed, namely a variant of the Self-Organizing Map (SOM) [18] known as Parameter-less Growing SOM [19]. SOM is a special type of artificial neural network with capabilities to reduce vast amounts of multidimensional data, project and cluster them in the 2-dimensional space while preserving their initial characteristics and relationships at some adequately high degree. In this particular scenario, it is constantly provided with measurements of network load, accompanied by parameters indicating the time of the observation (i.e. the time of day, the day of week, the week of year) as well as its location (i.e. a base station identifier). The mechanism is then expected to discover the correlations, if any, between the input data parameters and the dependent variable in question, (i.e. the load), in a way similar to how a human would observe, for instance, that "in the business area X, network usage is increased during morning and midday while in the residential area Y network usage is usually increased during afternoons, excluding Saturdays."

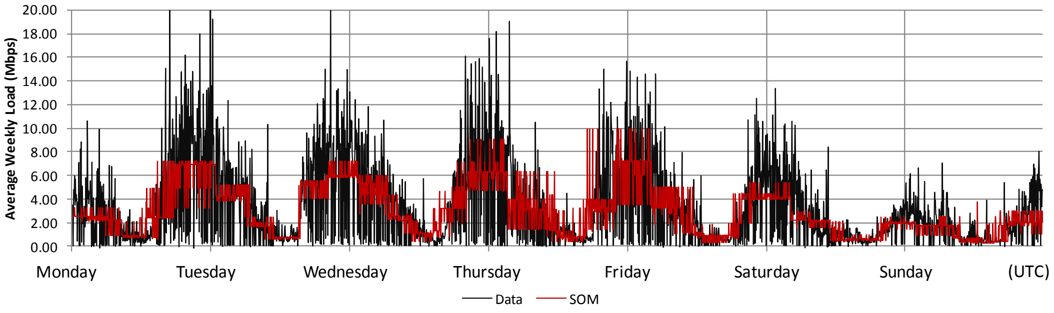


**Figure 29: Load of base station 4 compared to the prediction of the LLE NEM using SOM**

The NEM has been validated both using simulation data (online application in the context of use case 6 demonstrations), as well as real data, acquired from the CRAWDAD database [20], which were collected from, (i) the Dartmouth College [21], and (ii) the "Île sans fil" Wi-Fi hotspots in Montréal, Québec, Canada [22]. Indicatively, Figure 29 depicts averaged weekly predictions for a particular base station, the formulated pattern, and how the mechanism manages to follow it. The deviation between the average actual and predicted load is better depicted in Figure 30, where a moving average has been applied to the same two time series, and similarly in Figure 31 for a different area.

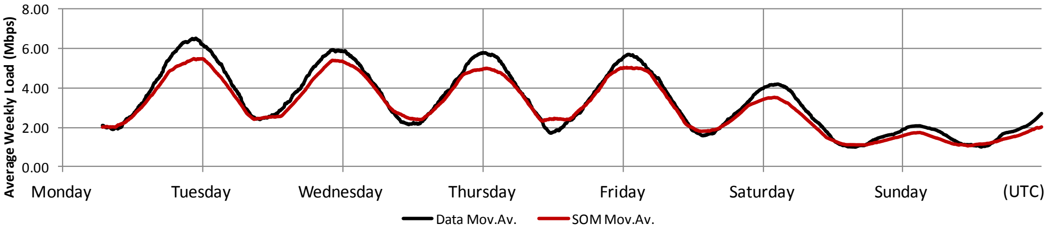


**Figure 30: Centered moving averages of traffic load of base station 4 compared to the prediction of the LLE NEM using SOM**

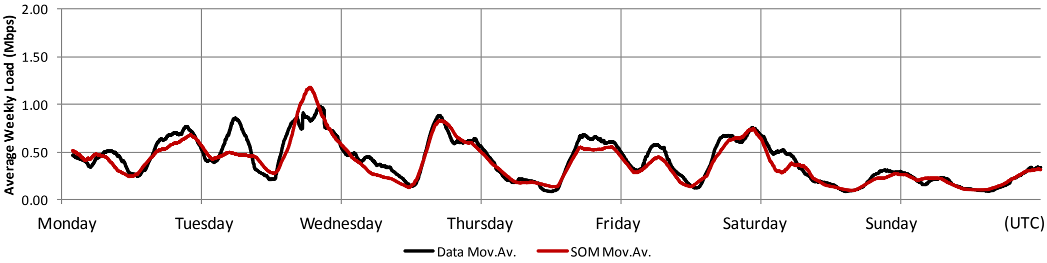
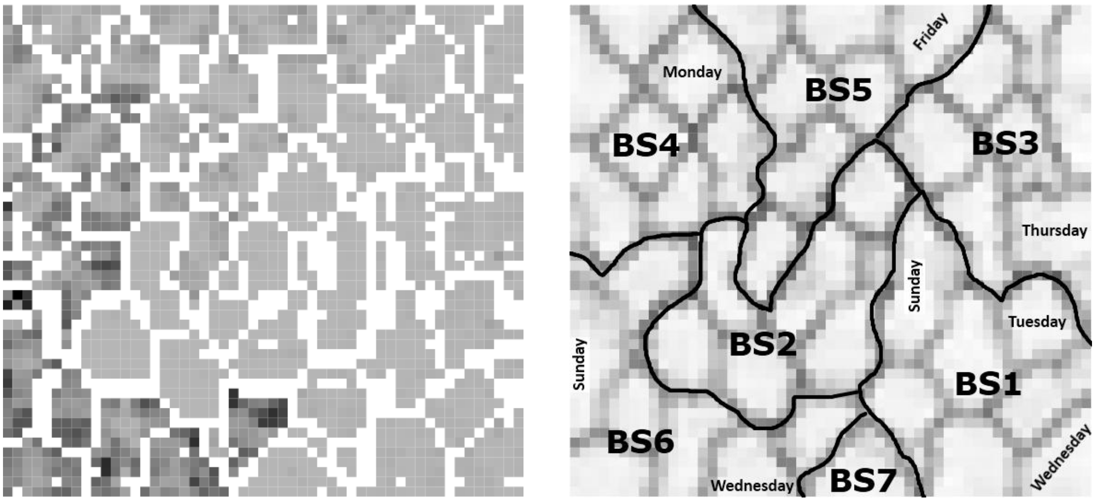


**Figure 31: Centered moving averages of traffic load of base station 3 compared to the prediction of the LLE NEM using SOM**

Figure 32a) and Figure 32b) are two visualizations for the same snapshot of the generated map by the LLE NEM, where each pixel represents a point in the multidimensional space of the input data and, in the case of Figure 32a), its brightness represents the load (the darker the higher), while in the case of Figure 32b) it represents the distance between its neighbouring pixels (again, the darker the higher). These two particular visualizations, if combined, can unveil significant correlations between the time, the area, and the respective network load, that would otherwise require time-consuming data-processing and -analyzing tasks to be carried out by humans. For instance, one can easily conclude that BS4 and BS6 are by far the most loaded base stations. The most loaded days and even hours of each can be also noticed (although not all day labels are drawn in this figure for clarity reasons).



(a) (b)

**Figure 32: Visual output of the Self Organizing Map generated by the LLE NEM. (a) Visualization of load, where lighter is lesser and darker is higher and (b) U-matrix of the same grid depicting the formulated clusters on the 2D projection of the processed input data**

In the core network segment, the CORE – Routing Traffic Engineering NEM provides a solution to the problem of routing optimization with respect to different operator's policies. The offered solution is based on a heuristic algorithm that evaluates network’s status and finds the optimal routing configuration, exploiting the capability of splitting traffic and forwarding it through different multiple MPLS paths, when this is needed. Main objectives that have been examined are load balancing and energy efficiency. Load balancing is achieved through splitting traffic, while energy efficiency is achieved through the aggregation of traffic into minimum number of links and deactivation of unused network elements. Furthermore, our solution comprises two important features, monitoring network and informing other NEMs.

In particular, Figure 33 depicts network’s status after the insertion of the first traffic request, while the operator’s objective is load balancing. Traffic is split and is forwarded through two LSPs, resulting in the activation of seven links between ingress (LSR1) and egress (LSR12) routers. Availability and utilization of links is presented in Figure 34 and Figure 35. Figure 36 depicts network’s energy consumption for the accommodation of this request.

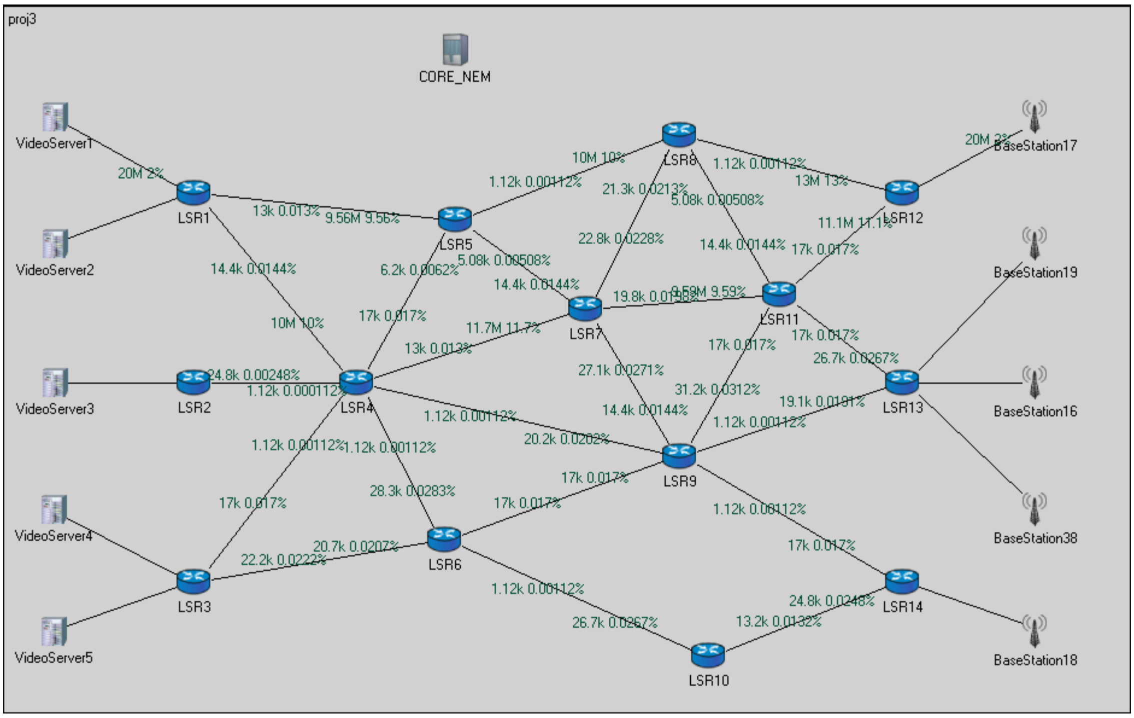


**Figure 33: Multipath enabled (Load Balancing)**

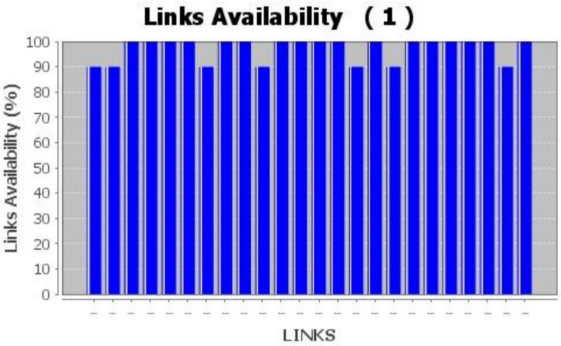


**Figure 34: Availability of links**

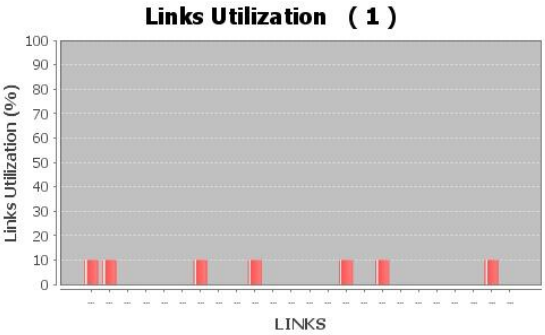


**Figure 35: Utilization of links**

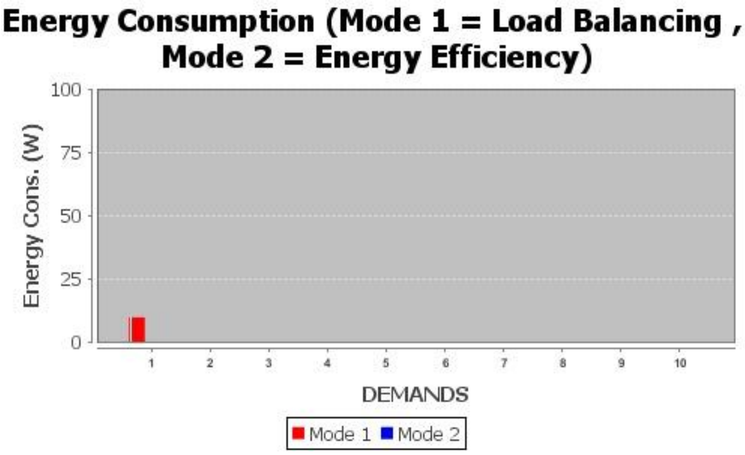


**Figure 36: Energy consumption**

When the operator's policy is changed, the new goal is an energy efficient network operation. The next request should be accommodated with respect to this objective. Furthermore, already established paths should be evaluated in order to increase energy savings. Taking this into account, the algorithm evaluates network's status, the new policy and the new traffic demand and finds the routing configuration that is depicted in Figure 37. Both traffic demands are forwarded through single LSPs that activate minimum number of links.

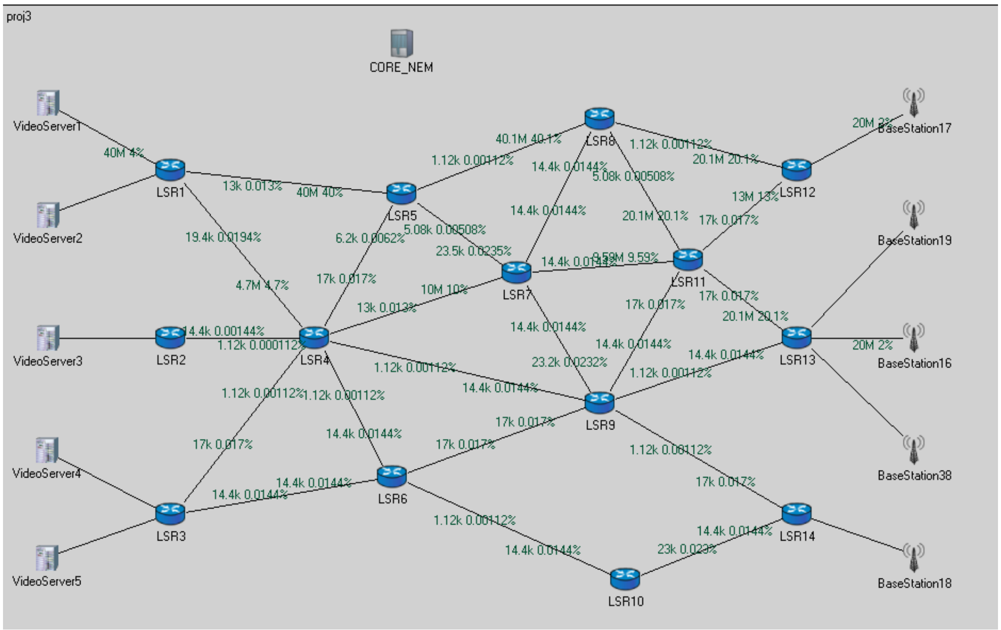
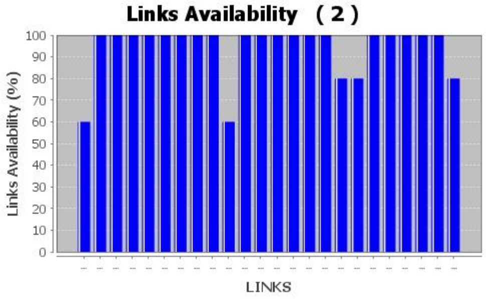


Figure 37: Multipath disabled (Energy Efficiency).



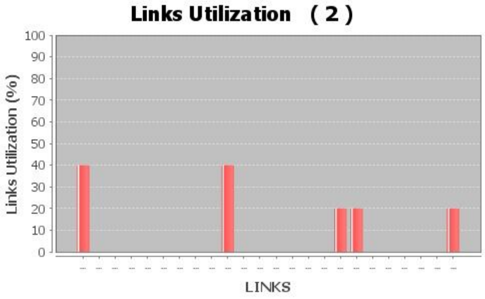


Figure 38: Availability of links

Figure 39: Utilization of links

Availability and utilization of links is presented in Figure 38 and Figure 39. The consumed energy for the accommodation of the two demands is depicted in Figure 40. In the same figure the amount of energy savings is evident for the two policies.

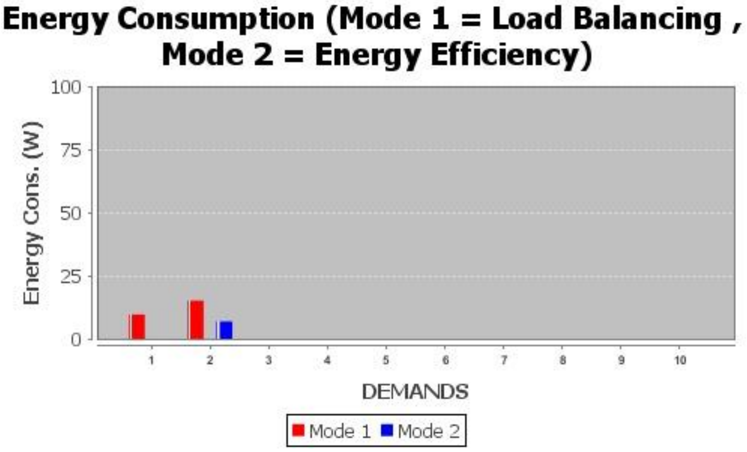


**Figure 40: Energy consumption**

Last but not least, Use Case 6 gives a positive proof of concept on designing and deploying governance systems that:

- Provide the operators the means to express their goals and govern (control) their possibly self-x capable network through a robust policy based framework.
- Achieve end-to-end integration, federation of wireless/wired access and core/backhaul network segments and their associated management systems, and all these under the "auspices" of operator policies.
- Maintain and exploit always-up-to date inventories and knowledge (possibly derived through incorporated learning mechanisms) in all the situations above, thus increasing the reliability and adaptability of management decisions, and contributing to autonomicity.
- Exploit the self-x features of the autonomic infrastructures/system in the management functionality in order to adapt the networks to the different situations encountered.

### 2.6.3 UC6 Key solution deployment assessment

The deployment of UC6 brings into surface its numerous merits and strengths in several fields of quality. The Human to Network (H2N) graphical user interface is of great usability, allowing the operator to navigate easily in a wide range of information about his multi-technology and multi-vendor infrastructure. At the same time, its capabilities for defining models, associations and policies are almost endless in terms of extensibility. Scalability is also not an issue, since this Governance implementation can handle any number of NEMs, policy rules, network segments, requests etc without any deficiency in its operation. Moreover, ultimate interoperability is ensured by using the corresponding UMF interfaces.

As regards the RAN and CORE network segment optimization NEMs that are involved in the UC6 prototype, both of them show evidence of supreme efficiency in their functionality and their ability to manage resources in the most effective way. They are also characterized of an extended level of adaptability, being able to evolve together with the highly heterogeneous environment in which they operate, while their interoperability is concretely based on their compliance with the UMF specifications.

Finally, the knowledge building NEM exhibits remarkable accuracy in its estimations on the load levels, contributing very much in the accuracy of all the related decisions that are made based on this knowledge. Nevertheless, the computational resources that it requires in order to run are nothing special, a fact that applies as well to all the entities that comprise the UC6 prototype implementation. Its usability and adaptability are also top level, since it is capable of learning and making estimations for any kind of input data parameters and a dependent variable in question, presuming of course that there is a relation between them. The exploitation of the relevant UMF communication mechanisms guarantees a unique interoperability with any NEM that is interested to retrieve knowledge.

## 2.7 Use Case 7

### 2.7.1 Use Case 7 key solution

To ensure cost-competitive broadband services, operators must dramatically reduce operation costs. Supervision, diagnosis and repair processes require good hardware and software tools to be able to quickly react to problems and restore the affected services with as little impact as possible. Existing tools and processes, while having heavily automated the network operation, seem not to be able to address the requirements of the highly dynamic and distributed future networks. While it is widely believed that autonomic networks with self-* capabilities will help in the automation of the provisioning and runtime phases, there is still the need of providing the human operator with the tools to command and guide the behaviour of the autonomic elements.

The use case Network and Service Governance aims at solving the following problem: to provide operators with automated decision oriented operational tasks and reduce the human intervention for service & network configuration/deployment. At the same time, the QoS requirements for the customers should be also fulfilled. To this end, UC7 targets to demonstrate the feasibility of a policy-based network management approach, governing both fixed and mobile network segments. The mobile network is based on a WIFI connection on DSL network, while for the fixed environment Fiber-To-The-Home (FTTH) is the selected technology.

In both cases, UC7 aims to develop self-* functionalities (e.g. self-monitoring, self-diagnosis and self-healing) which will enable the early detection and resolution of network and service problems. Figure 41 graphically shows the described design, where the same UMF core implementation instance is able to govern both FTTH and wireless network segments.

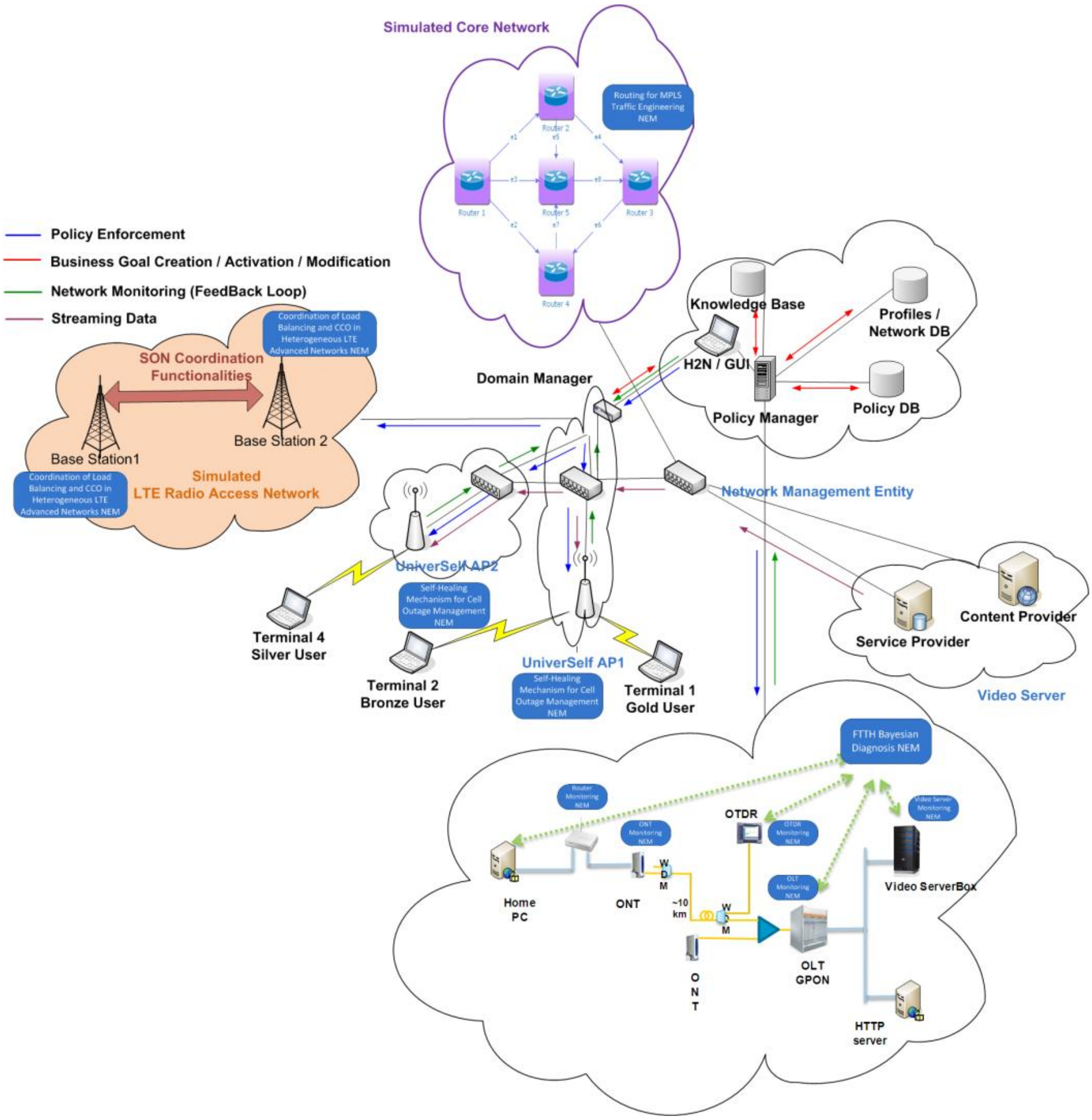

Simulated Core Network
Routing for MPLS Traffic Engineering NEM
Router 1
Router 2
Router 3
Router 4
Router 5
e1
e2
e3
e4
e5
e6
e7
e8
Policy Enforcement
Business Goal Creation / Activation / Modification
Network Monitoring (FeedBack Loop)
Streaming Data
Coordination of Load Balancing and CCO in Heterogeneous LTE Advanced Networks NEM
SON Coordination Functionalities
Base Station1
Base Station 2
Coordination of Load Balancing and CCO in Heterogeneous LTE Advanced Networks NEM
Simulated LTE Radio Access Network
Domain Manager
Knowledge Base
H2N / GUI
Profiles / Network DB
Policy DB
Policy Manager
Network Management Entity
UniverSelf AP2
Self-Healing Mechanism for Cell Outage Management NEM
Terminal 4 Silver User
Terminal 2 Bronze User
UniverSelf AP1
Self-Healing Mechanism for Cell Outage Management NEM
Terminal 1 Gold User
Service Provider
Content Provider
Video Server
FTTH Bayesian Diagnosis NEM
Router Monitoring NEM
ONT Monitoring NEM
OTDR
OTDR Monitoring NEM
Video Server Monitoring NEM
OLT Monitoring NEM
Video ServerBox
Home PC
ONT
WDM
~10 km
WDM
ONT
OLT GPON
HTTP server

**Figure 41: UC7 testbed managing FTTH and wireless testbeds including UMF core & NEMS**

In order to fulfil its mission, the following UMF Core functions have been implemented:

- GOVERNANCE functions: The following GOVERNANCE functions have been implemented (Java) in UC7 proof-of concept prototype to enable the dynamic definition of business goals, their translation to policy rules, the enforcement of the derived rules/actions and their evaluation:
    - Policy Derivation & Management (PDM) and its operations:
        - Policy Translation
        - Policy Efficiency
    - Human to Network interface (H2N)
    - NEM Management
- KNOWLEDGE functions to support the necessary information exchange and storage required to ensure policy-based management:
    - Interaction with Information Sharing
    - Information Collection & Dissemination, Information Storage and Indexing

In order to fulfil its mission, a set of NEMs have been designed, implemented and tested in UC7 testbeds, including:

- FTTH Bayesian Monitoring and Diagnosis NEMs: A set of monitoring NEMs able to retrieve data from the different network elements and probes in the FTTH access segment have been implemented. Based on the gathered information, a diagnosis NEM finds the most probable root cause of failure and the corresponding probability, using Bayesian inference. This NEM is described in detail in Deliverable D3.7 [4].
- Routing for MPLS Traffic Engineering NEM has been implemented including Matlab simulations and corresponding Graphic User Interfaces for visualising the network topology, providing dynamic routing adaptation of traffic flows across network elements. Evolutionary Techniques have been utilized in order to exploit the benefits stemming from these approaches compared to other optimization techniques such as the constrained non linear optimization technique provided in MATLAB toolbox. Multi-point search in the area leads to fast convergence to near optimal solutions. The details of our approach are described in depth in Deliverable D3.5 [3].
- Self-Healing Mechanism for Cell Outage Management NEM has been implemented in a proof-of-concept real implementation for Soekris devices (linux-based programmable Access Points). This NEM adjusts the transmission power of neighbouring base stations in case of a failure in one of the access points. This NEM has been extended with simulation results that have been described in depth in Deliverable D3.5 [3].
- Coordination of Load Balancing and CCO in Heterogeneous LTE Advanced Networks NEM, already described in section 2.5.

The above mentioned NEMs can be governed via GOVERNANCE core mechanism, and exchange information through the KNOWLEDGE block. The NEMs and main core functions have been implemented and fully tested in the wireless testbed. Regarding the FTTH segment, monitoring and diagnosis NEMs have been implemented and tested, while the implementation of the interfaces between NEMs and UMF core is ongoing at the moment of the edition of this deliverable.

### 2.7.2 UC7 Key solution evaluation

As mention in the previous subsection, Network and Service Governance Use Case has been deployed for evaluation on two testbeds: a fixed one, based on FTTH technology, and a wireless one. Figure 41 shows a schematic view of the FTTH testbed, and the corresponding NEMs. The testbed represents the typical deployment of an FTTH network, from the Central Office (CO), to the customer premises. The setup is a point to multipoint connection, with the fiber cable starting at the OLT (Optical Line Terminal), and being divided by two levels of splitters before reaching the customer home. This allows a single fiber from the CO to be shared among a number of subscribers. The fiber at the customer premises is connected to an ONT (Optical Network Terminal), which converts optical signals into electrical signals that can be used within the home. A router is then connected to the ONT, and then a set of customer equipments, typically one or more PCs, and TV sets. Apart from the network and service equipment, an optical reflectometer is located at the CO, to help in the detection of the location of cut failures. For each of the network and probe elements described, a monitoring NEM has been implemented, to interact with each of those in order to gather data about their status and behaviour. These data is used by the FTTH diagnosis NEM, which implements a Bayesian diagnosis approach able to infer the most probable root cause of failure.

FTTH monitoring and diagnosis NEMs have been implemented as JADE (Java Agent Development Framework) [5] multi-agent platform, where each agent embeds the behaviour of a NEM (monitoring or diagnosis). JADE is a multi-agent Java platform based on FIPA standards, whose development is led by Telecom Italia. JADE allows the distributed deployment of Java coded agents with minimal requirements of CPU and memory on servers and devices. Given the impossibility of embedding software into the hardware elements of the testbed, due to vendors' restrictions, the developed NEMs have been installed in a virtual machine running Ubuntu 11.

As regards the wireless testbed, it comprises of mobile devices and programmable access points (Soekris devices). They are based on Geode single chip processors with an x86 architecture and target on running open source operating systems, like FreeBSD, OpenBSD, NetBSD and Linux. The key characteristic of these devices is the provision of greater programming flexibility than dedicated network devices. The access points provide Wi-Fi connection to a set of terminals.

In this UC, we showcase the following steps: First, the use of the governance framework is demonstrated, to specify the operator's parameters (supported services, user classes, available levels of availability, reliability, speed and security, etc) and a set of predefined business goals by the network operator which are related to the different classes of mobile devices. Then, traffic is injected considering a video service in the mobile terminals. The video service will be provided by a local video server. The implemented H2N/GUI evaluates the parameter values using a threshold-based approach and takes into account the defined business rules taking into account the user class (e.g. A Gold class user consuming a Streaming service should experience "Excellent" availability, "Excellent" reliability and "Excellent" speed). The outcome of this evaluation is the triggering of a network management action. Specifically, the actions that are demonstrated include the following possibilities:

A. Splitting of traffic flows in the core network (implementation in MATLAB using Genetic Algorithm toolbox), showcasing the following algorithms: Constrained Non Linear Minimization technique (FminCon) provided in MATLAB toolbox, Genetic Algorithm and Particle Swarm Optimisation (PSO) algorithm.
B. Coordination of two SON functionalities in a LTE-Advanced heterogeneous network with macro- and relay stations (implementation in Matlab network simulator). A high link usage event occurs in operator's cellular network (LTE). The Governance FB receives this event and the automated translation of business goals to policy rules triggers the following remedy action: SON coordination between cells for load balancing. As an outcome, the link usage is fairly distributed to neighbouring relays.
C. Handover of a user to the neighbouring UniverSelf Access Point (AP): The injection of load in the network through the video server increases the cell utilisation. The governance framework specifies a handover decision for the bronze users.
D. A self-healing action in the network – in this case, we consider that one of the two APs faces a failure. The system 'understands' these failures and triggers the Tx power increase of the neighbouring AP to handle the users.

In the following text, some key results obtained from the evaluation of UC7 using the above mentioned proof-of-concept prototype system are analysed in more detail.

Governance Framework

The scalability of the proposed policy translation process of the governance framework is examined under various numbers of generated policies. In general the number of policies generated by H2N/GUI depends on the size and the heterogeneity of the network (number of different services, technologies etc.). A set of measurements were performed in the aforementioned testbed in order to study the performance of the proposed policy translation process in terms of delay under different load of generated policies. The results of this study are illustrated in Figure 42. It is obvious that the delay of the translation process is insignificant for a low number of deployed policies (e.g. in case of 3 policies the translation delay is around 2.5 sec), while it highly increases in case of a high number of generated policies (e.g. in case of 50 policies the translation delay is around 6 min). However, since this procedure is to be executed offline and during the system bootstrap, the introduced delay does not affect the operation of the overall system. Moreover, the introduction or update of the defined business goals does not require the re-execution of the whole translation procedure; the sole evaluation and translation of the particular business goal should be only realized.

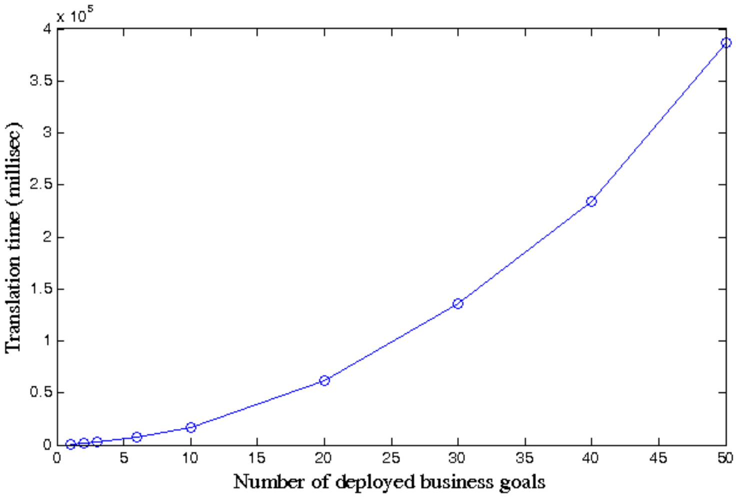


**Figure 42: UC7 Results: Policy translation delay**

Splitting of traffic flows in the core network

It is proven that evolutionary techniques achieve better load distribution over the core network. The Genetic algorithm and PSO produce similar results and outperform the FminCon routing algorithm. Figure 43 highlights the outcomes of our experimentation. More specifically, we have tested 100 random topologies and as it is shown in Figure 43 (a), Evolutionary Techniques obtain better performance in terms of edge utilization in approximately 40% of the tested topologies. FminCon performed better in approximately 30% of the tested topologies; in addition, the rest 30% of topologies resulted in similar performance for both approaches. Figure 43 (b) shows that FminCon achieves 15% average increment of utilized edges for the topologies resulting in better performance compared to the other algorithms. The utilized edges are increased significantly (i.e. 30% increment) for the cases that Evolutionary Techniques result in optimized performance.

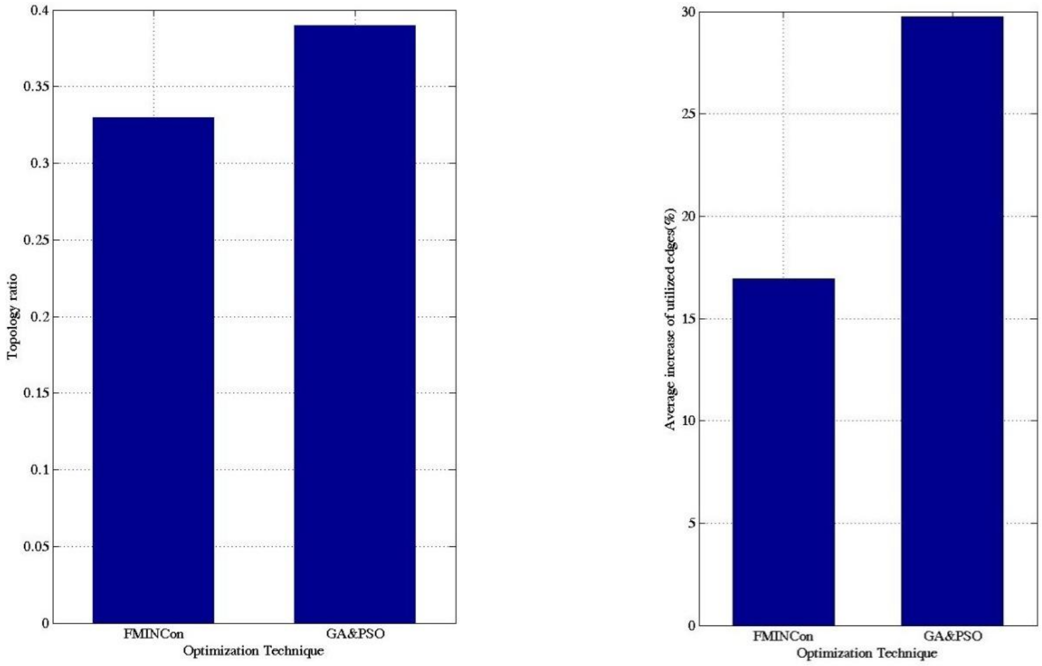


**Figure 43: (a) Efficiency of Optimization Techniques over 100 topologies (b) Average increase of utilized edges over the 100 topologies**

Self-healing

Below, we provide some indicative simulation results for a more extended practical WLAN topology.

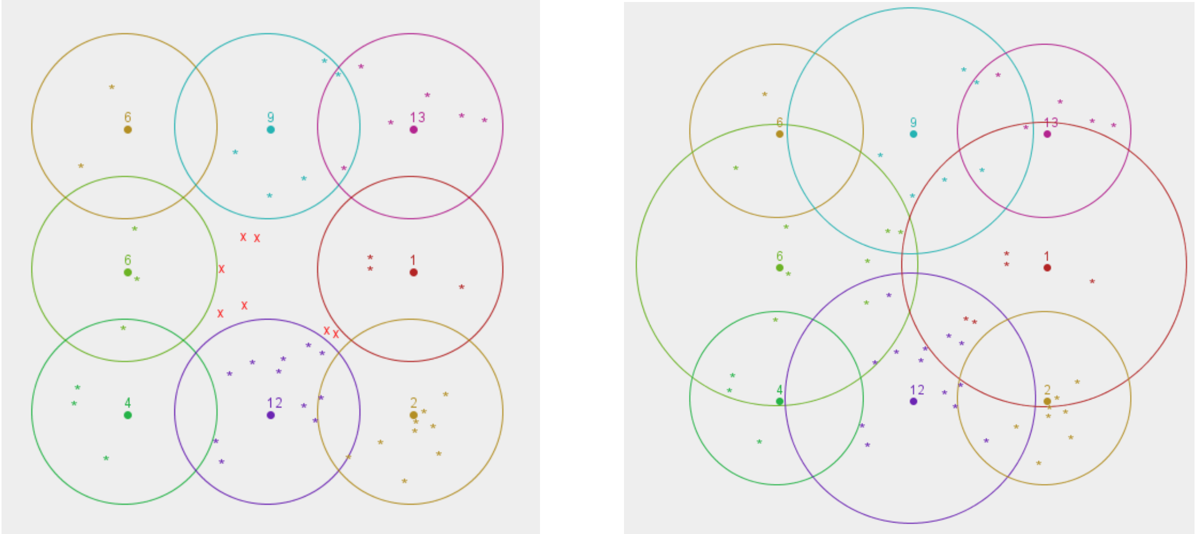


**Figure 44: a) WLAN topology after the failure of the central AP.**
**Red 'X' symbolizes the disconnected MTs., b) WLAN topology after the cell outage compensation**

We consider a grid topology of 9 APs and 49 randomly distributed mobile terminals (MTs). The channel assignment for each AP is shown by an integer (1-13) above its position point.

Figure 44 a) depicts the WLAN topology after the central AP failure where 7 MTs are affected (disconnected). The created coverage gap is covered by the neighbouring AP as shown in Figure 44 b) based on the calculations of the proposed fuzzy inference system. The evaluation metrics for the algorithmic case and power maximization case are shown in the following table.

| Number of MTs | Evaluation Metrics | | | | | |
|---|---|---|---|---|---|---|
| | Load balance index (β) | | Reconnected MTs' percentage ($P_r$) | | Total Interference Indicator ($I_t$) | |
| | Self healing Algorithm | Power Maximization | Self healing Algorithm | Power Maximization | Self healing Algorithm | Power Maximization |
| **49** | 0.711 | 0.650 | 100.0 | 100.0 | 2.152 | 2.662 |

**Table 10: Evaluation metrics**

The adjusted transmission power levels of the compensating APs are shown below. We consider that the maximum transmission power for the APs is 20dBm.

| AP id | Tx Power (dBm) | Assigned Channel |
|---|---|---|
| **2** | 16.9 | 6 |
| **4** | 15.0 | 9 |
| **6** | 15.2 | 12 |
| **8** | 17.1 | 1 |

**Table 11: Tx power levels and assigned channel for the compensating APs after the fuzzy inference system calculations**

Simulation results for this indicative grid topology of APs and randomly positioned MTs show that our algorithm can provide a good solution for the coverage-interference trade-off problem while ensuring sufficient load fairness among APs.

At this point, it is worth noting the different implementation choices (JADE, OSGI) employed in the development of FTTH and wireless NEMs. This use case demonstrates that the UMF specifications delivered by

WP2 can be effectively implemented to govern NEMs with independence of the underlying technologies. The Unified Management Framework does not impose any implementation mechanism, but ensures the efficient management of autonomic network and services as long as the proper interfaces are used. Vendors can then deliver they own NEMs without exposing the inner details of their solution, and operators can still govern those NEMs with their UMF-compliant management system.

In the current telecommunication context of high competitiveness and need of cost reduction, solutions for the automation of the network operation as the ones provided by autonomic networks are becoming more and more important and strategic for network operators. With the usage of self-optimization functionalities which control network resources, as the ones achieved in this use case, better reactivity and hence performance gains are achieved. Self-diagnosis and self-healing lead to a faster isolation of the failures, automatic mitigation of the impact and implementation of repair actions. This technological advance improves the QoS offered to the end-user by preventing problems and lowering the Mean Time-to-Repair, which leads to an increased service availability. This translates into a reduction in the cost per service and the need of specialized personnel, both of which have an important impact in the company OPEX.

All of the above raises the question of how can the operator maintain its control over the autonomic network. The implemented governance framework revolutionizes management of large heterogeneous network infrastructures, since it enables network operators to automate end-to-end management of their services assuring operational processes like monitoring and diagnosis. This plays an essential role in alleviating human intervention and decreasing operation complexity and costs of future networks. With this in view, network operators will thus have the opportunity to further enhance the quantity and quality of services across multiple heterogeneous domains meeting customers' expectations and further improving customer experience.

One of the main concerns for operators when considering the rollout of an autonomic network is the management of legacy networks. Network and Service Governance Use Case has proved that UMF and NEMs can be used to seamlessly govern both types of network elements. On one hand, the software in the wireless testbed is embedded into the Soekris access points, as it would be in a fully autonomic solution. On the other hand, as the equipment in FTTH testbed did not allow software embodiment, NEMs have been developed as external agents able to communicate with the network devices, thus probing that UMF can be used for the management of legacy networks as well.

In summary, Network and Service Governance Use Case has demonstrated that UMF can effectively manage both autonomic and legacy networks from a high-level point of view. This is achieved with independence of the implementation details and underlying technological choices that are left to NEM developers. This proof of concept has demonstrated that self-* capabilities can be really developed for different types of network segments. The evaluation has been performed on hardware testbeds, with the aim of helping to convince network operators of the feasibility and advantages of the deployment of a UMF compliant autonomic network.

### 2.7.3 UC7 Key solution deployment assessment

The Network and Service Governance Use Case has been evaluated following the ISO/IEC 9126-1 quality model. The analysis shows that the use case mainly contributes to the usability, efficiency, extensibility, scalability and interoperability criteria.

- The attractive, friendly and easy to understand and operate Human to Network interface contributes to the usability of the use case.
- The UC7 Network and Service Governance have proved its efficiency given its ability to manage the network with good level of performance and adequate time response, while the implemented NEMs are being able to run with a limited amount of resources (for instance, the NEMs in the wireless testbed are embedded into the Soekris Access Points)
- Since the most network-related functionalities of the system have been implemented using a set of distributed NEMs, the use case can be easily extended by adding new NEMs or extending the already implemented NEMs when new functionalities are needed. Even more, the implementation choices (JADE agents on the FTTH testbed, OSGI in the wireless segment) also facilitate the inclusion of new functionalities in the existing NEMs.
- The fact that the use case is based on distributed NEMs, being there embedded into the network elements or not, enables the scalability of the system. In the case of the wireless testbed, the growth of the system is intrinsic to the growth of the network, since the software is embedded into the Access

Points. In the case of the FTTH testbed, the scalability is achieved by deploying new agents to handle new network elements. In addition, the Governance functionalities achieve acceptable performance even under large network topologies (high number of policies).

- The coexistence of two different implementation choices (JADE and OSGI) and network elements from different vendors (Soekris, Alcatel) has proved the interoperability of UC7 solution.

Network and Service Governance Use Case has demonstrated that it can efficiently govern existing wireless and FTTH network segments. The deployment of the software in real production level is subject to the technological requirements of the solution. In a wireless segment this is translated into the utilization of Soekris Access Points, since it is in these devices where the Self-Healing Mechanism for Cell Outage Management NEM is embedded. In the FTTH testbed, only adaptors for Alcatel-Lucent ONTs and OLTs have been developed at the moment, since these are the devices available in the testbed. Nevertheless, the modularity of the design and the implementation choices guarantees that adaptors for other vendor's equipments can be easily implemented. A dedicated server for the UMF core should also be setup, and HTTP channels opened so NEMs can communicate with the UMF core blocks using REST interfaces.

In addition, operators usually rely on the OS systems they already have, so at least in a first phase, interfaces with the operator OSS should be designed and implemented. In particular, there are two types of existing systems strongly related: inventory and alarms management systems. The current version of the FTTH testbed needs access to the inventory data, so each agent gets attached to a given network element. Concerning the alarms, operators usually have a centralized view of the alarms using systems that act as collectors of all the alarms. Even when the UMF core through the H2N interface can inform the operator about the malfunctioning of the system, it may be convenient to add interfaces to the existing management systems of the operators in order not to change significantly the daily operation.

# 3 UMF deployment assessment discussion

In the previous section, UniverSelf key solutions are presented and their first deployment assessments are described. Additionally to the deployment assessment of solutions, UMF by itself, as a Framework, should also be assessed and this is the objective of this section.

Making the UMF deployment assessment is related to its credibility, leading the way to the implementation and deployment of UMF compliant management systems by operators and vendors. In the following section we identify some first elements to show our progress related to UMF deployment. It will be refined in the next deliverable related to deployment results following UMF Release 3 and integrated prototype.

Several dimensions can be considered to assess UMF (as a framework) deployment.

First UMF is network technology independent and must support the management of self-management mechanisms on various network and service technologies. In UniverSelf, we are addressing six uses cases targeting various network technologies in various network domains for various self-management mechanisms. This diversity was a main requirement to enable a good unification as done for UMF release 2. Here we can have evidence that we are on the right direction to be able to cope with various network technologies or self-management features. However UMF is not only a specification but also a set of tools to help deployment. As part of the existing implementations, we demonstrate key UMF mechanisms and prove their usefulness (e.g. coordination mechanisms).

Second, UMF needs to be adopted by industrial, especially operators and vendors. Certification by the means of Conformance Testing and Interoperability Testing is needed. Both testing can be applied on the UMF as a whole and on each UMF block separately. The purpose of conformance testing is to determine to what extent a single implementation of a particular standard conforms to the individual requirements of that standard. The purpose of interoperability testing is to prove that end-to-end functionality between (at least) two communicating systems is as required by the standard(s) on which those systems are based [29]. Therefore in UMF case, the conformance testing will assess that the UMF implementation conforms to the individual requirements of the UMF described in D2.2, while the interoperability testing will prove that the UMF end-to-end functionality is as required by the D2.2. In order to assess the compliance of a specific NEM to UMF we should perform Conformance Testing on the NEM in order to assess that the NEM implementation conforms to the requirements imposed by the UMF specification. In addition, the execution of Interoperability Testing on the NEM with the UMF serves as the “Qualified Equipment” and the NEM serves as the “Equipment Under Test”, will showcase if the end-to-end functionality between the two systems conform to the UMF and NEM specifications.

In order to tackle with complexities, we should create a set of detailed Test Cases during the Conformance and Interoperability tests in order to examine the system behaviour under the various possible states/behaviours of a specific complexity. Since the implemented software passes successfully both the Conformance and Interoperability test, we can assume that the specific complexity is handled.

At the end, standardization is tightly related to UMF deployment.

Deploying UMF is also related to the migration path from legacy networks to Future networks. Operators must be able to deploy an UMF management system on legacy networks. UniverSelf is already providing concrete deployment scenarios. For illustration, we provide a first attempt to map UMF functions on a 3GPP LTE network architecture (it is available in annex A of the deliverable).

Discussion and assessment will be refined in the next deliverable release.

# 4 Conclusions and next steps

This deliverable is the first release of the deployment results and they will be refined for the next deliverable (D4.12, June 2013).

Making a deployment assessment is not a straightforward action as there are many facets to consider. For example, it corresponds to assess the gap between our project solutions and their deployment in real networks, the migration path from legacy management system, the implementation maturity, the performance… In addition to these facets, implementation of UniverSelf solutions is following Use Case life cycles. It implies that solutions don't have the same implementation level depending on methods/algorithms implementation and integration within an UMF compliant management system.

In order to structure and prepare refined deployment assessment, we introduce a set of criteria based on ISO/IEC 9126-1 which was extended to support autonomic computing characteristics.

In this document, we present a large amount of results answering to the problems identified by the Use Cases. Multiple network technologies and multiple self-management features are covered.

Depending on the solution implementation level, we make a first evaluation of our deployment assessment criteria. This evaluation has to be continued as our solutions are reaching higher implementation levels, based on UMF release 2.

As part of Task 4.2, integration works related to the integrated prototype are under progress and are considering the present deployment assessment to improve the final solutions based on UMF release 3.

Then the next release of deployment results will be based on refined solutions, closer to full implementation level.

# 5 References

[1] E. Xavier, C. Thierry, V. Guy, "Experiences in Benchmarking of Autonomic Systems", Autonomic Computing and Communications Systems, Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, Volume 23, 2010, p. 48.

[2] K. Chan and I. Poernomo, "Consistent Metric Usage: From Design to Deployment", Dependability metrics book, Pages 19-36, ISBN:3-540-68946-X 978-3-540-68946-1.

[3] Deliverable D3.5: Adaptation and fine tuning of Parameter Optimization Methods

[4] Deliverable D3.7: Adaptation of learning and operation methods to specific needs of future networks and services

[5] JADE (Java Agent DEvelopment Framework). http://jade.tilab.com

[6] John Strassner, Policy-Based Network Management: Solutions for the Next Generation (The Morgan Kaufmann Series in Networking), Morgan Kaufmann Publishers Inc., San Francisco, CA, 2003

[7] S. Davy, B. Jennings, and J. Strassner, "The Policy Continuum - A Formal Model," in Proc. of the 2nd IEEE International Workshop on Modelling Autonomic Communications Environments, MACE, pp. 65- 79, October 2007

[8] F.P. Kelly, " Charging and rate control for elastic traffic". European Transactions on Telecommunications 1997; 8:33–37

[9] F.P. Kelly, A. Maulloo, D. Tan, "Rate control in communication networks: shadow prices, proportional fairness and stability". Journal of the Operational Research Society 1998; 49:237–252

[10] M. Chiang, S. H. Low, A. R. Calderbank, J. C. Doyle, "Layering As Optimization Decomposition: A Mathematical Theory of Network Architectures", Proceedings of the IEEE, Vol. 95, No. 1. (05 January 2007), pp. 255-312, doi:10.1109/JPROC.2006.887322

[11] OVAL, Open Vulnerability and Assessment Language, Mitre Corp, http://oval.mitre.org/

[12] M , R. Badonnel, and O. Festor. Supporting Vulnerability Awareness in Autonomic Networks and Systems with OVAL. Proceedings of the ACM SIGCOMM / IFIP / IEEE International Conference on Network and Service Management (CNSM'11), October 2011

[13] re, R. Badonnel, and O. Festor. Towards Vulnerability Prevention in Autonomic Networks and Systems. Proceedings of the IFIP International Conference on Autonomous Infrastructure, Management, and Security (AIMS'11), PhD Workshop, June 2011

[14] Cfengine, Cfengine AS, http://www.cfengine.org/

[15] , R. Badonnel, and O. Festor. Supporting Vulnerability Awareness in Autonomic Networks and Systems with OVAL. Proceedings of the ACM SIGCOMM / IFIP / IEEE International Conference on Network and Service Management (CNSM'11), October 2011

[16] , R. Badonnel, and O. Festor. Ovalyzer: an OVAL to Cfengine Translator. PhD Student Demo Contest of the IFIP/IEEE Network Operations and Management Symposium (IFIP/IEEE NOMS'2012), April 2012

[17] RFC218, IETF, What makes for a successful protocol?, http://tools.ietf.org/html/rfc5218

[18] T. Kohonen, Self-Organizing Maps, Series in Information Sciences, 2nd ed., vol. 30, Springer, Heidelberg, 1997

[19] T. Kuremoto, T. Komoto, K. Kobayashi, and M. Obayashi, "Parameterless-Growing-SOM and Its Application to a Voice Instruction Learning System, Research Article," Journal of Robotics, vol. 2010, Article ID 307293, 9 pages, doi:10.1155/2010/307293

[20] CRAWDAD database, available at http://crawdad.cs.dartmouth.edu/index.php, last accessed: August 30th, 2012

[21] Dataset: dartmouth/campus (v. 2009-09-09), available at http://crawdad.cs.dartmouth.edu/meta.php?name=dartmouth/campus, last accessed: August 30th, 2012

[22] Trace ilesansfil/wifidog/session/04_07 (v. 2007-08-27), available at http://crawdad.cs.dartmouth.edu/meta.php?name=ilesansfil/wifidog#N1006B, last accessed August 30th, 2012

[23] P. Botella, X. Burgués, J. P. Carvallo, X. Franch, G. Grau, J. Marco, C. Quer , 'ISO/IEC 9126 in practice: what do we need to know?', In Software Measurement European Forum (SMEF 2004 )

[24] 3GPP TS 23.401: "General Packet Radio Service (GPRS) enhancements for Evolved Universal Terrestrial Radio Access Network (E-UTRAN) access".

[25] 3GGP LTE-Advanced official websitehttp://www.3gpp.org/LTE-Advanced/

[26] Evolved Universal Terrestrial Radio Access (E-UTRA) and Evolved Universal Terrestrial Radio Access Network (E-UTRAN); Overall description, 3GPP TS 36.300, V11.0.0 (2011)

[27] Long Term Evolution (LTE), A technical overview, Motorola, Tech. White Paper

[28] IF Akyildiz, DM Gutierrez-Estevez, EC Reyes, The evolution to 4G cellular systems: LTE-Advanced. Phys Commun. 3, 217–244 (2010). doi:10.1016/j.phycom.2010.08.001

[29] ETSI EG 202 237, "Methods for Testing and Specification (MTS); Internet Protocol Testing (IPT); Generic approach to interoperability testing"

# 6 Abbreviations

| | |
|---|---|
| 3GPP | 3rd Generation Partnership Project |
| 3GPP LTE | 3GPP Long Term Evolution |
| 3GPP SAE | 3GPP Service Architecture Evolution |
| AB | Autnomic Behaviour |
| 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 | Fibre 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 |
| 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 Organized 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 |

# Annex A: UMF functions in 3GPP-LTE systems

This section provides the mapping of the UMF system to 3GPP LTE network architecture [24][25][26]. Specifically, the main functions of the UMF core blocks have been examined and optimally distributed to the appropriate network elements as shown in Figure 45.

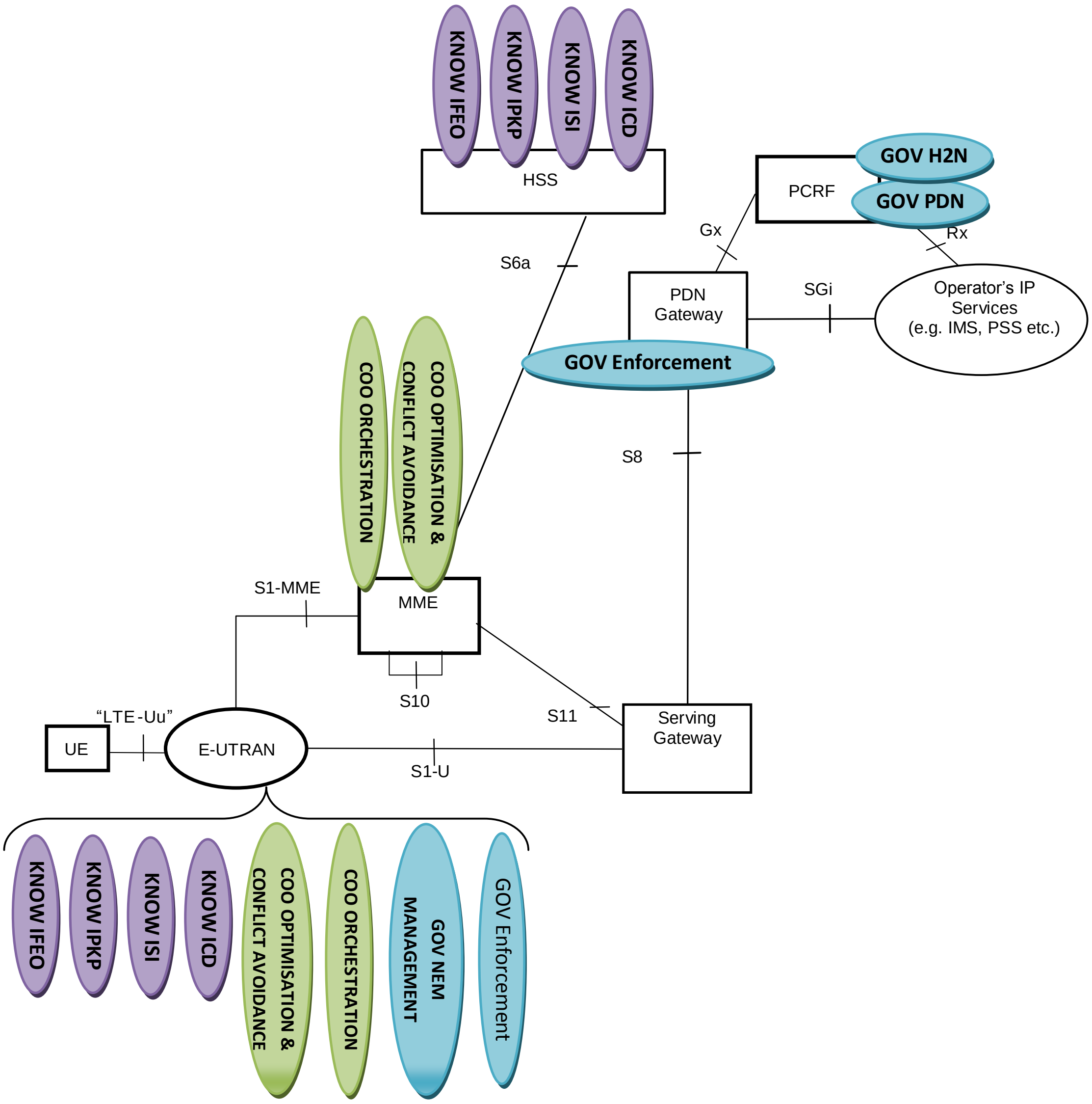


**Figure 45: Mapping of UMF functions on 3GPP-LTE network elements**

In detail, the GOVERNANCE Human to Network Interface and the Policy Derivation and Management (PDM) functions are mapped to Policy Control and Charging Rules PCRF function. This is justified since the PCRF function is responsible for a) policy control decision-making, b) controlling the flow-based charging functionalities in the Policy Control Enforcement Function (PCEF), which resides in the P-GW and c) providing the QoS authorization that decides how a certain data flow will be treated in the PCEF and ensures that this is in accordance with the user's subscription profile. The GOVERNANCE Enforcement function is mapped to the following elements: a) the Policy Control Enforcement Function (PCEF), which resides in the P-GW. The PCEF function is responsible for enforcing gating and QoS for individual IP flows on the behalf of the PCRF. To this end, the GOVERNANCE enforcement function is mapped to the PCEF for the application of the outcomes of the PDM function. b) the E-UTRAN, as regards the enforcement of the GOVERNANCE PDM functions in the e-NBs, the GOVERNANCE NEM Management function is mapped to the E-UTRAN. It should be noted that since there exists no direct link with E-UTRAN, evolutions in the respective interface and protocols are required [24][27][28].

Moreover, the KNOWLEDGE Information Collection & Dissemination – ICD, Information Storage and Indexing – ISI, Information Processing and Knowledge Production - IPKP and Information Flow Establishment and Optimization - IFEO functions are mapped to the following elements: a)  the Home Subscriber Server (HSS). The latter contains: 1) users' SAE subscription data such as the EPS-subscribed QoS profile and any access restrictions for roaming, 2) information about the PDNs to  which the user can connect, 3) dynamic information such as the identity of the MME to which the user is currently attached or registered and 4) authentication-related information as it may integrate the authentication center (AUC), which generates the vectors for authentication and security keys. To this end, the abovementioned KNOWLEDGE functions are mapped to HSS for the appropriate information collection and management related to users (specifically, subscription data, QoS profiles, access restrictions for roaming, information about the PDNs to which the user can connect, dynamic user attachment and authentication information).  b) the E-UTRAN, as regards the application of the information-related functions in the e-NBs [24][27][28].

Finally, the coordination block functions, namely Orchestration and Optimization and Conflict Avoidance are mapped to the following elements: a) the MME entity. The latter is responsible for many functions for managing mobile devices. It also controls establishment of EPS bearers in the selected gateways. Among other functions, it is in charge of managing security functions (authentication, authorization, NAS signalling), handling idle state mobility, roaming, and handovers. Therefore, the abovementioned COO functions are mapped to the MME for the orchestration and operation of control-plane operations related to mobility management and user authorisation. b) the E-UTRAN, as regards the coordination of the different functions residing in the e-NBs, e.g. the coordination between load balancing and coverage and capacity optimisation functionalities residing in the e-NBs and relay stations as in the respective NEM developed within UniverSelf [24][25].