Πανεπιστήμιο Πειραιώς
Τμήμα Ψηφιακών Συστημάτων

ΜΕΤΑΠΤΥΧΙΑΚΟ ΠΡΟΓΡΑΜΜΑ ΣΠΟΥΔΩΝ

«Διδακτικής της Τεχνολογίας & Ψηφιακών Συστημάτων»

ΜΕΤΑΠΤΥΧΙΑΚΗ ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΤΕΧΝΙΚΕΣ ΚΑΙ ΣΥΣΤΗΜΑΤΑ ΑΥΤΟΔΙΑΧΕΙΡΙΣΗΣ ΚΑΙ ΜΑΘΗΣΗΣ ΓΙΑ ΔΙΚΤΥΑ ΥΨΗΛΩΝ ΤΑΧΥΤΗΤΩΝ ΑΣΥΡΜΑΤΗΣ ΠΡΟΣΒΑΣΗΣ

Ακεζίδου Α. Μαρία
Πτυχιούχος και κάτοχος τίτλου σπουδών τμήματος
Ψηφιακών Συστημάτων

Πειραιάς 2011

University of Piraeus
Department of Digital Systems

M.Sc. COURSE

“Technology of Education and Digital Systems”

M.Sc Thesis

Techniques For Governing Autonomic and Self-Learning High Speed Networks

Akezidou A. Maria
B.Sc University of Piraeus,
department of Digital Systems

Piraeus 2011

Abstract

The concept of “Future Internet” (FI) is a globally emerging theme that currently operators are far from able to respond to. With this objective, cognitive/autonomic management of networks and services are seen as a viable direction for basing the management in the FI era. Governance of autonomic management is also proposed as management processes/systems of an operator on heterogeneous systems. Those processes are depending on the technology and on the vendor of the technology. The above will stress network operators to specify high-level policies should be transformed in low-level policies under specific mechanisms. Thus, a Human-to-Network (H2N) interface would enable operators to set business goals.

Keywords: future internet, network and service autonomic management, policies, business goal.

ΠΕΡΙΛΗΨΗ

Η έννοια του Διαδικτύου του Μέλλοντος είναι ένα παγκοσμία αναδυόμενο θέμα στο οποίο οι παρόν φορείς των τηλεπικοινωνιών δεν είναι σε θέση να ανταποκριθούν. Για το σκοπό, η γνωσιακή/ αυτόνομη διαχείριση των δικτύων και υπηρεσιών θεωρείται ως η πλέον πιο κατάλληλη κατεύθυνση για την διαχείριση της FI περιοχής. Διακυβέρνηση της αυτόνομης διαχείρισης προτείνεται επίσης ως οι διαδικασίες / συστήματα διαχείρισης ενός φορέα σε ετερογενή συστήματα. Αυτές οι διαδικασίες είναι ανάλογες της τεχνολογίας και για του χρήστη της τεχνολογίας. όλα τα παραπάνω πιέζουν τους φορείς των τηλεπικοινωνιών και δικτύων να καθορίσουν υψηλού επιπέδου πολιτικές, οι οποίες θα πρέπει στη συνέχει να μετατραπούν σε χαμηλότερου επιπέδου πολιτικές ακολουθώντας το πλαίσιο των συγκεκριμένων μηχανισμών. Έτσι, μία H2N (Human-to-Network) διεπαφή θα επιτρέψει στους διαχειριστές να ορίσουν νέους επιχειρησιακούς στόχους.

Λέξεις – Κλειδιά: διαδίκτυο του μέλλοντος, αυτόνομη διαχείριση δικτύων και υπηρεσιών, πολιτικές, επιχειρησιακοί στόχοι.

Dedicated to my parents Andrew and Marina

to my brother Joseph.

Στους γονείς μου, Ανδρέα και Μαρίνα

στον αδερφό μου, Ιωσήφ.

FOREWORD

Pursuing this M.Sc thesis was a demanding process, which required both attention and dedication. This effort, however, has been accompanied by the fact that I had the opportunity to expand my knowledge in the field of telecommunications networks and services. None of the above, though, would have been accomplished without God's help and the actual support of certain people that stood by me and therefore deserve special mention and my thanks.

Above all, my family that supported, endured me and being anxious with me in all phases of my life. My father, Andrew and my mother, Marina, who managed to follow me step by step in my life with great effort as loyal supporters, and taught me valuable principles and the way to be a proper human being that would strive every day with faith and patience. My brother, Joseph who became paradigm to me even from the first years of my life and showed me how to set goals and stay concentrated on those so that to succeed them. Hence, I initially owe to grant the largest share of the moral satisfaction for the outcome of this course and to deeply thank them...

My supervisor Associate Professor Panagiotis Demestichas has a considerably big part of this effort. First of all, I would like to express my deepest gratitude for demonstrating confidence in my face and skills even from the beginning of our cooperation and then for both moral and psychological support on technical and non issues. Mr. Demestichas was a real "teacher" for me and leader whose dedication and academic experience were invaluable. So I need to thank him and hope to be seen even more worthy of his expectations.

I feel also the need to thank the co-supervisors, Assistant Vera Stavroulaki and Kostas Tsagkaris with their patience and the very significant scientific guidance and advice in our cooperation.

Finally, I would like to mention to the people with who mutually share the pressure, anxieties and successes in a daily route. They are the members of *307 laboratory at Androutsou's street building* and more specifically, Yiouli Kritikou,Vangelis Thomatos, Aimilia Bantouna, Andreas Georgakopoulos, Panagiotis Vlacheas, George Athanasiou and Panagiotis Vasileiou. I thank them and wish to each one separately success in their lives.

Sincerely,

Maria A. Akezidou

ΠΡΟΛΟΓΟΣ

Η συγγραφή της παρούσας μεταπτυχιακής διπλωματικής εργασίας ήταν απαιτητική προσπάθεια που απαιτούσε προσοχή κι αφοσίωση. Η προσπάθεια αυτή, όμως συνοδεύτηκε με την γεγονός ότι μου δόθηκε η ευκαιρία να διευρύνω ακόμα τις γνώσεις στον τομέα των τηλεπικοινωνιακών δικτύων κι εφαρμογών. Ωστόσο, τίποτα από τα παραπάνω δε θα είχε επιτευχθεί χωρίς τη βοήθεια του Θεού και την έμπρακτη υποστήριξη των ανθρώπων που στάθηκαν δίπλα μου και γι' αυτό αξίζουν ειδική αναφορά και τις ευχαριστίες μου.

Υπεράνω όλων, η οικογένειά μου, ήταν εκεί να με στηρίζει, να με υπομένει και να αγωνιά μαζί μου σε όλες τις φάσεις της ζωής μου. Ο πατέρας μου, Ανδρέας και η μητέρα μου, Μαρίνα που με πολύ κόπο κατάφεραν να με ακολουθούν ως πιστοί συμπαραστάτες βήμα βήμα την ζωή μου διδάσκοντας μου πολύτιμες αρχές αλλά και το πως να γίνω ένας σωστός άνθρωπος που με πίστη κι υπομονή να αγωνίζεται καθημερινώς. Ο αδελφός μου, Ιωσήφ που από τα πρώτα χρόνια της ζωής μας αποτέλεσε το πρότυπο μου, και μου έδειξε το πως να θέτω στόχους και να μένω συγκεντρωμένη σε αυτούς ώστε να τους καταφέρνω. Σε αυτούς οφείλω αρχικά να χαρίσω το μεγαλύτερο μερίδιο της ηθικής ικανοποίησης για το αποτέλεσμα αυτής της πορείας και να τους πω ένα μεγάλο ευχαριστώ…

Σημαντικότατο μερίδιο αυτής της προσπάθειας έχει ο επιβλέπων καθηγητής μου, Αναπληρωτής Καθηγητής Παναγιώτης Δεμέστιχας. Θέλω να το εκφράσω τη βαθύτατη ευγνωμοσύνη που πρώτα από όλα επέδειξε εμπιστοσύνη προς το πρόσωπό και τις ικανότητες μου από τις αρχές της συνεργασίας μας καθώς επίσης ηθική και ψυχολογική υποστήριξη σε τεχνικά και μη θέματα. Ο κ. Δεμέστιχας υπήρξε για μένα πραγματικός «δάσκαλος» και καθοδηγητής του οποίου η αφοσίωση καθώς και ακαδημαϊκή του εμπειρία αποτέλεσαν ανεκτίμητη βοήθεια. Γι' αυτό νιώθω

την ανάγκη να τον ευχαριστήσω κι ελπίζω να φανώ ακόμα περισσότερο αντάξια των προσδοκιών του.

Αισθάνομαι επίσης την ανάγκη να ευχαριστήσω τους συνεπιβλέποντες μου, Επίκουρο Βέρα Σταυρουλάκη και Κώστα Τσαγκάρη για την υπομονή τους αλλά και την ιδιαίτερα σημαντική επιστημονική καθοδήγηση και συμβουλές τους στην συνεργασία μας.

Τέλος, θα ήθελα να αναφερθώ στους ανθρώπους που καθημερινά αλληλοϋποστηριζόμαστε στις πιέσεις, τις αγωνίες και τις επιτυχίες. Αυτοί είναι μέλη του εργαστηρίου *307 του κτηρίου της οδού Ανδρούτσου* και πιο συγκεκριμένα οι Γιούλη Κρητικού, Βαγγέλης Θωμάτος, Αιμιλία Μπαντούνα, Ανδρέας Γεωργακόπουλος, Παναγιώτης Βλαχέας, Γιώργος Αθανασίου, και Παναγιώτης Βασιλείου. Τους ευχαριστώ πολύ και τους εύχομαι στον καθένα ξεχωριστά κάθε επιτυχία στη ζωή τους.

Με τιμή,

Μαρία Α. Ακεζίδου

TABLE OF CONTENTS

LIST OF FIGURES

1. INTRODUCTION

The last decade has seen considerable developments in network devices, communications protocols and services. This continuous evolution of communications networks has led to complex, dynamic and heterogeneous networks that are evolving into a global service-rich communication infrastructure. Data and telecommunications networks and the services they offer have been growing enormously in size, which with the addition of the above mentioned characteristics make them difficult, if not impossible; to be effectively managed and operate within desired bounds with the use of traditional approaches that rely on human monitoring and intervention.

In addition the evolution of Future Internet (FI) ([2], [3], [4]) introduces a free exchange of information between enterprise applications and networking systems which promotes the personalization of services and enables many different types of end-user applications and management operations optimizing the network performance. As result of this free information exchange, it is needed the facilitation of the federation of information between these applications, harmonization of the differences between operation, management of data and information models in heterogeneous networks, and introduction of application management systems.

On the other hand, operators will have additional complexity to address, compared to today, in the management of networks and services. The reasons can be summarized in the following main points:

- The support of a very wide set of applications, as the Internet is part of every facet of our lives;

- Powerful network infrastructures, which offer several options for efficiently providing the applications;
- Evolving as well as new business model aspects.

In this context, researchers and practitioners are pursuing the vision of autonomic network management, which seems to be the most appropriate in this situation by introducing the capability of network entities to self-govern their behaviour within the constraints of business goals that the network as a whole seeks to achieve.

Although, the network operators' services of today span technological boundaries between the wireless and wireline domains, they need also to be (self)-managed just like the systems supporting them. In this sense, it should be built a more service-centric view and an "Everything as a managed Service" paradigm so that to be accelerated the convergence of fixed and mobile.

Cognitive/autonomic management systems ([5]-[10], addressed also in WG6 (working group 6) of [11]), are seen as a viable direction for basing the management in the FI era. Evidence of the appropriateness is provided by the promising application of cognitive systems in the management of the complex wireless world ([5]). The promising results have lead also to important standardization initiatives ([12], [13]).

The key factor of the success lies on the materialization of a Unified Management Framework (UMF) for the different existing and emerging architectures like IMS (IP Multimedia Subsystem), or for architectures designed in past autonomic research. This unified and evolvable framework will constitute a cross-technology (wireless and wireline), common substrate for both systems and services – advantageously filling the technology and standard gaps. This would capitalize both on research done in autonomic networking and demonstrate its applicability to industry standards.

This framework will first ensure that multiple diverse management systems implemented upon different autonomic architectures will be able to interoperate and federate. Secondly, it will also guarantee that autonomic functions may be implemented (apart from optional interfacing) independently of the architecture chosen for the management system. Finally, it will ensure that common processes may be used across all the technological domains of an operator for enabling functions like service establishment, adaptation, fault management, and optimization of resources in response to changing contexts and in accordance with applicable business goals and governance policies of the separate domains.

The Unified Management Framework will be realized by pursuing a dual approach and continuously ensuring the complementarity of these two axes:

- a bottom-up axis, in which a scenario based approach, considering services and systems, wireline and wireless technologies, will be used to design a system that aims at resolving operators' day-to-day problems identified in live networks and on existing service/network architectures.
- a top-down axis, which will capitalize previous autonomic architecture research to achieve a coherent set of autonomic network management functionalities that can interwork in a scalable manner.

Autonomic systems are cantered on closed feedback control loop. Only specific implementations of this loop at the proper level enable a system to dynamically adapt to changing situations.

This objective will have a strand that is between the operator and the systems, providing the former with enough knowledge such that it will always control the latter, and will be able to take over the autonomous functions (i.e., taking the operator outside the decision loop but not fully outside of the control loop). In a sense, this can be seen as off-line confidence (or vertical trust), allowing the deployment of systems.

A valuable functionality of this framework is the assessment of a governance method for guiding infrastructure behaviours and offering a service view. This empowerment of network operators with governance capabilities is one of the requirements for the successful deployment of autonomic management systems. This means to enable network operators to introduce objectives that should be pursued, constraints that should be respected, and processes and rules that should be followed. In principle this is achieved through policies ([14]).

The focus of this thesis is in this direction. Governance would perform as a service role through the formulation and implementation of policy. Particularly, the main idea is that the operator is triggered to define business goals/policies, in high-level terms. Policies are derived according to the higher level goals, to provide constraints and priorities and then are assessed against existing goals/policies so as to identify and resolve conflicts (in fact, conflicts can arise if the defined goal/objective/policy are antagonist with respect to previous goals or the impact of these goals on already deployed applications).

In legacy systems, there is not such human-to-network interface that will be used to introduce the business level goals in high level terms and leave the system to autonomously work out the situation and meet the objectives. The implementation of this interface, which functionality is characterized as governance of autonomic management, is part of this thesis too.

In more detail, the thesis is structured as follows. A brief analysis of Future Internet and its characteristics is introduced in section 2 while the presentation of autonomic management of it can be found in 3rd section. Furthermore, section 4 has a conceptual presentation of the governance of an autonomic management in order to drive us at the analysis of human-to-network (H2N) governance tool in the 5th section. Thus, this thesis concludes in section 6 with a summary of main points along with directions for future extension of the work presented. Last but not least, at the end of this thesis two appendix can be found, APPENDIX I: and APPENDIX II: , which include a presentation of some representatives codes that were used in the implementation of the tool and a brief summary of this thesis in Greek, respectively.

2. FUTURE INTERNET

2.1. Introduction

Future Internet (FI) has become the federating theme for European research on communication networks and services. At the core lies research on communication networks toward an efficient, scalable, and reliable Future Internet coupled with research on the underlying technologies, in particular mobile and wireless access and optical networks.

This focus on Future Internet is motivated by the changing requirements to the current Internet that was designed in the 1970s to support communication between computing systems for communities of expert users. It was not designed to cope with the wide variety and ever growing number of networked and mobile users and applications, business models, edge devices, networks, and environments that it now has to support. Its structural limitations are increasingly being recognized worldwide.

In the Future Internet, access to the network will be made available ubiquitously, and connectivity will become a fundamental service that communities use and rely on. Since the current Internet has grown to become so large, it will be commercially and operationally very challenging to introduce new architectural principles [20]. The FI shall provide a twofold path: enabling focus on a specific user need and developing a solution without paying attention to the network infrastructure. Once a leader on a specific application emerges, the FI should also enable the growth of an ecosystem of players able to build their own applications on top of its system.

In sequel, it is presented the characteristics of current internet in contrast with the specifications and changes that would arise with Future Internet.

2.2. Technical limitations of the current Internet

Internet technology can be characterized by its design principles: layering, packet switching, a network of collaborating networks, intelligent end systems, as well as the end-to-end argument ([18], [19]).

It has been simple to link any new network to the Internet, providing instant benefits resulting from the interconnectivity with a huge range of communicating peers. The transparency of the Internet has facilitated the deployment of successively more complex network-agnostic applications and services. However, with its increasing success, the Internet architecture is progressively losing its original simplicity and transparency. Some of the main causes are the rise of new classes of applications, additional operational and management requirements, variety of business models, security mechanisms, and scalability enablers that give rise to ad-hoc solutions that extend the architecture without regard to the original key design principles.

Examples are firewalls to support end-user and site security, and Network Address Translation (NAT) to cope with the exhaustion of IPv4 address space. There is, however, a growing consensus among the scientific and technical community that the methodology of continuously patching the Internet technology will not be able to sustain its continuing growth at an acceptable cost speed.

This loss of flexibility is already being felt as the number of Internet nodes grows another order of magnitude. The size and scope of today's Internet make the deployment of new network technologies very difficult while experiencing increasing demand in terms of connectivity and capacity. Examples are the slow deployment of technologies such as multicast or Internet Protocol version 6 (IPv6). Innovation has happened mainly in the applications and underlying transmission technologies, rather than in the core technology, the network and transport layers TCP/IP. The following technological limitations have to be overcome to meet the future challenges for the global communication network [17].

- Limited Support of Mobile Wireless terminals:

While the Internet was designed for stationary computers, today laptops and smart phones are constantly on the move. With today's technology, a laptop changes its address and reconnects as it moves from one wireless network or access point to another, disrupting the data flow. Alternatively, the Internet standard Mobile IP allows routing all traffic back to the first access point as a laptop moves to a second or third location, but delays and inefficiencies may result. As a clean slate solution, the address system would have to be restructured so that addresses are based more on the device and less on the location. This way, a laptop could retain its address even if it moves from one wireless network or access point to the other.

- Lack of built-in security:

With the evolution toward Internet-based services, traditional telecom networks as backup for mission-critical services are expected to gradually disappear. Therefore, built-in security mechanisms are one of the main goals of future Internet design.

The Internet was designed to be open and flexible, and all users were assumed to be trustworthy. Thus, the Internet protocols were not designed to authenticate users and their data, allowing spammers and hackers to easily cover their tracks by attaching fake return addresses onto data packets. Internet applications such as firewalls and spam filters attempt to control security threats. But because such techniques do not penetrate deep into the network, bad data still get passed along, clogging systems and possibly fooling the filtering technology.

The network would have to be redesigned to be sceptical of all users and data packets from the start. Data would not be passed along unless the packets are authenticated. Faster computers today should be able to handle the additional processing required within the network.

- Scalability issues:

The current Internet has dealt with scalability by using a hierarchical architecture separating the different issues of routing in three different levels. The lowest level deals with local connectivity and configuration of interfaces IP addresses link layer mechanisms (essentially through Ethernet). The second level introduces IP routing between subnets by assuming that the local connectivity is provided inside a network mask. The third level implements operators' policies through BGP filtering and announcement rules, assuming that an AS operator is wise enough to provide optimal connectivity inside itself.

Just when the Internet is becoming critical infrastructure, the core protocols may become increasingly fragile as more manual configuration is needed to avoid cascading problems due to overload, accidental misconfiguration, or attack. The IPv6 standard allows expanding the address pool, but nearly a decade after most of the standard was completed, the vast majority of software and hardware still use the older IPv4 technology. Even if more migrate to IPv6, not all addressing issues would be solved. Researchers are questioning whether all devices truly need addresses. Sensors in a home could communicate locally and relay the most important data through a gateway bearing an address. As routing and addressing are becoming the main challenge of a global network with billions of users and objects, scalability is one of the most critical design criteria for future Internet architectures.

- Performance and Quality of Service challenges:

While mechanisms for providing quality of service (QoS) within the Internet as well as asynchronous transfer mode (ATM) networks have been very well studied, the interaction problems between the network layers are still unresolved, and the management of such services, including configuration, policy setup, charging, and interprovider setups, is still a challenge [18].

2.3. Specifications and changes because of FI

The Internet has emerged as a critical infrastructure for society and the economy as a whole, similar to any other utility (e.g., infrastructure for electricity and water supply). Society is undergoing a paradigm shift, the evolution of the society and the Internet now being tightly interconnected. Daily life factors including health, transport, knowledge, and culture rely increasingly on the Internet in the developed world, and it is bringing economic development to emerging economies.

Since the Internet was designed for fixed terminals, it shows inefficient behaviour for mobile and nomadic terminals. Therefore, generic and efficient support of mobile terminals and mobile applications is one of the major design goals of future Internet architectures and technologies.

2.3.1. Application requirements for FI

Besides the increase in number of users and connected devices expected, new application requirements for the future Internet are emerging.

- High-Quality and shared content dissemination:

As digitalization of data progresses, it is now expected that the majority of new media will arrive in digital form, with the analog form being the exception. For instance, digital videos with better resolution, 3D videos, virtual reality, and gaming not only made their appearance but also they made clear that the their size will be constantly increase which means that they progressively penetrate the Internet space.

Not only high-quality content is made available over the Internet by large content providers. Users are also enabled to easily produce, offer, share, and consume content on the Internet, and are becoming *prosumers*. Whereas communication will remain an important service to be supported by the global network, dissemination of content, either distributed by content providers or made available by prosumers, is expected to be one of the main functions of the future Internet.

- Connection objects and things:

While the current Internet is a collection of rather uniform devices, it is expected that the Internet of Things will be characterized by a much higher level of heterogeneity, as objects totally different in terms of functionality, technology, and application fields are expected to belong to the same communication environment. The Internet of Things can be defined as "a worldwide network of uniquely addressable and interconnected objects, based on standard communication protocols." This enables applications involving real-world objects, but also business applications based on network-assisted machine to machine interaction.

- Service- Oriented internet:

Whereas a lot of computing and storage applications are still executed locally on end-user devices such as PCs, a service-oriented Internet would allow access to complex physical computing resources, data, or software functionality in the form of services. One example is the cloud computing approach to infrastructure services, where large-scale data centres provide virtual execution and storage environments as Internet services with the same functionalities as physical machines but far greater flexibility and scalability.

These sets of expected applications for the future Internet are only examples; the target architecture should generically and flexibly support different requirements and traffic patterns.

2.3.2. FI dealing with current major shortcomings

The current Internet was originally designed for “fixed” terminals and can hardly support mobility. It is necessary to develop new mobility management schemes for the future Internet.

- Flexibility:

In the current business mode, Internet Service Providers (ISPs) are responsible for both maintaining the Internet substrate and providing Internet services. Most ISPs are unable to manage both availability of applications and increasing number of users at the same time. On one hand, they have inadequate impetus to upgrade their physical networks because there are still new subscribers. With the availability of their own physical network resources and subscribers, most ISPs have never been well-driven to provide flexible and diverse Internet services and applications. During the recent years, a creative approach named network virtualization has been widely proposed to deal with the problem of Internet inflexibility. This business model separates effectively the aforementioned two functions of an ISP, decoupling traditional ISPs into VNPs (Virtual Network Provider) and PIPs (Physical Infrastructure Provider) [21]. VNPs are responsible for providing Internet users with flexible and diverse services through subscribing physical resources from PIPs, which will be exclusively required to maintain the health of physical networks.

- Security

Security has always been a significant problem concerning “netizens” along the history of the Internet. The classical paper composed by David Clark, The Design Philosophy of the DARPA Internet Protocol [22], even had no words about network security, indicating that security was absent in the mind of the inventors or designers of the original Internet. Nowadays, every single day is witnessing numerous cyberspace attacks including Distributed Denial of Service (DDoS) ones sponsored by hackers. A new Internet with ensured security from architectural foundation is increasingly needed by a large number of secure-aware users. Security as an outstanding problem must to be supported by the Future Internet in clean-slate approaches.

- Mobility

The current Internet was originally designed exclusively for fixed network nodes which were the sole entities to be connected at that time. However, the situation has no longer been true since the invention of mobile devices that have access to the Internet, such as laptops, handheld computers and smart phones. The rapid spread of these mobile network nodes has made it clear the lack of mobility support from the Internet itself. The current architecture including the IP layer can not support the well being of those mobile nodes from ground-up. Instead, there have been workarounds used to bridge this gap temporarily, like mobile IP and geographic routing and so on.

It is necessary to point out that these workarounds are solely focused on the IP layer. Future Internet researches on mobility are aimed to support current and new mobile network devices completely, by conceiving innovative designs in an architectural way.

- Manageability

The current Internet architecture is nearly invisible to end users. Even network administrators have considerably limited privilege to maintain network health. The inadequate support for network manageability always makes it difficult for network administrators as well as users to efficiently and timely deal with any inexplicable periods of degraded performance or outright non-function. Internet manageability is very closely associated with its performance reliability, which in turn determines the possibility to deploy various high-end network applications. Current solutions to the lack of network manageability and control are actually ad-hoc. For instance, inter-domain and inner-domain routing algorithms are used to compute routes. Packet filters and tunnels are utilized to realize access control and virtual private networks. And manual configuration is widely used to carry out network management. Similar to other requirements aforementioned, manageability is to be better supported by Future Internet from ground-up.

2.3.3. Requirements of wireless networking

The FI is expected to embrace billions or even trillions of wireless devices, ranging from tiny sensors to powerful mobile phones and even novel form factors not yet invented. As a result, a wireless Internet of unprecedented scale is emerging. In the quest for the Future Internet, certain requirements must be considered, taking in mind wireless networking. To this end, we highlight four requirements, deemed as important for future wireless networks [15]:

1. Address scalability, catering for unprecedented numbers of mobile devices and extreme densities. Nowadays it is reasonable to expect several wireless devices carried by the same person, while several persons gathered in the same area create extremely dense wireless deployments.
2. Counter complexity and heterogeneity. By providing abstractions and encapsulating implementation details, the gap between high-level management goals and low-level processes can be bridged. Technology independence of management solutions as well as

standards conformance are increasingly important to maintain interoperability.

3. Offer flexibility to adapt to change. Changes increasingly happen to the business models implemented by networks. Fast and efficient adaptation of a network to changing business policies is critical to maintain profitability for owners and improved experience to end-users.

4. Maintain a user-centric character. According to ISOC (Internet Society) [16], this very characteristic combined with the openness of Internet standards, have fuelled user creativity and innovation, offering us today the Internet as we know it. These principles should also be taken on board, to cater for a user-centric Future Internet.

3. AUTONOMIC MANAGEMENT OF FUTURE INTERNET

3.1. Introduction

A key challenge of FI is to provide means that will enable cognitive network management through dynamic, ad-hoc and optimized resource allocation and control, fault tolerance and robustness associated with real-time trouble shooting capabilities.

The introduced management functionality should enable each network of a business domain in providing a *service offer*. These offers should lead to the provision of the best possible QoS level, among those allowed. QoS provision should be done in the most efficient way, in terms of technology and resource utilization.

Rising to this challenge, the vision of self-managing systems has been adopted for FI [23]. Under the umbrella self-management concept, self-awareness, self-configuration, self-protection, self-healing and self-optimization stand as prominent properties [24] of FI systems. The objective is that as business applications and processes that span organizations have become more prevalent, problems with the management of such applications and processes across multiple management domains using heterogeneous management technologies have become more apparent.

Required or desired behaviour of systems and applications can be expressed in terms of management policies. Such policies can in turn be used to express expected operational characteristics of these systems and possible management actions. The effective use of policies in autonomic management requires that the policies be captured and translated into actions within the autonomic system.

Through autonomic management we seek to employ techniques to enable the efficient operation and utilization of systems and services with little or no human intervention. Such autonomic management systems are expected to operate within a set of expected constraints or conditions, i.e., operational or behavioural requirements. The requirements may be defined by either system administrators or defined as part of service agreements.

3.2. Autonomic and cognition-based operations of FI

- Self-management:

It naturally appears as one the most important issues for FI's design; in the actual approach, the term implies a more generalized context, incorporating all potential autonomic and cognition-based operations in a "system". Some distinct methods of self-management are identified, as follows [25]:

- Self-awareness:

It stands for the knowledge building process as a non-stop necessity in various self-management systems. It is a continuous process where awareness is perceived as conclusions derived by the system on being present or status related to a particular operational state, at relevant time frame.

- Self-optimization:

It is system's ability to execute modifications of its operations for achieving the targeted "optimality point" in terms of the relevant performance metrics for any specified event. Here "optimality point" is considered as a "broader" term, including a variety of parameters all subject to particular types of events in the system, evaluation criteria applied. Thus, self-optimization can relate to maintaining operational conditions in the system and/or improving and/or maximising them.

It is also the systems' automatically ability to improve its performance and efficiency (i.e., to maximize resource allocation and utilization to meet user's requirements and needs with minimum intervention).

- Self-configuration:

It is the system's ability to dynamically configure itself following high-level policies, while the rest of the system adjusts automatically and seamlessly with minimum intervention in response to changes in the network behaviour or deployment of new network devices.

It is also system's ability to accommodate new operational aspects in terms of the network elements (NEs), hardware, software, functional improvements and/or services/facilities that have been provisioned either by the system itself or by the network operator. The essence is in the capability to "add"/"accommodate" new functional components.

- Self-healing:

It is the ability of system to automatically detect problematic operations, diagnose and repair the localized failures (software or hardware) by taking the initiatives. This will improve the networks' resiliency.

It is also the ability to "respond" to any unplanned/unpredicted events (i.e. failures) requiring corrective actions and, therefore, to restore or to improve its operational aspects. This method is quite diverse in terms of the targeted operational aspects related to the degree of the "healing" required.

- Self-protection:

It is the ability of system to automatically take autonomous actions against malicious attacks or cascading failures to protect itself from unauthorized accesses, viruses and general failures. This is in using the gathered knowledge for deducing the events in advance to their occurrence and, hence, for selecting proper directing operations by following similar “knowledge patterns”.

More specifically, autonomic systems have the ability to monitor, detect, diagnose, and heal themselves by taking autonomous actions to insert and remove code(s) at the run-time.

- Self-organisation:

It is a method demonstrating ways of collaborations of NEs (or elements clusters) in the context of specific management functions; it is built on specific management principles, i.e.: absence of centralized control, continual adaptation to a changing environment, autonomous interacting elements, simple rules for autonomic entities local interaction and interactions based on local knowledge.

3.3. The policies in autonomic management of FI

Autonomic management in FI strives to ensure that systems and applications operate as expected. Policies can be used to express required or desired behaviour of systems and applications and so can be used to specify to an autonomic management system the expected operational characteristics of these systems. Such policies can be captured and mapped into autonomic management actions, providing the kinds of directives on which an autonomic management system can rely in order to meet requirements.

Policy-based management is widely seen as an appropriate management paradigm to facilitate higher level, human-specified cognitive decision making; therefore, many researchers are examining how policy-based management can be leveraged to help realize the autonomic vision.

This requires numerous advances in the state of the art, chiefly with regard to:

- Management of heterogeneous functionality:

One of the problems in applying autonomic principles to networks is that networks are made up of many different devices. These devices have different programming models and provide different management data, describing the same or similar concepts. This makes it imperative to harness information models and ontologies to abstract away vendor-specific functionality to facilitate a standard way of reconfiguring that functionality. Achieving this will enable legacy resources with no inherent autonomic capabilities to be managed by autonomic systems.

- Adaptability:

One of the promises of autonomic operation is the capability to adapt the functionality of the system in response to changes in user requirements, business rules, and/or environmental conditions. This requires a more flexible governance approach than is provided to date. In particular, the system must sense context changes and use policies specific to the new context to effect the required re-configuration of network devices.

- Application of learning and reasoning techniques to support intelligent interaction:

Current examples of network device configuration and management rely on vendor-specific snapshots of static data. For example, statistics can be gathered and analyzed to determine if a given device interface is experiencing congestion. However, existing management data does not tell the user why congestion is occurring. This information must be inferred using these and other data and retained for future reference. Hence, there is a need to incorporate sophisticated, state-of-the-art learning and reasoning algorithms into autonomic network management systems.

3.4. Human intervention in management of FI

The central problem in FI management is the critical human intervention that is time-consuming, expensive, and error-prone. The autonomic computing initiative ([26], [27]) (launched by IBM) aims at addressing this problem by reducing the human role in the system management process. This initiative proposes a self-management approach that can be described as developing the management capabilities of the autonomic systems to anticipate requirements and to resolve problems with minimum human assistance. To do so, the human role should be limited; he only has to describe the high-level requirements and to delegate control tasks that the system performs in order to achieve these requirements. This approach is generally called an autonomic or a goal-oriented management approach [28]. This approach starts with the specification of high-level requirements. Then, these requirements are analyzed, refined, composed and formulated into policies implemented in the target system specific technologies [28]. These operations present the most important tasks in the specification and the implementation of the autonomic management approach. They have recently been the focus of several research studies interested in developing analytical methods of refining high-level goals into low-level policies ([29], [30]).

3.5. Design issues of the implementation of autonomic management of FI

The implementation of the autonomic management of FI is a value chain depicting how business goals correlate to IT infrastructure. As a result typical problems found in rigid solution implementations, include the following.

- Flexibility:

Both providers and consumers of IT infrastructure are frustrated by the complex and time-consuming process of deploying infrastructure. Once resources are in place, it is difficult to implement changes to them by their very nature.

- Coordination:

The traditional approach of designing, procurement, deployment, and management activities is not holistic; instead, the tasks are separated into silos of control, making coordination difficult.

- Resource Provisioning:

The amount of resources needed to supply a service is not fixed but rather demand characteristics vary over time. This means the service spends a large majority of its existence either over- or under provisioned due to the difficulty of adjusting the scaling resources to meet demand.

- Virtualization Scope:

The introduction of solutions utilizing virtualization and varied dynamic technologies are treated in the context of a single (or few) application(s); therefore, only a portion of their value is captured.

- Peak Load Capacity:

The team usually designs the hardware to process two to five times the current anticipated peak load, leaving the resources significantly underutilized at most times. But because of inherently fixed configurations, no other application or service can share that unused capacity.

- Local Optimization:

Deployment and management fall under the control of operations and/or engineering where configurations and physical placement are arranged by hardware asset type (e.g., database servers have a section of the data-centres, application servers are placed in a different section).

- Dispersed Resources:

Application growth may force new dedicated servers to be dispersed across the data-centre, which also slows response times and introduces additional latency.

- Collaboration:

Highly collaborative applications and application tiers are rarely placed in close enough proximity, which increases latency, affecting response times and overall performance.

- Physical Movement:

Movement of hardware resources around data-centres is currently unacceptable due to a lack of instrumentation to gauge and report on impacts, not to mention the potential for interruptions of service.

- Managing Peaks:

Applying needed resources to mitigate processing peaks is difficult, time-consuming, and requires new asset procurement.

- Workflow:

Existing processes and procedure are manual and do not consider management/administration of new technologies; workflow automation is nonexistent, which is key to productivity improvements.

- Staff Overhead:

Design and procurement are based on individual application need and are typically marshalled through an organization by the application team.

- Tools:

Metrics and monitoring tools are rarely robust enough to report, anticipate, or react to change; therefore, the consequences of spikes in demand are rarely correlated across the entire application.

3.5.1. Introduction of Real Time Infrastructure

Real Time Infrastructure (RTI) may be introduced as an approach for autonomic management of FI. This will give IT the grounding required to become a strategic enabler of the business goals and become more agile with buying particular sets of technology.

The purpose of RTI is to implement fluidity of resources to meet demand in real time while enhancing the quality of experience (QoE), enabling transparency of processing so latency can be understood for every component of an application and facilitating faster deployment time, providing a better competitive edge. This requires autonomic management capabilities that predict problems, enable self-healing in case of failure of a system, a global dashboard view that relates business transactions to its effect on infrastructure supply, and guaranteed execution of designated transactions based on business policy.

The key operating characteristics of RTI are as follows:

- Dynamic allocation of IT supply to real-time demand based on service contract requirements, including guaranteed execution response, throughput volume, work characteristics, and time constraints or cost/margin rules. Dynamic allocation is further augmented through workflow automation that extends the capability of policy-based alignment of resources.
- Real-time transaction work management that synchronizes work characteristics (demand) with resource capabilities (supply) to meet execution management service contract requirements.

- Self-healing attributes, such that failures are seamlessly repaired without human intervention.
- Efficient execution of work on infrastructure that is tailored to the type of processing needed.

3.5.2. RTI goals

The essence of autonomic management, and thus autonomic communication in FI, is to enable the autonomic network (and the systems and component that make up the network) to adapt the services and resources that they offer in accordance with changing user needs, environmental conditions, and business goals, all the while ensuring that context-sensitive business goals are protected. RTI and businesses goals must work together and think of IT as an investment. This requires a shift in process, people skills, and technology usage.

Business goals combined with IT goals can define the goals of RTI's design.

- *Reduced Time to Market:* IT becomes agile enough to add new services and products, reducing lead time from months or weeks to days.
- *Business Capacity:* IT can increase its processing capability as the business grows, handling unexpected spikes in transactional demand in real time.
- *Budget Predictability:* IT can show that its costs are predictable and proportional to business growth.
- *Competitive Advantage:* When fully realized, RTI represents a significant step toward IT becoming a strategic asset for the business instead of just another cost centre. IT becomes a strategic enabler for the business.

- *Business Assurance:* IT can demonstrate SLA performance guarantees through consistent availability of resources, regardless of business demands.
- *Supply Efficiency:* RTI mandates a fluid set of infrastructure services that make resources available to meet oncoming demand at the level that is needed, when it is needed.
- *Utilization Efficiency:* RTI assumes demand fluctuates for different applications at different times.
- *Reuse:* RTI respects and incorporates physical limitations where the operational characteristics of performance, security, and availability are compared across the constituent patterns of applications to determine similarity.
- *Performance through Localization:* RTI principles include consideration for proximity when assembling an execution environment for collaborating applications (i.e., taking latency and response time into account).
- *Performance through Optimization:* RTI designates four fundamental operating environments (infrastructure patterns) that facilitate specific operational behaviours. Once analyzed, any application/service will be able to operate in one of those four environments.
- *Quality of Service:* RTI provides a tactical capability to utilize resources based on business polices that balance: user experience, efficiency, and cost.
- *Improved Processes:* RTI tactics affect how systems and infrastructure are designed, managed, serviced, and utilized.

- *Organizational Roles:* RTI tactics positively affect how IT organizations are structured in terms of roles and responsibilities, providing more agile responses to new needs and problem resolution and determination.
- *Service and Change Management:* Dynamic models of operation require processes and procedures of service, delivery, and change management to accommodate the "on-the-fly" and "as needed, when needed" operations of IT.

3.5.3. RTI business benefits

In summary, these are some of the more readily obtainable benefits of RTI:

- Become more flexible and responsive to the dynamic needs of the business.
- Simplify and reduce the complexity of the IT environment.
- Get more value out of project and operational IT spending (both systems and people).
- Reduce costs while delivering improved service.
- Eliminate dedicated silos of data, systems, and infrastructure as they exist today.
- Reduce the time it takes to build and deploy new business services.
- Implement and sustain predictable qualities of service.

4. GOVERNANCE OF AUTONOMIC MANAGEMENT

4.1. Introduction

To make networks autonomic, it is important to identify the high-level requirements of the administrators. Autonomic infrastructure implies a quantum step on network operations automation and intelligence that requires human technicians to go further from the command and control paradigms.

Governance has appeared as the term to describe the new way (techniques and functionalities) for managing autonomous behaviours instead of being based on the stovepipe type of traditional network management.

Network governance is almost always interwoven with policies lying at the highest level of the so called policy continuum. Typically, business level policies are defined in this level i.e. high level expression of business objectives. In the sequel and according to the policy continuum concept these policies are propagated to the network going through an arbitrary set of levels (related to different aspects of the management of a communications network) where they are being transformed into lower level policies, until they finally reach the element(s) in which to be enforced in terms of low level, technology-specific commands.

This chapter is the main work that carried out in this thesis. It starts with a definition of the Network Governance as part of the UMF, followed by the most important challenges this task must face in order to successfully accomplish its mission. Section 4.3 summarizes technological challenges that the building of network governance framework faces.

In contrast, Section 4.4 presents a summary of the current state of the art of the governance topic and of the different models and mechanisms that could be of utility when building a governance framework. Section 4.5 presents the approach that is followed in order to achieve the above mentioned goals.

4.2. Definition of Network Governance

Although one of the goals of Autonomic Infrastructure is that of self-management, a framework aiming to manage an autonomic network must include tools to facilitate the control and supervision of the network. It is Operators who need to lead the business transformation and it is a must to ensure human to network communication, if they are to control the infrastructure, focusing on the business level rather than technical aspects of the network - which ought to self-manage thanks to autonomics. Policies seem to take new relevance on this scenario. The perception brought to the operator by this paradigm shift is that of keeping focused on network governance while network management goes autonomic.

Governance is a high level mechanism which involves all functionalities necessary to address the gap between high-level specification of human operators' objectives and existing resource management infrastructures towards the achievement of global goals. Governance also encompasses Human-to-Network (H2N) communication and the introduction of policies and business goals to the network. It should be underlined that orchestration features are required so as to coordinate various network management entities towards achieving global goals (Control of control loops).

Moreover, the term of "Governance" can be found in several definitions, for instance:

- Governance is a decision making process that comprises the distribution and coordination of lower level decision making processes towards the achievement of global goals. Network and Services Governance is a comprehensive framework of structures, processes and relational mechanisms that addresses the gap between high-level specification of client performance objectives and existing resource management infrastructures. Structures involve the existence of responsible functions (i.e. human operators). Processes refer to decision making and monitoring. Relational mechanisms include business participation and partnerships.
- Governance is the set of authorities, processes, and procedures guiding strategic and key operational decisions made for the enter-prise that is state government. It clarifies relationships and responsibilities among the entities making up the enter-prise.
- Network governance acts on a self-managed behaviours expressing high-level business goals which represent owner and user performance objectives and low level primitives and finally address resources configuration.
- Governance is appeared as the term to describe the new way (techniques and functionalities) for managing autonomous behaviours instead of being based on the stovepipe type of traditional network management.

4.3. Challenges of Network Governance

This section addresses the challenges for network governance.

An increasing number of heterogeneous devices used from different places to access a myriad of very different services and/or applications require a new reliable, dynamic, and secured communication infrastructure with highly distributed capabilities [31]. The autonomic network envisions meeting these features. The complexity of managing such infrastructure exceeds the capabilities of current Operational Support Systems (OSS) and is one of the main challenges that the telecommunications industry is currently facing. Operators need to change their vision on current management paradigms; otherwise they will collapse under the operational weight of managing complexity [32].

The new network infrastructure is highly adaptive and autonomous, and the resources that compose it operate with dynamic relationships. Some functions that were traditionally performed by management systems are no longer held by them, but autonomously carried out by the network itself.

Operators will be mostly settled about decision-oriented operational tasks for the different network elements. What these decision-oriented tasks are and how they impact the decision elements are main issues. After introducing automaticity, there is a re-assignment of tasks carried out by human network managers, which will focus on the network exploitation enforcement and planning for the future, rather than continuously monitoring the behaviour of particular components.

The building of a network governance framework also faces technological challenges in five main topics:

- Business language:

Network governance is meant to provide a mechanism for the operator to adjust the features of the demanded service/infrastructure using a high level language. In order to achieve this objective a business language may be required that will help the operator to express what it is needed from the network. Such a business language may be modelled by the use of ontology to add semantics and enable machine reasoning on the goals.

➢ Translation:

The indications that the operator defined with business language are afterwards translated to a set of policies that will clearly define the valid operating region for the autonomic functionality.

➢ Reasoning:

Reasoning is also an important challenge in the scope of network governance, as it can be exploited for the mediation and negotiation between separate federated domains. In other words, to allow interoperability between semantically equivalent, but differently instantiated models, it is required to cover multiple standards instead of relying on a single information model. This leads to the use of ontologies for allowing semantic fusion and reasoning with knowledge extracted from data/information.

➢ Policies:

Policies are inherent to network governance. Policies specify rules that should govern the behaviour of the managed elements. Therefore, policies may specify constraints, optimization objectives and functionality that should be followed by the configuration negotiation and selection component, in the particular context. Essentially, this component can refine the information that reflects "what" is generally allowed, in the current situation. This component constrains the options indicated by the context component.

In particular, network governance is almost always interwoven with policies lying at the highest level of the so called policy continuum. In network governance policies are required for the selection of the optimal configuration of a service and for the translation from business level entries and high-level policies to low level policies.

➢ Configuration enforcement:

Configuration enforcement mechanisms are necessary in order to apply the configuration decision. First, it's needed to identify concerned equipments and request each of them to perform the appropriate configuration actions. Then, each of the targeted equipments has to translate and enforce the decision. The term configuration implies self-configuration and includes both configuration and reconfiguration actions (re-optimizations). Reconfiguration actions can be triggered in order to adjust the configuration parameters following network, service and customer conditions.

In short, the real challenge of this task is to design a Network Governance framework based on four technological pillars (high-level language, reasoning, policies and distribution) that results in a system that is able to:

- work with proper business rules and policies, while connecting high level goals and network resources in order to provide the administrator with an appropriate governance interface.
- guide infrastructure behaviours while offering a service view
- offer mechanisms that assist the operator to express goals, objectives, constraints and rules to ensure the desired operation of his autonomic network.
- provide a friendly human interface, aiming to be easy of use, that is not to be used only by highly specialized technicians.
- work in a reliable way, and be able to demonstrate its reliability
- help to convince the operators of the bondage of adopting autonomic approaches.

4.4. State of the art

Governance has appeared as the term to describe the new way (techniques and functionalities) for managing autonomous behaviours instead of being based on the stovepipe type of traditional network management. Prior to putting this in our analysis, we first recall that *policies* are intrinsic to *network governance*. In particular, network governance is tightly interlaced with the concept of policy continuum. Typically, business level policies are defined in the highest level, that is, they express *business objectives*. In the sequel and according to the policy continuum concept these policies are *propagated* to the network going through an arbitrary set of levels (related to different aspects of the management of a communications network) where they are being *transformed* into lower level policies, until they finally reach the element(s) in which to be enforced in terms of low level, technology-specific commands.

4.4.1. Network Governance

In this section is presented a summarized state of the art of autonomic architectures, frameworks and projects, including their contribution to governance.

4.4.1.1. Autonomic Architecture "4WARD"

4WARD architecture was developed during the FP7 4WARD Project [33] which lasted from 1st of January 2008 to 1st of July 2010. 4WARD project [33] aimed to increase the competitiveness of the European networking industry and to improve the quality of life for European citizens by creating a family of dependable and interoperable networks providing direct and ubiquitous access to information. 4WARD's goal was to make the development of networks and networked applications faster and easier, leading to both more advanced and more affordable communication services.

The approach proposed in the 4WARD project is called In-Network Management (INM). Its basic enabling concepts are decentralization, self-organization, and autonomy. The idea is that management tasks are delegated from management stations outside the network to the network itself. The INM approach therefore involves embedding management intelligence in the network, or, in other words, making the network more intelligent. The managed system now executes management functions on its own. It performs, for instance, reconfiguration or self-healing in an autonomic manner.

In order to realize this vision, a management entity with processing and communication functions is associated with each network element or device, which, in addition to monitoring and configuring local parameters, communicates with peer entities in its proximity. The collection of these entities creates a management plane, a thin layer of management functionality inside the network that performs monitoring and control tasks.

The potential benefits of the INM paradigm include the following properties:

- *A high level of scalability* of management systems. For instance, in terms of short execution times and low traffic overhead in large-scale systems. This will allow for effective management of large networks.
- *Fast reaction times* in response to faults, configuration changes, load changes, etc. This increases the adaptability of the network. Together with embedded functions, this will lead to a high level of robustness of the managed system.
- *A high business value* for INM technologies. The higher degree of automation of INM solutions, compared to traditional ones, permits the reduction of operational expenditures. Capital expenditures are also reduced since INM technologies enable a more efficient use of the network infrastructure.

Governance of the network in 4WARD architecture was proposed under the concepts "Governance Stratum" and INM in different work packages and partially demonstrated by an integrated prototype. For the network a policy-base rule [34].

4.4.1.2. Autonomic Architecture "AUTOI"

This section presents a new autonomic management architectural model developed in the AutoI project ([35] - [39]), which will last from January 2009 – September 2011. This model applies to the self-management of virtual networks over any existing network infrastructures. This architectural model consists of a number of distributed management systems described with the help of five abstractions - the OSKMV planes: Orchestration Plane (OP), Service Enablers Plane (SP), Knowledge Plane (KP), Management Plane (MP) and Virtualization Plane (VP). Together these distributed systems form a software-driven network control infrastructure that will run on top of all current networks (i.e. fixed, wireless and mobile networks), servicing physical infrastructures. This provides a means for networked devices or attachments to connect to each other and through each other to the outside world, providing seamless service provisioning. The main purpose of the OSKMV planes is to make the Future Internet of Services self-knowledgeable, self-diagnosing and ultimately fully self-managing.

In the AutoI model the governance functionality was developed as part of the Orchestration plane. A specific Distributed Orchestration Component (DOC) was developed as a network service. It collects appropriate monitoring data in order to determine if the virtual and non-virtual resources and management services that it governs need to be reconfigured. Business goals, service requirements, context, capabilities and constraints are all considered as part of the decision making process.

The governance is performed by the set of Autonomic Management Systems (AMS) that are responsible for managing each component or set of components; each AMS uses the Virtualization System Programming Interface (vSPI) to express its needs and usage of the set of virtual resources to which it has access. The vSPI is responsible for determining what portion of a component (i.e., set of virtual resources) is allocated to a given task. This means that all or part of a virtual resource can be used for each task, providing an optimized partitioning of virtual resources according to business need, priority and other requirements. Composite virtual services can thus be constructed using all or part of the virtual resources provided by each physical resource.

4.4.1.3. Autonomic Architecture "CASCADAS"

Integrated Project FET CASCADAS (FP7) ([40], [41]), which lasted from January of 2006 till December of 2008, put forward the vision of a service networking eco-systems, for convergence of Telecommunications, ICT and Future Internet. Architecture is based on a distributed framework of autonomic lightweight components, abstracting resources and interacting with each other to provide services and functions. Project developed and assessed a tool-kit based on Autonomic Communication Elements (ACE) and some related libraries (e.g. for self-management, self-organization, security, knowledge networking) to demonstrated experimentally a use-case.

ACEs represent an engineering effort aimed at providing a component-model through which heterogeneous basic services can be designed and developed in accordance with requirements drawn from self-* properties. In ACE model, high level policies are specified by human Operators through a number of states, along with the transitions that lead the ACE execution process from one state to another, in a way similar to I/O Automata. In terms of ACEs, these policies are called plans. They represent structured behavioural directions an ACE is required to follow, and the overall ACE behaviour can be specified within one or more plans that can execute concurrently or sequentially. The totality of plans specifies the overall behaviour for an ACE, and is enclosed in the ACE's self-model. The ACE behaviour can be enhanced with modification rules, through which it is possible to specify the circumstances under which the original behaviour can be relinquished, along with the new behavioural directions to follow. The ACE behaviour is specified explicitly, with the help of a definition language. It is possibility to isolate part of the behaviour specification within the self-model of one ACE and share it as a sort of behavioural pattern among ACEs of similar nature. This "Behavioural Reusability" is achieved through Common Behaviours, i.e., a self-contained set of plans, states and transitions that are made available among a set of ACEs requiring their usage. This feature enable the reuse of already existing plans, so as to ease the development of self-models. However, their use also opens a much wider range of possibilities in terms of dynamic adaptation and extensibility. For instance, consider an ACE-based ecosystem where ACE instances (phenotypes) behave according to their self-models (genotypes). A detailed description is reported in [42].

4.4.1.4. End-to-End Efficiency (E3) Autonomic Architecture

Designed in the context of the EU funded IP project "E^3 – End to End Efficiency" [43] which lasted from January 2008 until December 2009. Part of this architecture has been approved as a feasibility study by the ETSI Committee Reconfigurable Radio System (RRS) [44].

E^3 used an evolutionary approach i.e. gradual, non-disruptive evolution which incorporates principles of existing network management systems, such as reconfigurable radio networks based on Software Defined Radio

(SDR), with new approaches, such as Cognitive Wireless and self-x Networks.

The aspect of goal-based management and policy-based decision-making as an approach of governance has been investigated in E^3 in various mechanisms. For example, the approach for self-organizing cognitive network elements is based on such a framework and utilizes policy rules to achieve dynamic cluster formation of network nodes, targeting energy/topology optimization. The interfaces of the Dynamic Spectrum Management (DSM) and the Dynamic Self-organizing Network Planning & Management (DSNPM) are used as the human to network interface and specifically for the update of policy rules [45]. Since, business oriented policy based management is not generally introduced in E^3, support for governance is achieved with low level policy based management.

4.4.1.5. Autonomic Architecture "FOCALE"

"FOCALE" is an architecture developed in Motorola Labs - Waterford Institute of Technology ([46], [47]) and relies on a dynamic knowledge base with learning and reasoning mechanisms, and it emphasizes the role of the Information Model (DEN-ng) as a prerequisite for relevant knowledge dissemination. It makes use of distributed agent driven by a policy-based manager through business objectives.

Network governance in FOCALE is envisioned through the definition of high level policies that are translated to more specific ones so forth.

Business objectives described in terms of goals are enforced in the autonomic system and translated between different levels of the policy continuum. The Policy Continuum guides the transformation of goals into policies, which contain actions that are enforced at the appropriate control point.

So, governance criterion is ranked as Business oriented Policy based Management / Governance of autonomic network elements.

4.4.1.6. Autonomic Architecture "GANA"

Generic Autonomic Network Architecture (GANA) was developed during the EC FP7 EFIPSANS PROJECT (INFSO-ICT-215549) which lasted from 1st January 2007 till 31st December 2010 and it adopted incremental or evolutionary approaches ([48], [49]).

According to GANA architecture the network governance interface is meant to provide a mechanism for the operator to adjust the features of the demanded service/infrastructure using a high level language. The operator expressed what it was needed from the network through a business language and then that language was modeled by the use of ontology to add semantics and enable machine reasoning on the goals. These indications are afterwards translated to a set of policies that will clearly define the valid operating region for the autonomic functionality.

In order to achieve this objective the GANA framework provides the concept of "Network Profile" as the tool to distribute the information on how the network should behave. A Network profile is actually a mechanism for policy encapsulation and distribution.

So, governance criterion is ranked as Business oriented Policy based Management / Governance of autonomic network elements.

4.4.1.7. Autonomic Architecture "Self-NET"

The Self-NET architecture is the outcome of the Self-Management of Cognitive Future InterNET Elements Project of the 7th Framework Programme (FP7) of the European Commission whose duration is from 1st May of 2008 till 30th November of 2010 (30 months) [50]. Self-NET follows an evolutionary approach since successful principles of existing network management systems have been taken into account and new paradigms (cognitive systems, complex systems) have been adopted.

The Self-NET architecture allows the description and update of policy rules that are used for the transfer the goals of a network operator for communication systems' governance, for the identification of faults/optimizations and for the selection of the appropriate configuration. The Self-NET architecture used certain interfaces such as the human to network interface for the update of the policy rules.

4.4.1.8. Outcome of the several autonomic architectures

To conclude, many of the examined initiatives worked towards a real governance of self-managed behaviours aiming at resulting into autonomic networks and systems with the ability to dynamically adapt to changes in accordance with high-level business policies ([35], [46], [52]).

The analysis also showed that a great set of the examined architectures adopted traditional management solutions. They covered lower level policy based management with the focus being placed on the network side in particular, but they do not consider high level business goals nor offer a service view. Sometimes the aspect of governance was not even captured as an item of their research agenda [52].

Last but not least, little attention was paid to the novel type of dialogue between a human network operator (HNO) and the envisaged, self-managed network and in particular to the innovations and peculiarities that most probably the HNO will need to handle while migrating to this new type of management.

A more detailed elaboration on the state of the art can be found in the Milestone MS24 document of UniverSelf project [51].

4.4.2. Business language

In large-scale, distributed systems such as a production-level autonomic network, an implicit or explicit agreement between a client and a service provider specifies service level objectives, both as expressions of client requirements and as provider's assurances. These objectives are expressed in a high-level, service-, or application-specific manner rather than requiring clients to detail the necessary resources. There have already been some efforts aiming to consolidate high level information in the form of a business model or language.

4.4.2.1. Common Information Model

The Common Information Model (CIM) [53] provides a common definition of management information for systems, networks, applications and services, and allows for vendor extensions. The CIM model is proposed by the Distributed Management Task Force (DMTF), which is an industry organization that develops, maintains and promotes standards for systems management in enterprise IT environments. CIM's common definitions enable vendors to exchange semantically rich management information between systems throughout the network. CIM is composed of a Specification and a Schema. The Schema provides the actual model descriptions, while the Specification defines the details for integration with other management models. In addition, the CIM Policy Model [54] based on the CIM model enables administrators to be able to represent policies in a vendor-independent and device-independent way. Thus, service level and other high- level policy abstractions can be supported, and be translated to device-specific configuration parameters at a lower level, across an aggregate of heterogeneous managed entities.

4.4.2.2. Shared Information and Data model

The Shared Information and Data model (SID) [55] is a set of comprehensive standardized information definitions, developed by the

TeleManagement Forum (TMF), acting as the common language for building easy to integrate OSS (Operational Support System) and BSS (Business Support System) solutions. The SID model focuses on what on business entities and associated attribute definitions. The adoption of the SID as the industry’s standard information model is growing rapidly, with many service providers, vendors, and systems integrators using the SID as the basis for their development and integration.

4.4.2.3. Other type of formalization

Often the business information is formalized in a Service Level Agreement (SLA). It is worth mentioned the standardization work of Web Services Agreement (WS-Agreement) [56], which defines a protocol and the respective abstract model for linking agreements to services, irrespective of the domain-specific details of contract terms. The EU funded SLA@SOI project, as part of its research agenda, has proposed a syntax grammar to express SLAs [57], including the business terms [58] needed for the relationships with the customers and with third parties.

4.4.3. Translation mechanisms

Network operators, on the other hand, require low-level, resource specific performance criteria that can easily be interpreted and provisioned. As a consequence, a framework that addresses the gap between high-level specification of client performance objectives and existing resource management infrastructures of network operators is traditionally required.

Former European projects have developed some work relative to translation. EFIPSANS project has implemented a mechanism that allows the translation from high-level goals to network policies [59]. A prototype has been developed, based on an ontology model, and implementing the translation by means of SWRL rules. Nevertheless, the generated policies described only the specifications for the network monitoring. That is, only a fraction of the service lifecycle was covered. SLA@SOI project presents a theoretical approach [60] for the translation of SLAs across layers, but an implementation has not yet been achieved.

4.4.3.1. Semantics & reasoning

Ontologies are formal representation of knowledge within a domain, that is, a description of concepts and relationships between them. The importance of the ontologies comes from the fact that they enable knowledge sharing and reuse [61]. In the last years, several ontology languages have been developed, both proprietary and standards-based.

4.4.3.2. DARPA Agent Mark-up Language

The WWW Consortium (W3C) developed the Resource Description Framework (RDF) a language for encoding knowledge on Web pages to make it understandable to electronic agents searching for information. The Defence Advanced Research Projects Agency (DARPA), in conjunction with the W3C developed the DARPA Agent Markup Language (DAML) by extending RDF with more expressive constructs aimed at facilitating agent interaction on the Web.

4.4.3.3. Web Ontology Language

Some years later, the W3C Web Ontology Working Group presented the OWL (Web Ontology Language) [62], which is one of the most popular ontology languages. OWL is a mark-up language to describe the properties and capabilities of the information in such a way that the descriptions can be interpreted by a computer system in an automated manner. OWL allows applications to automatically discover, compose, and invoke services in a dynamic services-oriented environment. It can be enhanced with an inference engine in order to allow reasoning. In addition, OWL is maintaining as much compatibility as possible with the pre-existing languages, including RDF and DAML. Recently, the W3C OWL Working

Group, a follow-on group of W3C Web Ontology Working Group, developed the OWL2. OWL2 is an extension and revision of OWL and it is designed to facilitate ontology development and sharing via the Web, with the ultimate goal of making Web content more accessible to machines.

4.4.3.4. Semantic Web Rule Language

SWRL (Semantic Web Rule Language) [63] is a proposal for a Semantic Web rules-language, combining the Ontology Web Language (OWL) and the Rule Markup Language (RuleML) [64]. Both rules and ontologies are necessary for the service descriptions and play complementary roles: while ontologies are useful for representing hierarchical categorization of services overall and of their inputs and outputs, rules are useful for representing contingent features such as business policies, or the relationship between preconditions and post-conditions. SWRL enables to “build rules on top of ontologies": it enables rules to have access to ontological definitions for vocabulary primitives (e.g., predicates and individual constants) used by the rules.

4.4.4. Policy model, Policy language & Policy framework

Network Governance heavy relies on a policy based Management model for defining and controlling the network behaviour shifting from classical paradigms focused on individual devices/entities management. Policies are a set of pre-defined rules (defined actions to be triggered when a set of conditions are fulfilled) that govern resources, including conditions and actions that are established by the administrator with parameters that determine when the policies are to be implemented in the network. In the case of a Telco Operator, policies are defined based on the high-level business objectives of the services on one hand and on the other hand on the SLA agreed with its customers and third party Service Providers. Policies allow changing the behaviour of a system without changing its implementation, creating adaptable systems whose behaviour can be altered dynamically.

Although Policy Based management (PBM [62]) seems to fulfil some of the governance requirements, current architectures need to be enhanced to include the key concept of knowledge and context awareness. Most PBNM (Policy Based Network Management) systems define low-level policies that manage changes in routers, switches or firewalls. The link between the business needs and the configuration of the network resources and services is missing. Novel concepts such as the usage of the network context to determine the modifications in network services and resources are not present in these approaches.

In contrast, the DEN-ng information model [63] includes policy and context sub-models, and has been expressly constructed to facilitate the generation of ontologies, so that reasoning about policies constructed from the model may be done. A policy language can also be derived from the model. This holistic construction of information model, ontologies and policies makes it suitable for the government of management entities. DEN-ng is built on the 3 principles governing the system: capabilities, constraints and context with the following precisions. Capabilities normalize the set of functions available in the same type of managed object made by different vendors. Context defines the current environment, objectives, obligations and policies governing the behaviour of the system. Constraints define which capabilities can be used as a function of a particular context. These three principles enable the behaviour of the system to be abstractly modelled.

DEN-ng specific improvement is to define not only the static characteristics of the managed entities but also dynamic ones (behaviour) in a manner independent of any specific type of repository, software usage or access protocol.

Related with DEN-ng is the concept of the Policy Continuum [64], which defines a framework for the development of stratified sets of policy languages, tied together by a common information model. This helps ensure the consistency of the policies deployed across a system and facilitates policy-based analysis processes. This continuum concept seems to be suitable for the achievement of the objectives of Task 2.3.

The DEN-ng information model and the concept of Policy Continuum are realized in the context of FOCALE autonomic architecture ([53], [46]). The FOCALE architecture is based on five key concepts:

- The use of a shared information model capable of harmonizing the different data models that are used in Operational and Business Support Systems (OSSs and BSSs).
- The knowledge extracted from information and data models is augmented with the use of ontologies in order to be capable of representing the detailed semantics required to reason about behaviour.
- The existence of an information model able to generate ontologies for governing behaviour.
- The policy model is linked to a context model, so that policies can be written that adapt offered resources and services according to context changes.
- The use of machine learning and reasoning.

Finally, it is worth mention here the commercial solution PCRF (Policy and Charging Rules Function), defined as a node that at runtime determines the policy rules to be applied in a multimedia network. Currently different implementations are available from different vendors, such Huawei, Openet, Camiant/Tekelec.

4.4.5. Conflict resolution

The use of a policy based management system may potentially lead to the appearance of conflicts between policies. These conflicts may be resolved informally by human managers, but the goal of the Network Governance Framework is to provide an automated mechanism to recognise them and resolve them. A general purpose algorithm for policy conflict detection and harmonization has been implemented in KaOS project [65]. Using DAML policy ontologies, the method is based on an assignment of priorities to policies. In case of conflict, the numeric priority and the update times are the criteria used to determine the precedence.

[66] and [67] present an exhaustive work where the use of information models and ontologies that represent relationships between policy components facilitate the detection of conflicts. The approach includes the design of an analysis algorithm for the policy continuum concept that could be used to analyze policies at the different levels of the continuum.

A different approach is introduced in [68], which describes a framework for policy analysis, conflict detection and resolution applied to the QoS management for DiffServ networks. Event Calculus is used as the underlying forma representation of policies, systems behaviour and rules to detect the presence of conflicts.

4.4.6. Distribution & enforcement mechanisms

The different approaches have found different solutions to the problem of distributing the policies to the managed elements. EFIPSANS [48] project has defined its own mechanism for the communications between managed entities, the ONIX (Overlay Network for Information Exchange). ONIX is a distributed system of servers, that supports publish/ subscribe, query and find type of services for Information and Knowledge such as Capabilities of network elements, Profiles, Goals and Policies of the autonomic network, pointers to resources and Data, and other types of Information/Knowledge.

DEN-ng defines a hierarchy for policy management application [69]. It splits the functionality of a Policy Enforcement Point into a Policy Execution Point (PXP – that implements specified policy actions) and a Policy Verification Point (PVP – that ensures that the policy actions were executed correctly and with the expected results). A Policy Decision Point (PDP) distributes various levels of decision-making amount global and local scopes. DEN-ng defines a PolicyServer as consisting of at least one PDP, PXP and PVP, while a Policy Broker is the entity that enables multiple Policy Servers to negotiate and exchange policies.

4.5. Describe the adopted methodology / approach

4.5.1. Human-To-Network (H2N) interface

This section describes a Human-to-Network (H2N) interface that is being developed as part of the Governance functional block (Figure 1).

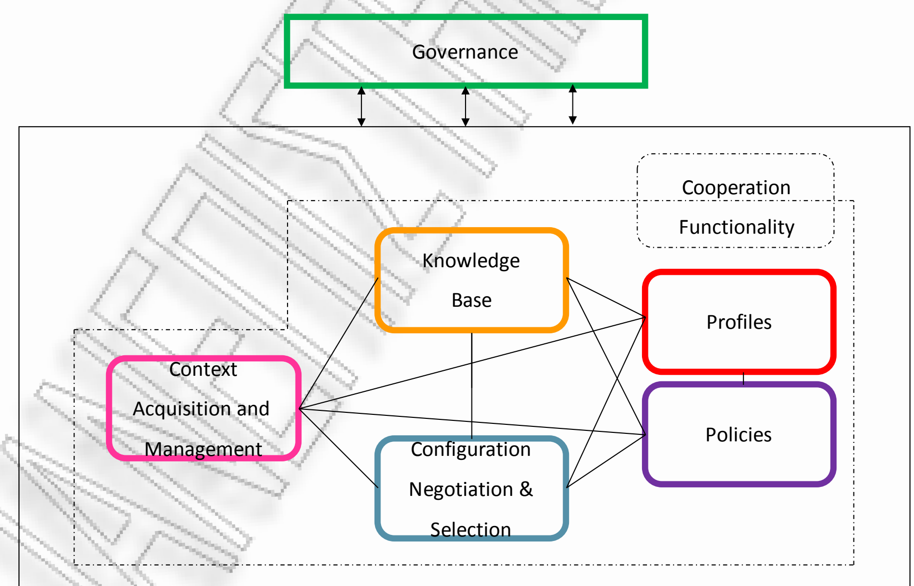


Figure 1: UMF functional blocks

The main functionality of this H2N interface is to provide a tool for the human operator to insert high-level business goals, which are then translated autonomously into technology-specific terms autonomously so that the human operator does not need to deal with any technical details. A high-level view of the role of the H2N interface is depicted in Figure 2. Business goals may be related to the introduction of a new application, sets of user classes for the application, sets of Quality of Service (QoS) levels for each user class of the application, etc. This introduction can be related to a specific location, time period, volume of users, etc. Furthermore, the H2N interface allows the associations of applications/services with User Classes, QoS levels, network technologies, other applications, QoS levels with QoS parameters. This influences the Policy functional block.

Figure 2: High-level view of H2N interface role

Moreover, the H2N interface allows for the configuration of the number of users anticipated for an application, the corresponding user class and Quality Level in a certain location and at specific time zone. These high-level objectives/policies need to be further propagated to the network going through an arbitrary set of levels (related to different aspects of the management of a communications network) and be transformed into lower level policies so that they reach the element(s) in which to be enforced in terms of low level, technology-specific commands. Consequently, the already set business goals are forwarded to the Context Acquisition functional block (Figure 1) in order be translated from service requirements into network configuration (technology-specific terms) and leave the system to autonomously work out the situation and meet the objectives. The H2N interface also allows feedback, e.g. the result of diagnosis or a visualization of the monitoring to the system administrator/operator.

4.5.1.1. Information flow

As already introduced, governance allows the introduction of the business level goals/policies in high level terms through the human-to-network (H2N) interface. Typically, business level policies define high level expression of business objectives. In the sequel and according to the policy continuum concept these policies are propagated to the network going through an arbitrary set of levels (related to different aspects of the management of a communications network) where they are being transformed into lower level policies, until they finally reach the element(s) in which to be enforced in terms of low level, technology-specific commands. Policy derivation and management translates high level goals/objectives provided through Governance into low level policies and often into low level self-configuration enforcement policies.

In the scope of the work done so far on the H2N interface, two types of high-level policies and corresponding messages have been identified: Business level entries and Associations, analysed in sub-sections 4.5.1.2 and 4.5.1.3 respectively. In terms of low level policies, enabling self-configuration of user devices and network elements, a first specification of Configuration policies is provided in sub-section 4.5.1.4.

4.5.1.2. Business level entries

This sub-section presents business level entries as an example of high-level goals (policies). Business level entries are information provided at the business level related to the number of users anticipated for an application, user class, in a certain location and time zone. In more detail, as can be observed in Figure 3, business level entries comprise information on the Number of users, the Traffic percentage, i.e. the number of concurrently active users anticipated, the Location (e.g. Piraeus, Athens-center, ...), the Time Zone (e.g. 08:00-11:00, 21:00-22:00, ...), the Application (e.g. IPTV, ...), User Class (e.g. Gold, Silver, Bronze, ...), Quality Level (e.g. High, Medium, Low), Quality level parameters (e.g. Bit rate, delay, jitter, ...) and the Mobility pattern (e.g. High, low, train, car, ...).

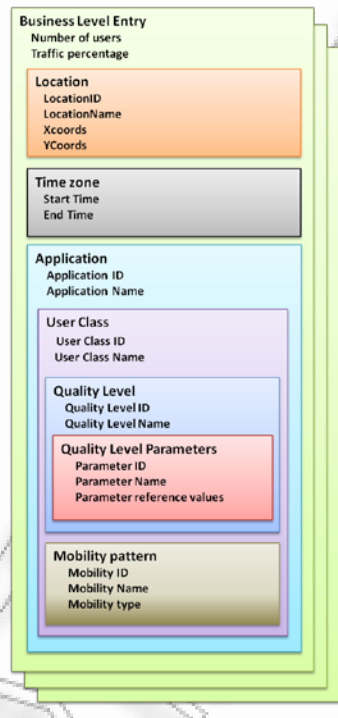


Figure 3: Business level entry structure

4.5.1.3. Associations

This sub-section presents associations entries as an additional example of high-level goals (policies). Associations are high-level policies that specify rules related to the relationship of applications with user classes and quality levels, the relation of a certain application with other applications and the relations between user classes. As can be observed in Figure 4, an association comprises information on a set of applications. Each application may be associated with one or more User Classes. Each User Class may be associated with one or more QoS levels. Each QoS level is associated with one or more QoS parameters.

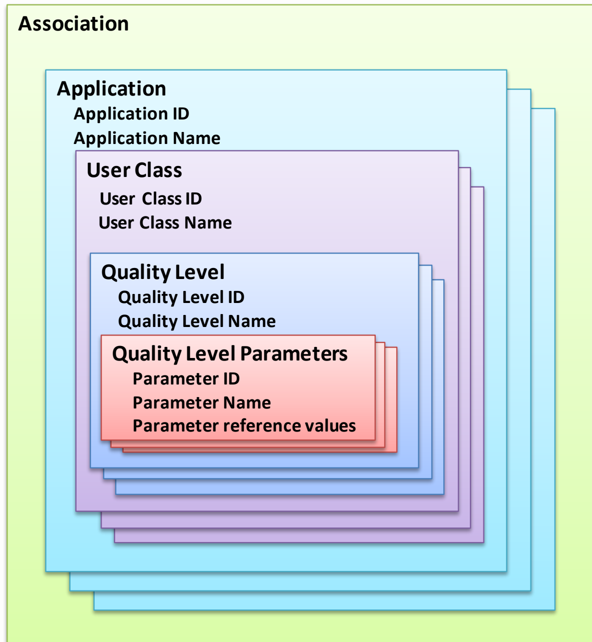


Figure 4: Structure of high-level policies (associations)

4.5.1.4. Configuration policies

Configuration policies specify rules or constraints that should be taken into account for the selection of the optimal configuration of a service area, Access Point or Device. Configuration policies should be derived from business level entries and high-level policies/associations (Figure 5). A configuration policy is formulated as a set of a Compound Policy Condition and a Policy configuration. A Compound Policy Condition comprises a Logical Expression (e.g. AND, OR, XOR) and one or more Compound Policy Conditions or Policy Conditions. A Policy Condition encompasses a Policy Expression (e.g. "equals", "greater than", "greater equal", and "less than", "less equal", etc) and a Policy Argument. A Policy Argument includes User Class, Location, and Time zone information and basically indicates the user devices that are affected by the specific policy. A Policy Configuration indicates the Radio Access Technologies (RATs) that can be operated by transceivers of Access Points, as well as certain frequency bands per RAT in a certain service area. Moreover it also may specify the services and corresponding QoS levels that can be provided over certain RATs.

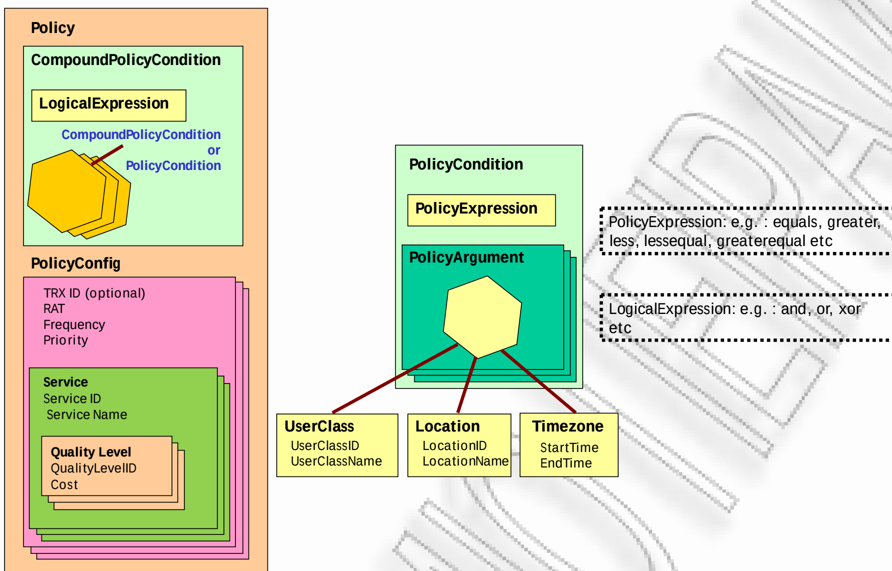


Figure 5: Configuration Policies structure

5. IMPLEMANTATION AND FUNCTIONALITY OF H2N INTERFACE

5.1. Introduction

In this part a detailed overall presentation of the interface that was developed will be made. As previously noted Human-to-Network interface is a high level governance mechanism which enable operators to express their objectives in order to introduce policies and business goals to the network.

The technical specifications of the integration and the design issues taken into account to built the interface are presented in section 5.2. Thus, the functionality of the H2N interface is presented in section 5.3. Finally, this part concludes with the introduction of the next phases and a brief presentation of the sequel UMF functional block.

5.2. Implementation Requirements

H2N provides a graphical user interface (GUI) allowing operator to interact with the system. The current version of the H2N interface is a Java code written interface part of which can be found in APPENDIX I:

Governance functional block is an agent that can be defined as an autonomous entities situated in some environment capable of flexible behaviour. Being an agent, it allows view a system as being composed of autonomous interacting entities which pursue their own goals and act in a rational manner. For that reason it is built in Java Agent DEvelopment Framework (JADE) [70], which is an open-source, agent platform that provides various application orchestration functionalities and allows for high- level interfacing mechanisms between components based on XML.

JADE Agent Platform [71] provides a full environment for agents to work. JADE is fully implemented in Java and it includes a bundle of tools that simplify platform administration, application development and deployment.

JADE *platform* is composed of agent *containers* that can be distributed over the network. Agents live in containers which are the Java process that provides the JADE runtime environment and all the services needed for hosting and executing agents. There is a special container, called the *main container*, which represents the bootstrap point of a platform. Main container is the first container to be launched and all other containers must join to a main container by registering with it. If another main container is started somewhere in the network it constitutes a different platform to which new normal containers can possibly register. As a bootstrap point, the main container has the following responsibilities: it manages the registry of all container nodes composing the platform; manages the registry of all agents present in the platform including their current status and location; and hosting the Agent Management System (AMS) and the Directory Facilitator (DF) agents.

JADE provides a homogeneous set of APIs that are independent from network and java version (i.e. same APIs for JEE, JSE and JME java environments) [72]. The JADE run-time environment can be executed on a wide range of devices ranging from servers to java enabled cell phones.

JADE allows agents to dynamically discover other agents and to communicate with them according to the peer-to-peer paradigm. From application point of view, each agent is identified by unique name and it provides a set of services; it can register and modify its services and/or search for agents providing given services, it can control its life cycle and, finally, it communicates with all other peers.

In Figure 6 is the depicted the Jade platform that was used in our implementation. This figure is constructed by two parts; the left part depicts the containers which functionality just described above and the right part which depicts the active agents and their messages' exchange. As already described in section 4.5.1.3 in H2N interface is composed an Association Notification message, including the specifications that are shown in Figure 4, which are:

- Associations of applications with user classes and quality levels
- Relation of a certain application with other applications
- Relations between user classes
- Recommendations regarding the association of application and network technologies (including potential affiliate providers).

When the other agent receives this message it responses with an acknowledgement which is called *AssociationNotificationAck*. The latter agent is the Context Acquisition Function block (brief mention can be found in section 5.4).

A second message is also shown at the below figure (*BusinessLevelEntryNotification*) which is also composed in H2N interface and includes specification of user classes, QoS levels, target location and time, volume of users etc of an application (see section 4.5.1.2 and Figure 3). Context Acquisition Function block response again at Governance agent with an acknowledgement by sending the requested parameters, called *BusinessLevelEntryNotificationAck*.

All above mentioned messages are Java classes which are referred as "plans" in the XML file of Governance agent. Their code can be found in the APPENDIX I:

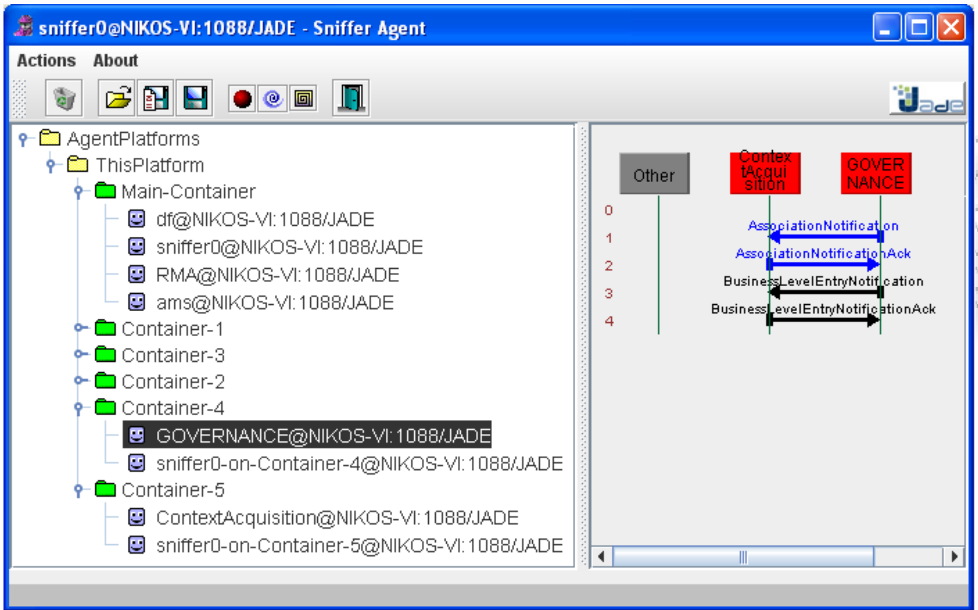


Figure 6: Depiction of Jade platform

To combine agents JADE in multi-agent environment that has «beliefs», «goals» and «plans» we chose the platform Jadex. The Jadex project accommodates these properties with an open research map and will allow us to consider especially the communication of agents in multi-agent environment. The beliefs, goals and plans are the objects of first-class for agent, which we can be create for agent and which we can be manipulated.

Firstly, in an agent file is used the <belief> tag that provides a name and a type. The name is used to refer to the fact(s) contained in the beliefs, and the type specifies the (super) class of the fact objects that can be stored in the beliefs. In our case the name is “gui” and the type is “class=GOVERNANCEgui”.

This belief is has a goal, called “startGOVERNANCE” which initialize the whole agent’s functionality. Last but not least, Governance agent contains «plans» and «events» which are both java classes and refer to the message event and the each message respectively.

Last but no least, for the integration of this tool is also used a SQL database. The current database was called "governancedb" and its functionality was to keep a registry of the changes and the rules which structure defined policies. In other words, its functionality can be characterized similar with that of Directory Facilitator.

The "governancedb" includes several tables which are modified in real time from H2N interface with "insert", "delete", "select" etc commands. Such tables are:

- initparameters
- initsq
- locationdata
- mobilitydata
- parametersdata
- poadata
- qualdata
- servdata
- servrules
- techdata
- timezonedata
- trfdata
- trfrules
- ucdata

This database was built with the use of "EMS SQL Manager 2005" program.

5.3. Functionality of H2N interface

Figure 21 provides an overview of parameters that can be modified through the Graphical User Interface (GUI) of the H2N interface developed so far. In sequel, each parameter is analyzed.

- Application

In other word is called as a service and it is the main body of H2N interface. Polices that are derived from this tool have as core the applications. Figure 8 depicts its modification. More specifically, there is a list in the upper part of this figure that addresses the already defined applications. These can be edited by changing their name or deleting them. In addition, a new one can be added by typing its name in the field at the bottom of this page and pushing the appropriate button.

The buttons hide the functionality of the tables' editing in the database. In current case the table that is modified is the "servdata".

- User Class

Similar functionality with applications has the modification of "User Class" parameter (see Figure 9). However, the "ucldata" table in the database is modified in this case.

User Classes offer a way to organize applications into groups, which can aid in performing group-level actions. For example, some these categorizations are *gold*, *silve*r, *bronze*, *normal*, etc.

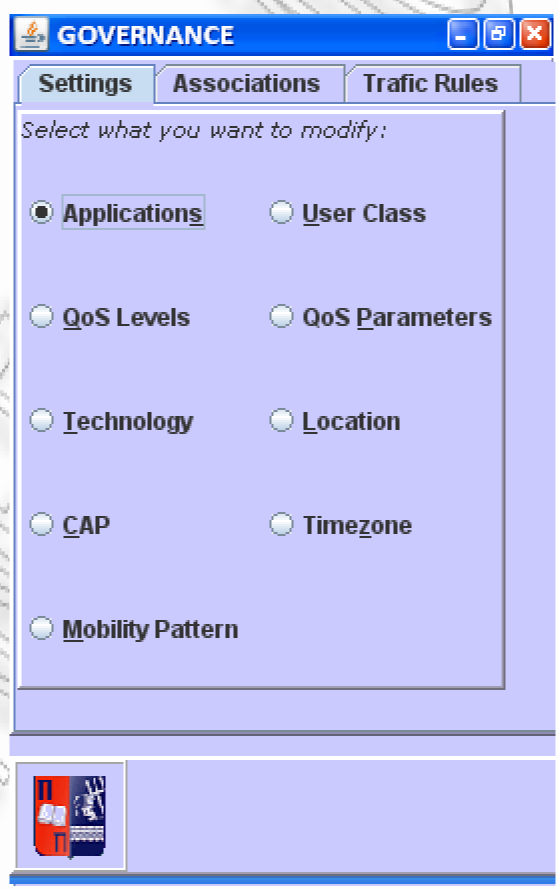


Figure 7: Menu that give options for parameters' modification

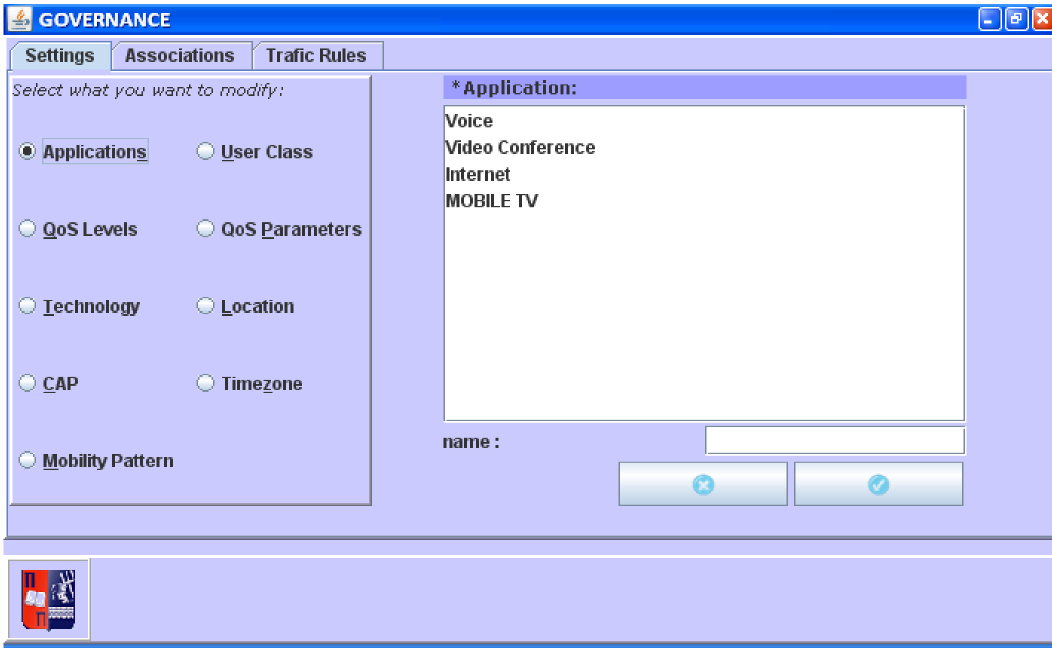


Figure 8: Modification of "application" parameter

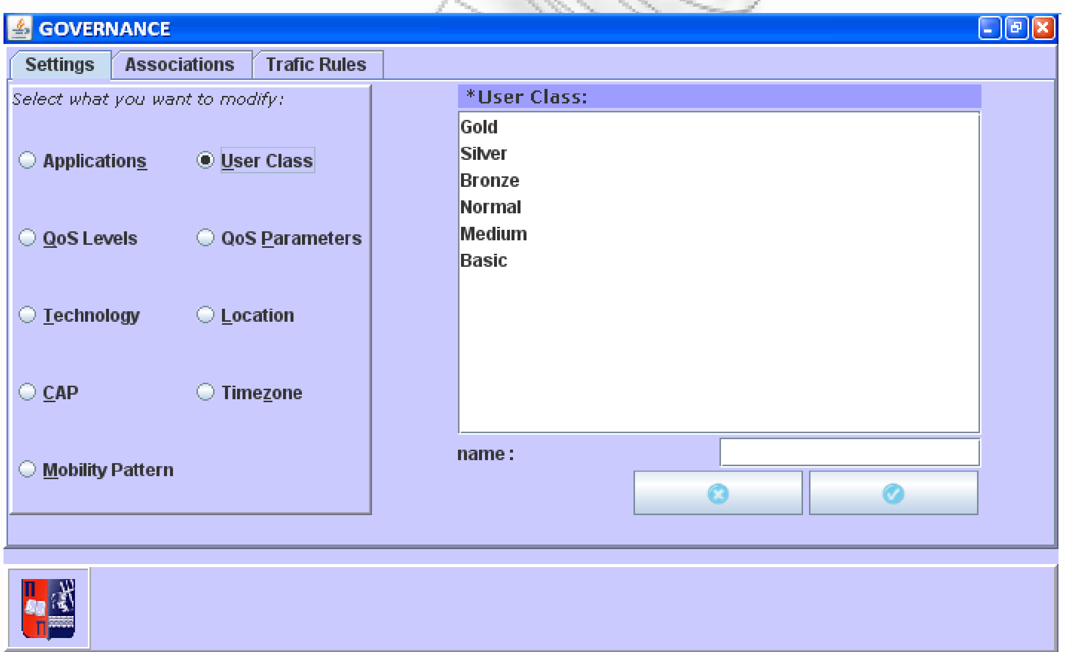


Figure 9: Modification of "user class" parameter

➢ QoS Levels

Each application can adapt to a new level of QoS. But what is the QoS level? The level of Quality of service is the ability to provide different priority to different applications, or to guarantee a certain level of performance. These can be guaranteed by QoS parameters that we will se in the sequel.

As we can see in Figure 10 the QoS level is defined by a "name", its association with an application and a set of parameters. For the editing of the parameters that match to each QoS level there is an additional option in this option of GUI, as Figure 11 shows, which offers menus with parameters names and values so that the user to select the right values for each QoS level. This is importance as the QoS guarantees the network's efficiency, especially for real-time streaming multimedia applications such as voice over IP, online games and IP-TV, since these often require fixed bit rate and are delay sensitive, and in networks where the capacity is a limited resource, for example in cellular data communication.

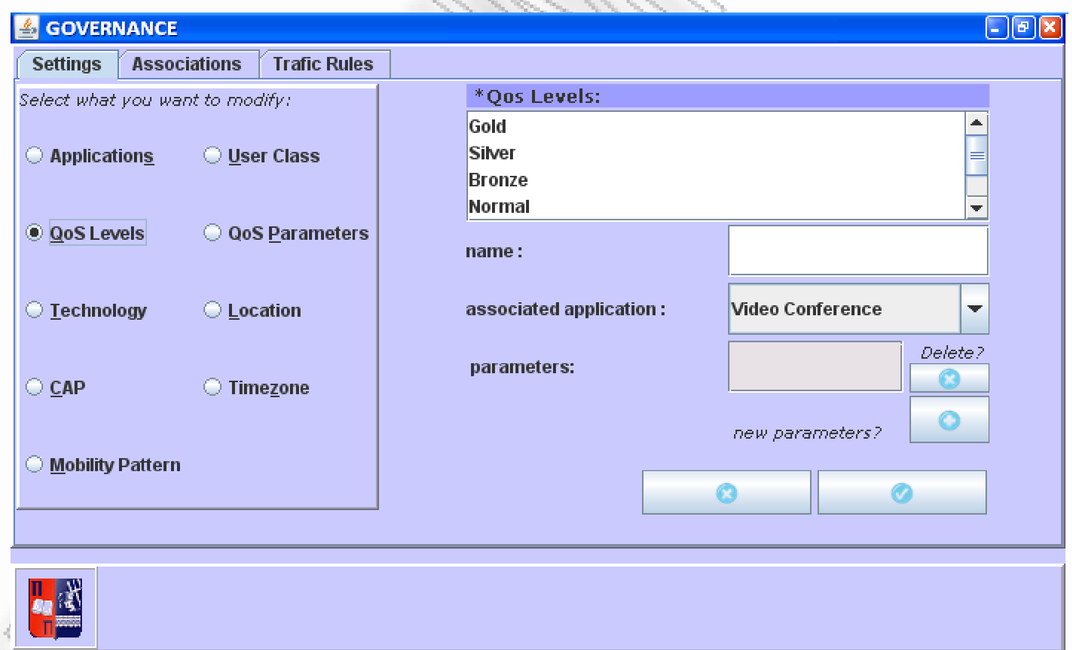


Figure 10: Modification of "QoS Level" parameter

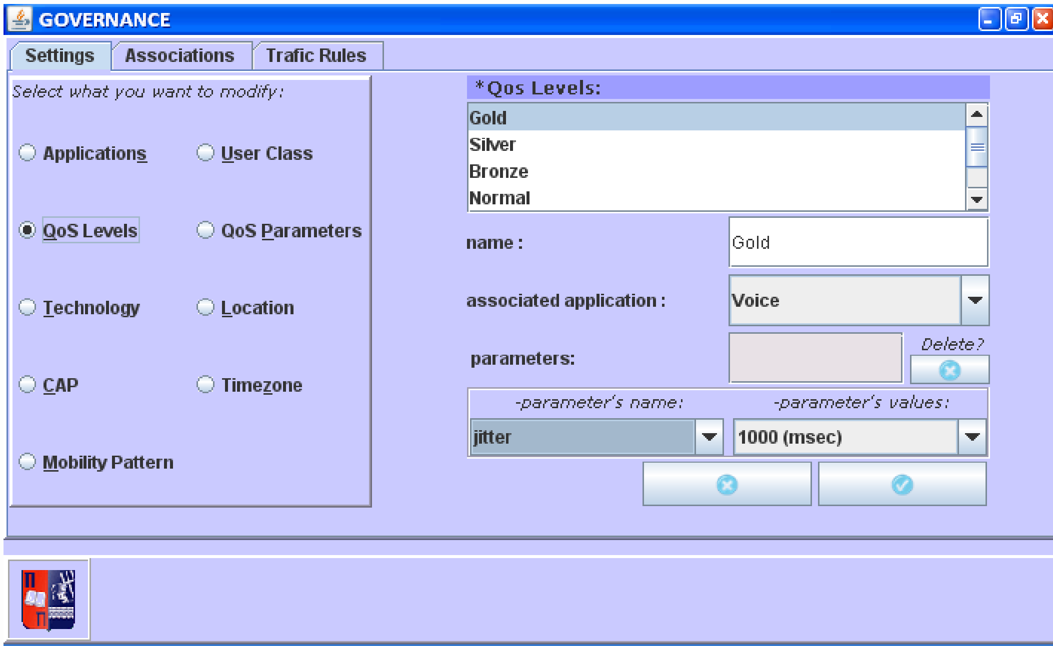


Figure 11: Modification of "QoS Level" parameter by clarification of certain parameters

➢ QoS Parameters

Quality of service (*QoS*) *parameters* are the parameters that control the priority, reliability, speed and amount of traffic sending over a network e.g., throughput, transit delay, error rate, a required bit rate, delay, jitter etc.

As we can see in Figure 12 a QoS parameter is defined by:

- a name
- reference values
- a measurement unit.

If a new name of QoS parameter is given, it is added in the first field at in interface which is a list. Then, (see Figure 13) the can set the new values by pushing the appropriate button for this option. At the same time, that area of the GUI offers to users more option for the parameters' editing.

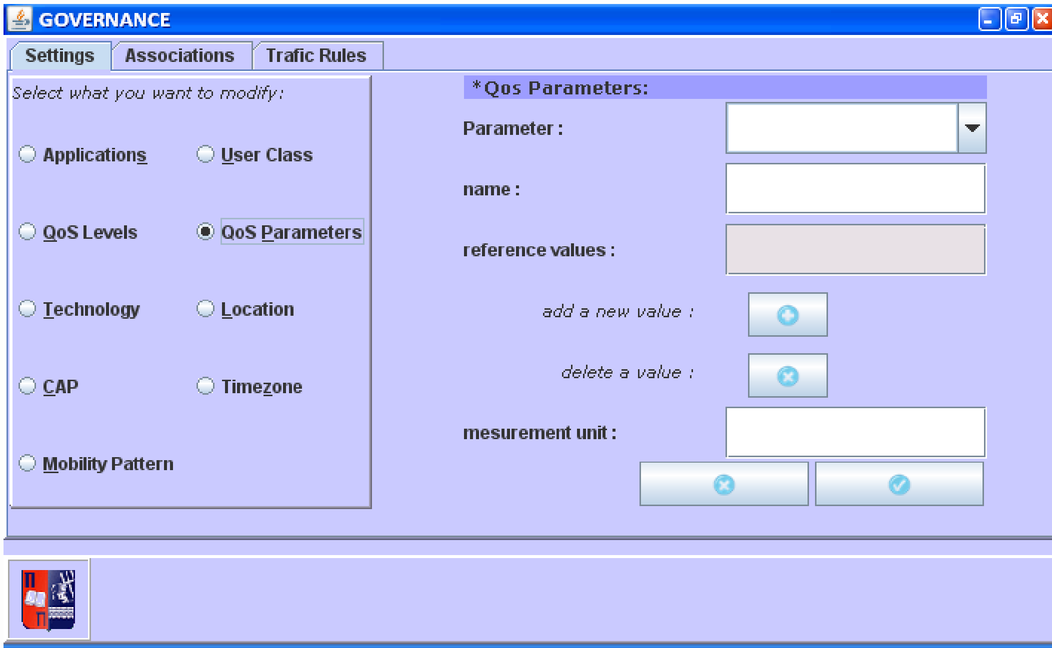


Figure 12: Modification of "QoS parameters"

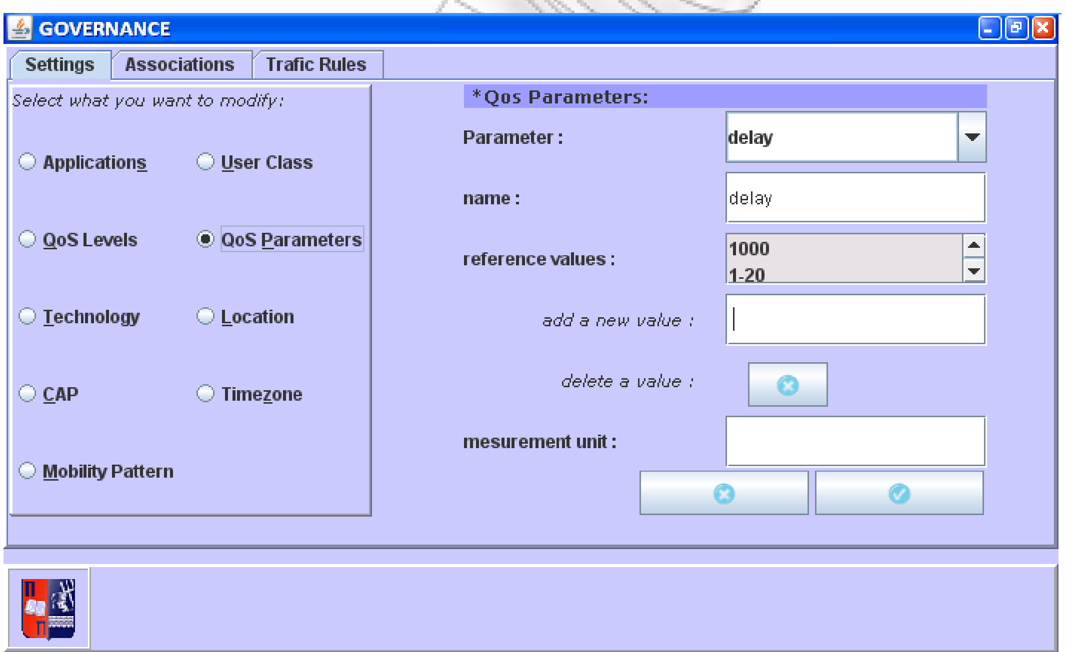


Figure 13: Modification of "QoS parameters" by modifying their reference value

- Technology

The next characteristic that is able to be modified from this GUI is the "technology". The arranged applications are strictly combined with the technologies and that is the reason why this configuration could not miss of an interface of network management (see Figure 14).

The definition of a technology is set by its name and range and registries are saved at our database, in the "techdata" table.

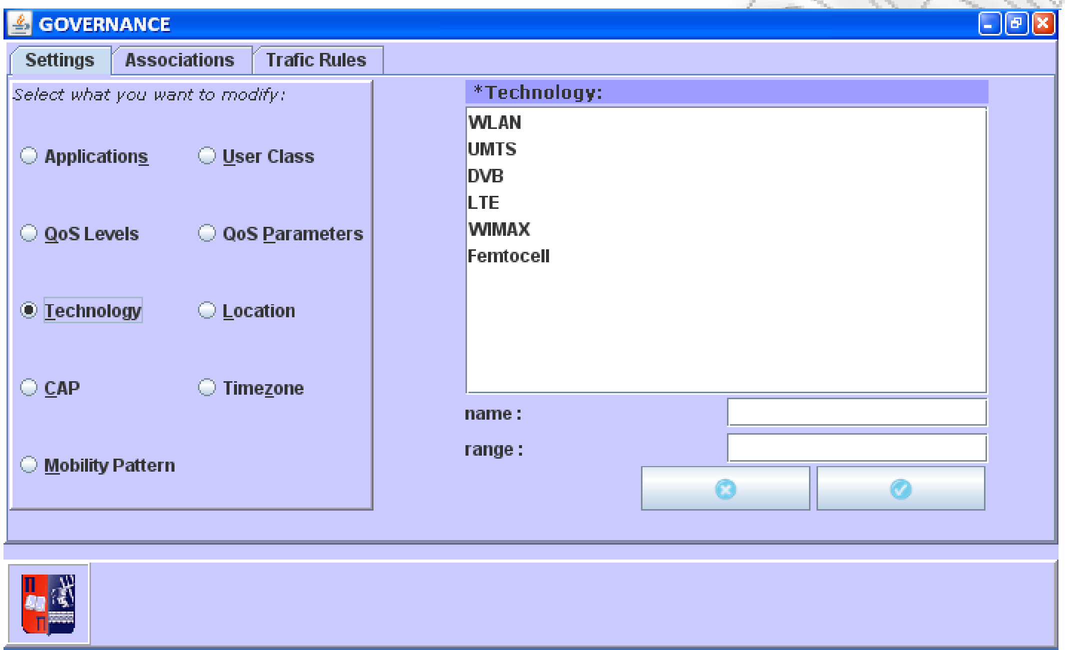


Figure 14: Modification of "technology" parameter

- CAP

A Cognitive Access Point (CAP) covers a portion of the service area and encompasses self-management functionalities. It is essentially a flexible, reconfigurable access point which comprises one or more transceivers capable of operating one or more RATs at various frequency bands, in compliance with the dynamic spectrum allocation concept. That's why its setting is associated with the location parameter (see Figure 15).

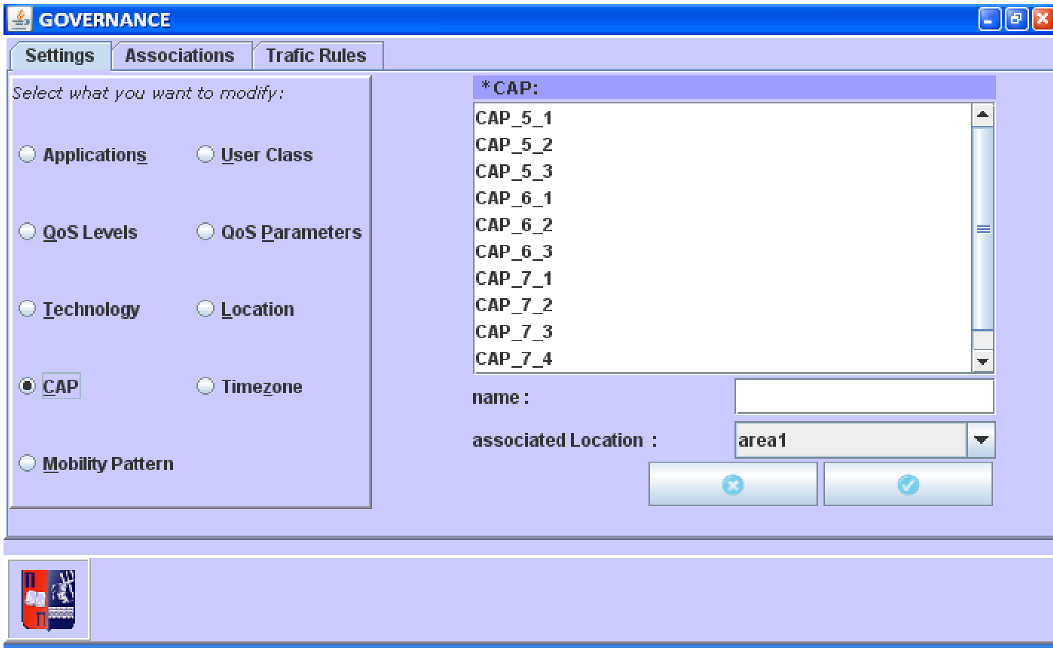


Figure 15: Modification of "CAP" parameter

- Location, timezone and mobility

In a network management a general business level objective is to offer a set of services able to serve multiple types of users, located in different settings (residential, public, or business) and in a variety of time zones. Within the different locations and time zones, a user may adopt various roles during the daily routine.

Seamless mobility hides the complexity of the underlying infrastructure from the user. This feature benefits the user by providing the same type of service, regardless of the equipment used, the current time zone and location.

The modifications of location, timezone and mobility pattern are depicted in Figure 16, Figure 17 and Figure 18 respectively.

Those three parameters are essential for the above mentioned reasons and for the composition of *BussinessLevelEntryNotification* message (its description can be found in section 4.5.1.2) which is one of the business goals of this H2N interface.

This functionality can be found in a second tab in the GUI interface, which is called “Traffic Rules” and it is depicted in Figure 19.

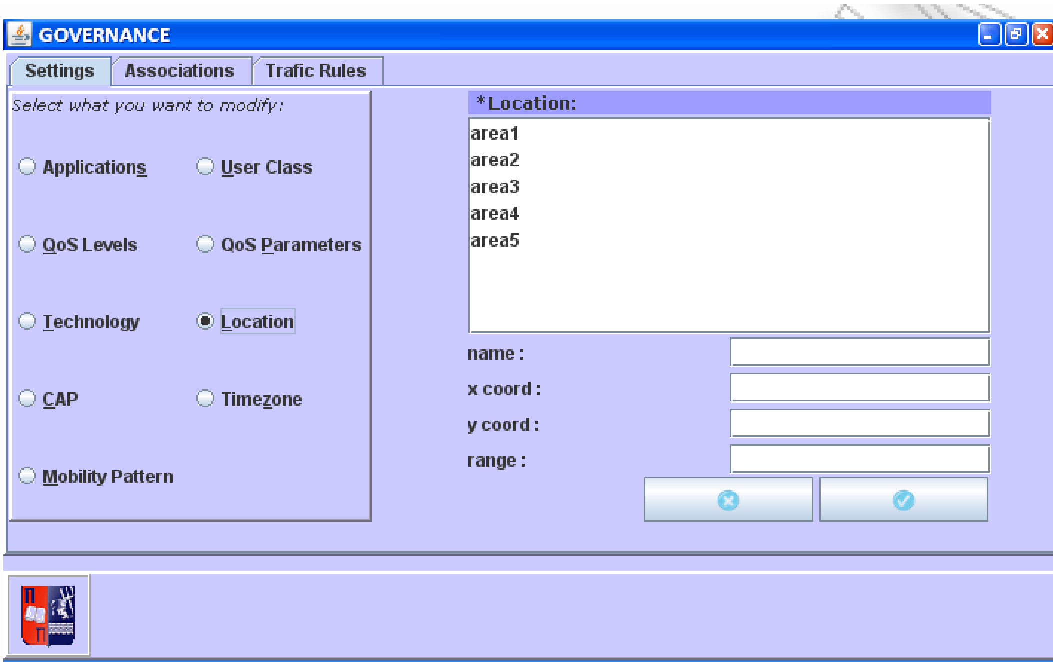


Figure 16: Modification of “location” parameter

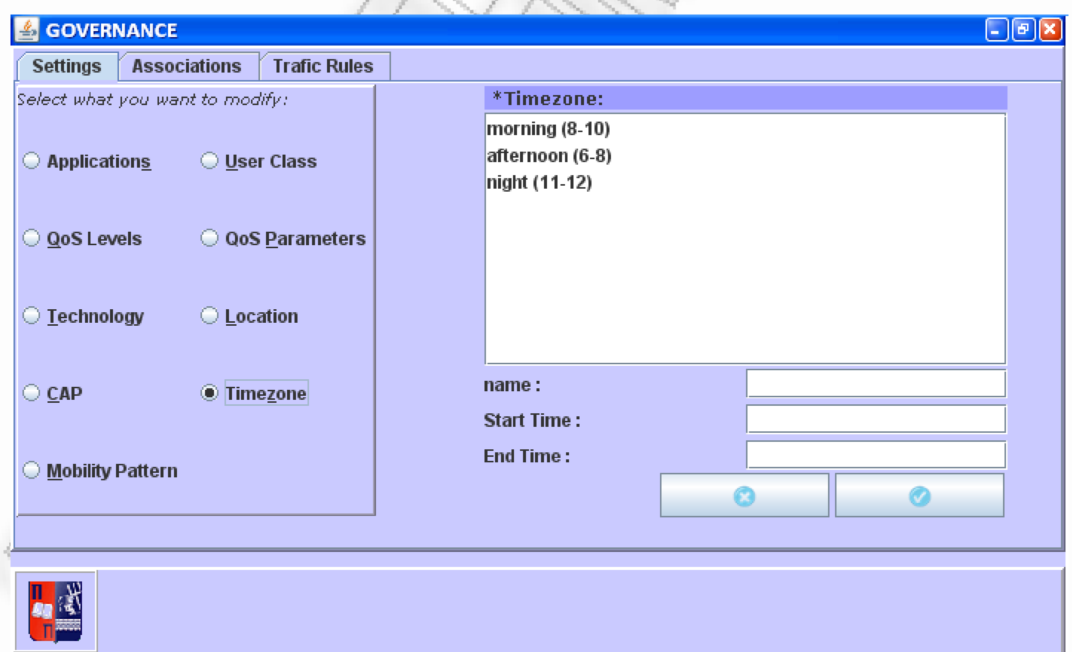


Figure 17: Modification of “timezone” parameter

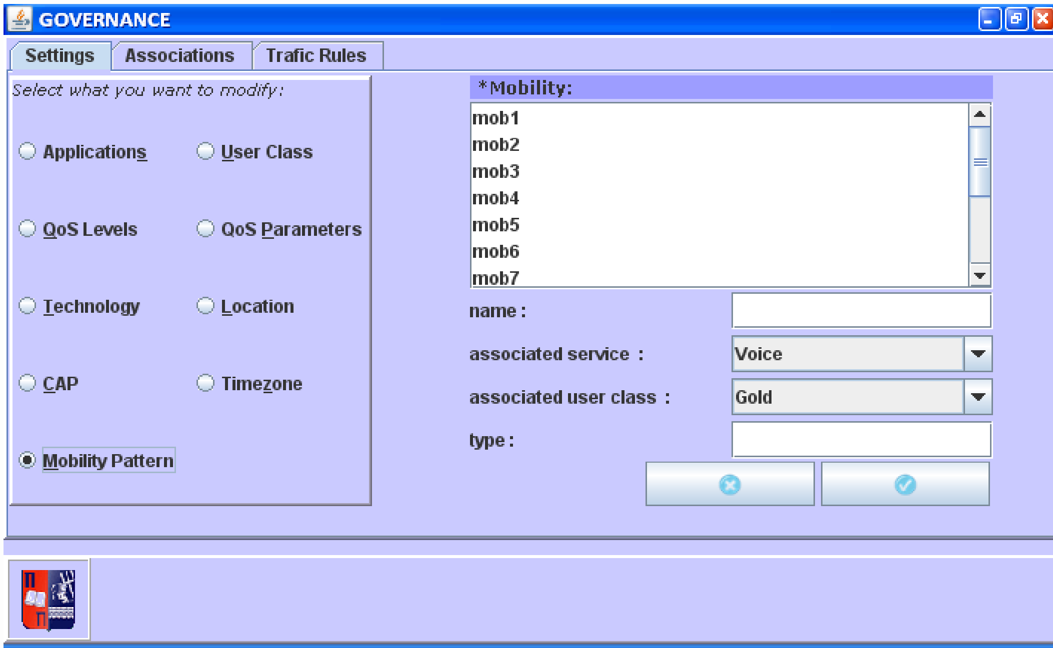


Figure 18: Modification of "mobility pattern" parameter

Figure 19: Configuration of new applications/services in terms of number of users anticipated for a specific user class, in a certain location and time zone

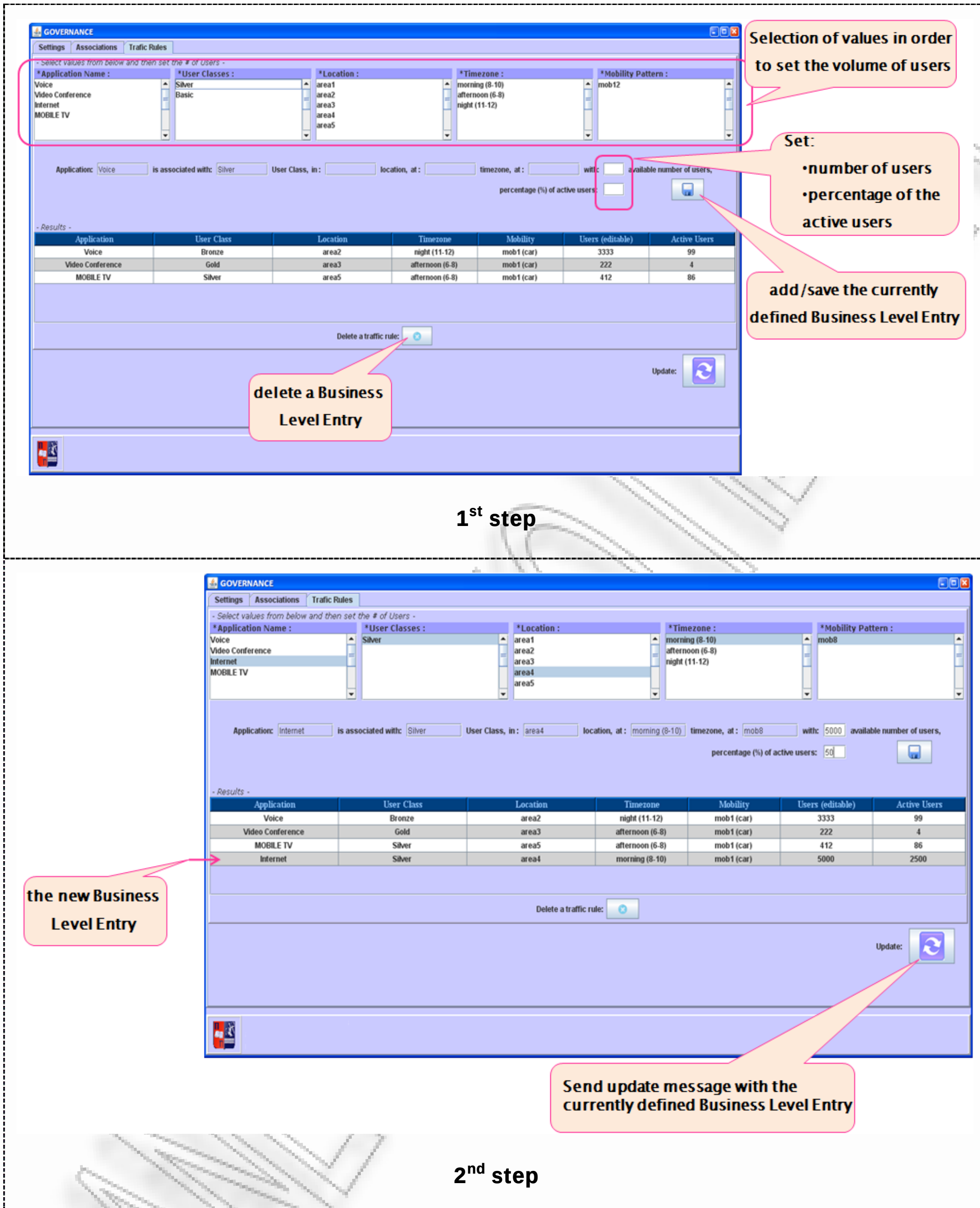


Figure 20: Steps for description of Traffic Rules' tab functionality

Finally, Figure 21 depicts the 3rd tab of the H2N interface in which the associations of the applications and User Classes with one or more Quality Levels are taking place, as well as the composition of *AssociationNotification* message (its description can be found in section 4.5.1.3.). The functionality of this tab is analyzed with two steps in Figure

21. Initially there is depiction of all already defined applications (yellow boxes), then they are associated with user classes (orange boxes) and finally with the QoS levels (purple boxes).

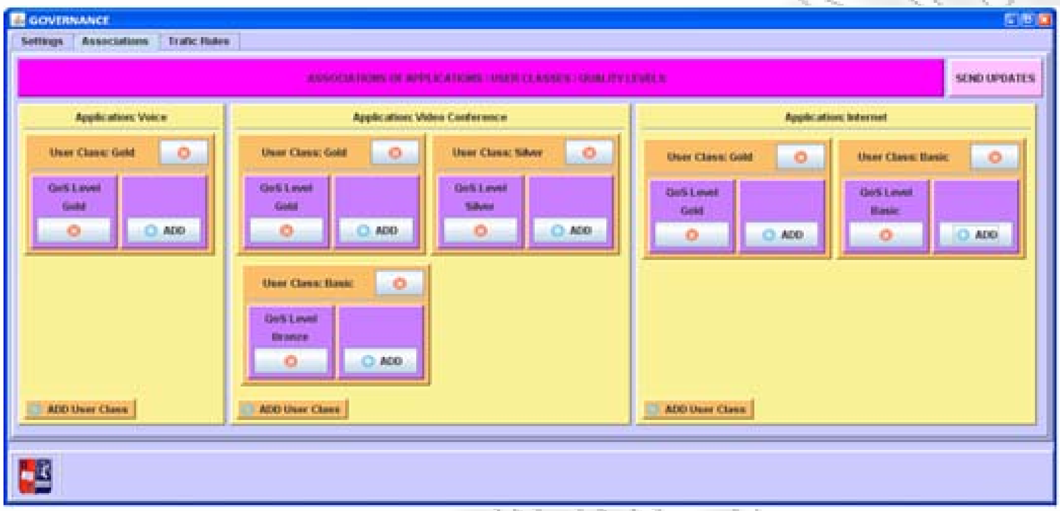
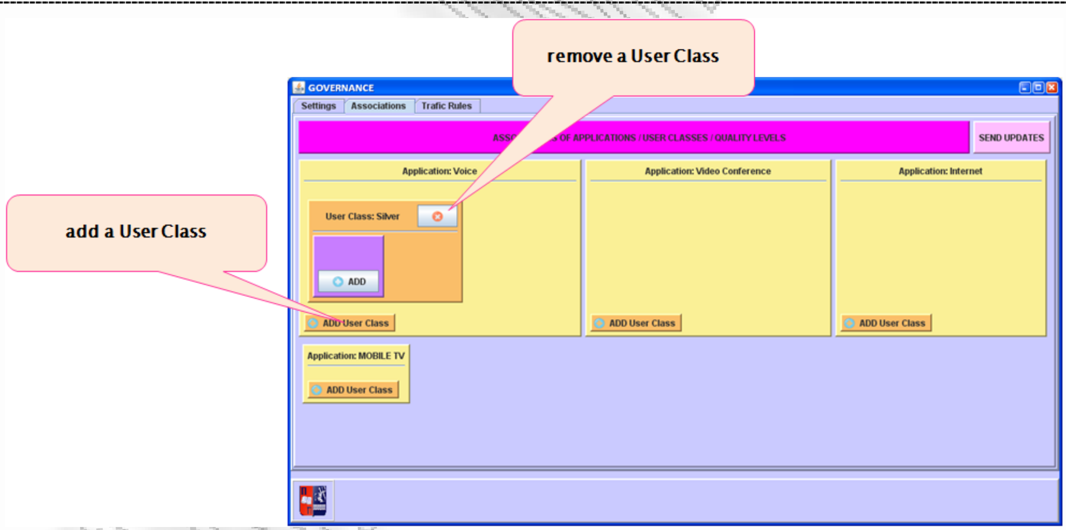
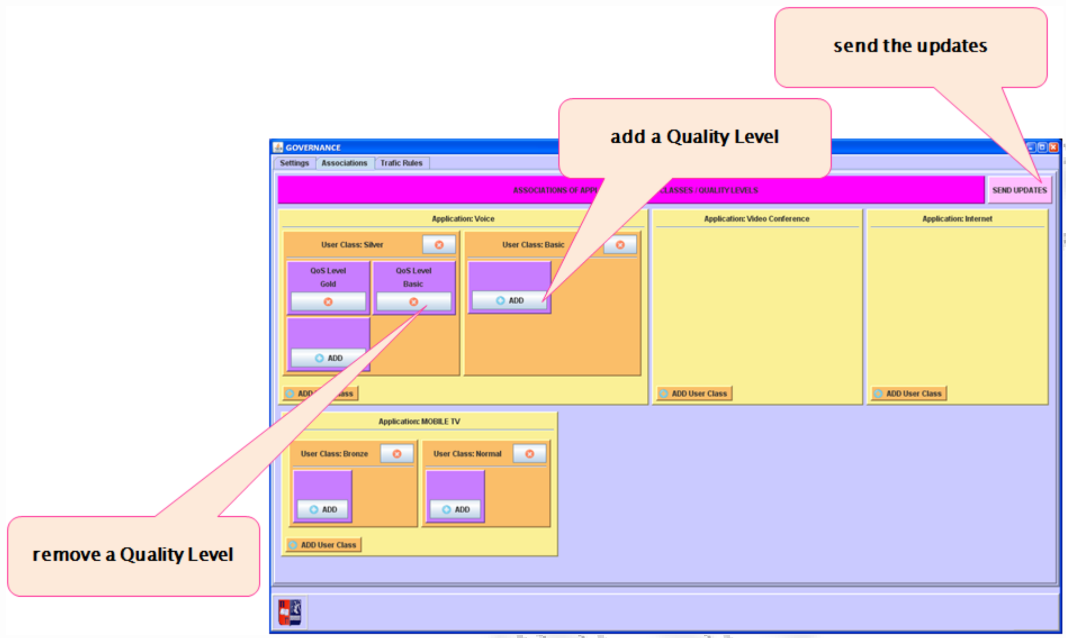


Figure 21: Setting associations



1st step



2nd step

Figure 22: Steps for description of Associations' tab functionality

5.3.1. Scenario Presentation – Legacy Situation

It can be assumed that a network operator wants to introduce a new application, an associated set of user classes for the application, and a set of Quality of Service (QoS) levels for each user class of the application. This introduction can be related to a specific location and time period.

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

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

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

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

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

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

The scenario is splitted into several phases that need to be solved. The first phase of the scenario decoupling is the setting of a business goal and the only phase that we will analyse as it is the part of this thesis. The other phases are the next steps (see section 5.4) that should be followed in order the systems work in completely autonomic mode.

The illustration of the first phase of the scenario is initialized by an expression of request which is a trigger to the operator to define business goals/policies, in high-level terms. Policies are derived according to the higher level goals, to provide constraints and priorities and then are assessed against existing goals/policies so as to identify and resolve conflicts (in fact, conflicts can arise if the defined goal/objective/policy are antagonist with respect to previous goals or the impact of these goals on already deployed applications.

Inputs:

The inputs for this phase are the expression of request in high level-business languages and the policies which are already active.

Outputs:

The outputs of this phase characterize an application *a* that will be offered in a certain location *loc* and time period *tp*. The location l*oc* and time period *tp* are also outputs with the addition of associations of application a with User classes ***UC(a)***, and the QoS levels ***Q(a, u)*** that are associated with the combination of the application *a* and a user class *u*. Then, a volume of users ***N(a, u, loc, tp)*** anticipated for application a and user class *u* in location *loc* and time period *tp* is also extracted under those specifications. As every phase this one should introduce a number of policies that will then initialize the next phase. Consequently, these policies include:

- Relation of a with other applications;
- Relations between user classes;

These high-level objectives/policies need to be further propagated to the network going through an arbitrary set of levels (related to different aspects of the management of a communications network) and be transformed into lower level policies so that reach the element(s) in which to be enforced in terms of low level, technology-specific commands. Consequently, the already set business goals are forwarded to the Context Acquisition functional block in order be manually translated from service requirements into network configuration (technology-specific terms) and leave the system to autonomously work out the situation and meet the objectives.

5.4. Next steps

The H2N interface developed so far provides functionality for specifying high level business objectives which are forwarded to relevant functional blocks (Context Acquisition, Policies). For instance, for the next phase is responsible the Context Acquisition functional block whose target is to analyze the request (made through the H2N interface), and to outline candidate solutions, taking also into account the current context, reusable models (of networks, applications, users) and the knowledge base. The inputs of this phase sum up the outputs of the previous phase which are the business level entries and the policies.

Analysis of request (Figure 23). The business goals introduced in the H2N are in high-level terms. They will be translated to technology-specific terms. The processing of the information sent by the governance functional block and the translation will yield an estimate of the resources that will be required.

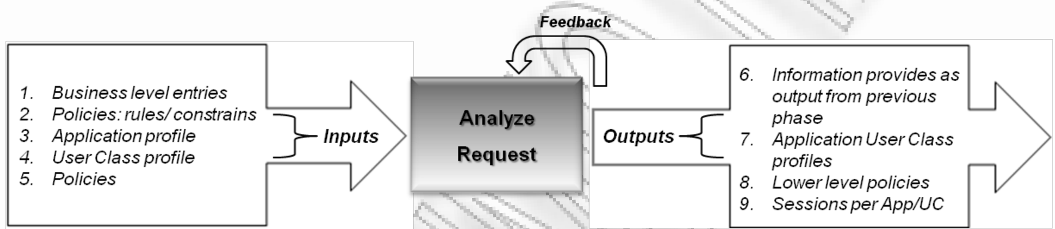


Figure 23: Analyze request

In a future realization there will be higher exploitation of the knowledge base (e.g., updated from offline simulations, or from the extraction and processing of network data).

These high-level objectives/policies need to be further propagated to the network going through an arbitrary set of levels (related to different aspects of the management of a communications network) and be transformed into lower level policies. In other words a translation of service requirements into network requirements is required, which in turn necessitates further elaboration on governance functionalities, the H2N interface and specification of corresponding messages.

6. CONCLUSION

The emergence of Future Internet made the operators unable to face it with the use of management techniques that currently exist. Applying autonomic principles to network management is challenging and at the same time seems to be the only way of reaction.

However, the complex nature of today's systems and applications necessitates the need to automate their management in order to meet operational or behavioural requirements. In fact, the essence of autonomic networking, and thus autonomic communication, is to enable the autonomic network (both the systems and component that make up the network) to adapt the services and resources that they offer in accordance with changing user needs, environmental conditions, and business goals, all the while ensuring that context-sensitive business goals are protected.

Nevertheless, what was extracted from this analysis was that many of the examined initiatives moved their attention towards a real governance of self-managed behaviours aiming at resulting into autonomic networks and systems with the ability dynamically to adapt to changes in accordance with high-level business policies.

Under this evolution human operators set new requirements, called business goals, through management policies. These business goals can be separated in high-level business goals which express the operator and the user performance objectives and low level primitives addressing resources configuration. There is need to bridge the gap between them that will ensure a shift towards the governance of self-managed behaviours.

Thus, our interest was in the development of that autonomic management system in order to enable the efficient operation and utilization of systems and services with little or no human intervention.

H2N interface is part of Governance tool, a functional block that extracts high-level policies, called business goals. The business goals introduced in the H2N refer to applications in a certain location and require certain resources according to the estimates. In order to ensure that system functionality adapts the changing business goals, those goals should be translated in technical low goals and inform all networks segment. That might consist a future step of integration.

On the other hand, this autonomic paradigm seeks to reduce the requirement for human intervention in the management process through the use of one or more control loops that continuously re-configure the system to keep its behaviour within desired bounds.

Nevertheless, the elaboration on the statement “automaticity does not mean independency”, little attention was paid to the novel type of dialogue between a human network operator (HNO) and the self-managed network and in particular to the innovations and peculiarities that most probably the HNO will need to handle while migrating to this new type of management.

The just mentioned factors may emerge a new field of research either on introduction a clearly independent autonomous system or on the discovery of the exact nature of the dialogue between a HNO and the self-managed network.

7. REFERENCES

[1] ...

[2] AKARI architecture design project, National Institute of Information and Communications Technology (NICT), http://akari-project.nict.go.jp

[3] European Future Internet Initiative (EFII), http://initiative.future-internet.eu/

[4] Global Environment for Network Innovations (GENI), National Science Foundation (NSF), http://www.geni.net/

[5] "End-to-End Efficiency" (E3) project, www.ict-e3.eu, European Commission, 7th Framework Programme (FP7) for Research and Technological development, Information and Communication Technologies (ICT)

[6] R. Thomas, D. Friend, L. DaSilva, A. McKenzie, "Cognitive networks: adaptation and learning to achieve end-to-end performance objectives", IEEE Commun. Mag., Vol. 44, No. 12, pp. 51-57, Dec. 2006

[7] J. Kephart, D. Chess, "The vision of autonomic computing", IEEE Computer, Vol. 36, No.1, pp. 41-50, January 2003

[8] P.Demestichas, D.Boscovic, V.Stavroulaki, A.Lee, J.Strassner, "m@ANGEL: autonomic management platform for seamless wireless cognitive connectivity to the mobile Internet", IEEE Commun. Mag., Vol. 44, No.6, pp. 118-127, June 2006

[9] P.Demestichas, G.Dimitrakopoloulos, J.Strassner, D. Bourse, "Introducing reconfigurability and cognitive network concepts in the wireless world", IEEE Vehicular Technology Mag., Vol. 1, No. 1, pp. 33-39, June 2006

[10] P. Demestichas, "Introducing cognitive systems in the wireless B3G world: motivations and basic engineering challenges", Telematics and Informatics journal, Vol. 27, pp. 256-268, 2010

[11] Wireless World Research Forum (WWRF), http://www.wireless-world-research.org/

[12] Institute of Electrical and Electronic Engineers (IEEE), Standards coordinating committee 41 (SCC41), "Dynamic spectrum access networks", http://grouper.ieee.org/groups/scc41/index.html

[13] European Telecommunication Standardization Institute (ETSI), Technical Committee (TC) "Reconfigurable Radio Systems" (RRS), http://www.etsi.org/website/technologies/RRS.aspx

[14] J. Strassner, J. Btrabsner, "Policy-based network management: solution for the next generation", Elsevier Science and Technology Books, Sept. 2003

[15] Hadjiantonis, G. Ellinas, "Towards Autonomic Wireless Networking for the Future Internet", Fourth IEEE International Conference on Self-Adaptive and Self-Organizing Systems Workshop, 2010

[16] Internet Society (ISOC), User Centric Internet, www.isoc.org/pubpolpillar/usercentricity

[17] P. Stuckmann, R. Zimmermann, "European Research on Future Internet Design", IEEE Wireless Communications, October. 2009

[18] Feldmann, "Internet Clean-Slate Design: What and Why?," editorial note submitted to CCR, June 2007; http://ccr.sigcomm.org/online/?q=node/229

[19] D. D. Clark, "The Design Philosophy of the DARPA Internet Protocols," Proc. ACM SIGCOMM, 1998.

[20] Feldmann, "Internet Clean-Slate Design: What and Why?," editorial note submitted to CCR, June 2007; http://ccr.sigcomm.org/online/?q=node/229

[21] P. Papadimitriou, O. Maennel, A. Greenhalgh, A. Feldmann, L. Mathy, "Implementing network virtualization for a Future Internet," unpublished.

[22] D. D. Clark, "The Design Philosophy of the DARPA Internet Protocol", Massachusetts Institute of Technology. Laboratory for Computer Science.

[23] A. Galis, H. Abramowicz, and M. Brunner, "Management and service-aware networking architectures (mana) for future internets," 2009. [Online]. Available: http://www.fp7-moment.eu/

[24] SELFNET, “System deployment scenarios and uses cases for cognitive management of future internet elements,” 2009. Available: http://www.ict-selfnet.eu/

[25] I. P. Chochliouros1, G. Diakonikolaou2, M. Belesioti3, E. Sfakianakis, A.S. Spiliopoulou, N. Alonistioti, “Dynamic Spectrum Reallocation via Autonomic Management”, Research Program Section, Hellenic Telecommunications Organization SA (OTE), Greece 2009.

[26] P. Horn, “Autonomic Computing: IBM’s Perspective on the State of Information Technology”, 2001.

[27] M. Parashar, S. Hariri, “Autonomic computing: an overview”, LNCS, vol. 3566, Springer Verlag, 2005, p. 247–259.

[28] X. Li, H. Kang, P. Harrington, J. Thomas, Autonomic and trusted computing paradigms, in: Autonomic and Trusted Computing, Third International Conference, ATC’06, Wuhan, China, September 3–6, 2006, pp. 143–152.

[29] A. Bandara, E. Lupu, A. Russo, N. Dulay, M. Sloman, et al, Policy refinement for IP differentiated services quality of service management, IEEE e-Transactions on Network and Service Management 2 (2) (2006).

[30] J. Rubio-Loyola, J. Serrat, M. Charalambides, P. Flegkas, G. Pavlou, “A methodological approach toward the refinement problem in policy-based management systems”, Communications Magazine, IEEE 44 (10) (2006) 60–68.

[31] J.A. Lozano, J.M. González, J. Morilla. A Telco approach to autonomic infrastructure management.

[32] Multi-operator telecommunication distribution of service content. http://www.wipo.int/pctdb/en/wo.jsp?IA=SE2005000483&DISPLAY=DESC

[33] (Online) IBM Corporation, Autonomic computing – a manifesto, www.research.ibm.com/autonomic, 2001

[34] Deliverable D4.5, available from http://www.4ward-project.eu/

[35] Autonomic Internet (AutoI) project http:/ist-autoi.eu/

[36] A. Galis, S. Denazis, A. Bassi, A. Berl, A. Fischer, H. de Meer, J. Strassner, S. Davy, D. Macedo, G. Pujolle, J. R. Loyola, J. Serrat, L. Lefevre, A. Cheniour – "Management Architecture and Systems for Future Internet Networks" – in "Towards the Future Internet – A European Research Perspective" –ISBN 978-1-60750-007-0, pp350, April 2009, IOS Press, http://www.iospress.nl/

[37] J. Rubio-Loyola, A. Astorga, J. Serrat, W. K. Chai, L. Mamatas, A. Galis, S. Clayman, A. Cheniour, L. Lefevre, A. Fischer, A. Paler, Y. Al-Hazmi, H. de Meer – "Platforms and Software Systems for an Autonomic Internet" – IEEE Globecom 2010, 6-10 December 2010, Miami, USA; http://www.ieee-globecom.org/

[38] S. Clayman, A. Galis, L. Mamatas; "Monitoring Virtual Networks", 12th IEEE/IFIP Network Operations and Management Symposium (NOMS 2010) - International on Management of the Future Internet (ManFI 2010); 19-23 April 2010, Osaka http://www.man.org/2010/

[39] L. Mamatas, S. Clayman, M. Charalambides, A. Galis and G. Pavlou "Towards an Information Management Overlay for Emerging Networks" - 12th IEEE/IFIP Network Operations and Management Symposium (NOMS 2010); 19-23 April 2010 Osaka, Japan; www.ieee-noms-org

[40] A. Manzalini, & F. Zambonelli, "Towards Autonomic and Situation-Aware Communication Services: the CASCADAS Vision". In: IEEE Workshop on Distributed Intelligent Systems: Collective Intelligence and Its Applications, 2006, pp. 383-388.

[41] A. Manzalini, F. Zambonelli, L. Baresi, & , A. Di Ferdinando, "The CASCADAS Framework for Autonomic Communications". In A. Vasilakos, M. Parashar, S. Karnouskos, & W. Pedrycz (Eds.): Autonomic Communication, Springer Verlag ISBN: 978-0-387-09752-7, pp. 147-168.

[42] A.Manzalini, N. Brgulja, R. Minerva, & C. Moiso, Specification, Development, and Verification of CASCADAS Autonomic Computing and Networking Toolkit. In Cong-Vinh, P. (Ed.): Formal and Practical Aspects of Autonomic Computing and Networking: Specification, Development and Verification (under publication). Hershey, NY: IGI, 2010

[43] E^3 Website. https://ict-e3.eu/

[44] ETSI RRS; Functional Architecture (FA) for the Management and Control of Reconfigurable Radio Systems ETSI TR 102 682 V1.1.1 (2009-07)

[45] K. Nolte, A. Kaloxylos, et al., "The E^3 architecture: Enabling future cellular networks with cognitive and self-x capabilities", To appear in the International Journal of Network Management, 2010

[46] Strassner, John and Agoulmine, N. and Lehtihet, E. "FOCALE: A Novel Autonomic Networking Architecture", In: Latin American Autonomic Computing Symposium (LAACS), Campo Grande, MS, Brazil, 2006

[47] J. Strassner, N. Agoulmine, and E. Lehtihet, "FOCALE: A Novel Autonomic Networking Architecture," In Proceedings of the Latin American Autonomic Computing Symposium (LAACS), Campo Grande, Brazil, July 2006

[48] EC funded- FP7-EFIPSANS Project: http://efipsans.org/

[49] R.Chaparadza et al.: "Creating a viable Evolution Path towards Self-Managing Future Internet via a Standardizable Reference Model for Autonomic Network Engineering", FIA Prague 2009 Conference, published in the Future Internet Book produced by FIA.

[50] Self-Net Project Home Page, https://www.ict-selfnet.eu [Last accessed: 24 Sep 2010]

[51] UniverSelf Milestone MS24. UMF specifications.

[52] ANA Project Homepage, http://www.ana-project.org/ [Last accessed: 25 Sep 2010]

[53] "CIM Core Model White Paper", DMTF, August 2000

[54] "CIM Policy White Paper", DMTF, June 2003

[55] Information Management Datasheet. Available at http://www.tmforum.org/BestPracticesStandards/DatasheetsInformation/4274/Home.html.

[56] A. Andrieux et al., "Web Services Agreement Specification (WS-Agreement)", Version March 2007. Grid Forum Document, GFD.107, Proposed Recommendation, Open Grid Forum, 2007. Available at http://www.ogf.org/documents/GFD.107.pdf.

[57] K. T. Kearney, F. Torelli, C. Kotsokalis, "SLA*: An Abstract Syntax for Service Level Agreements:, Service Level Agreements in Grids Workshop 2010.

[58] C.Bu. Royo, J. Rueda, Ó. L. D. Rugnon, B.Fuentes, A. Castro. Business Terms. Model for a Telecom Operator Business View of SLA. ICE-B 2010

[59] R.Chaparadza, M.Vigeraux, J.Lozano-López, and J.González-Muñoz, "Engineering Future Network Governance", European Journal for the informatics professional 2010.

[60] C.Kotsokalis, U.Winkler, „Translation of Service Level Agreements: A Generic Problem Definition", Workshop on Non-Functional Properties and SLA Management in Service-Oriented Computing .NFPSLAM'09

[61] Gruber, T. R., "Toward Principles for the Design of Ontologies Used for Knowledge Sharing", International Journal Human-Computer Studies, 43(5-6):907-928, 1995.

[62] OWL-S: Ontology Web Language Overview. W3C. http://www.w3.org/TR/owl-features/

[63] SWRL proposal: http://www.w3.org/Submission/SWRL/

[64] Rule Markup Web Site: http://www.ruleml.org/

[65] A. Uszok, J. Bradshaw, R. Jeffers, N. Suri, P. Hayes, M. Breedy, L. Bunch, M. Johnson, S. Kulkarni, J. Lott., "KAoS Policiy and Domain Services: Toward a Description-Logic Approach to Policy Representation, Deconfliction, and Enforcement".

[66] S.Davy, "Harnessing Information Models and Ontologies for Policy Conflict Analysis," PhD. Thesis.

[67] S.Davy, B.Jennings, and J.Strassner, “On harnessing information models and ontologies for policy conflict anaylsis”, In Proc. 11th IFIP/IEEE International Symposium on Integrated Network Management (IM 2009), pages 821–826. IEEE, 2009.

[68] M.Charalambides, P.Flegkas, G.Pavlou, J.Rubio-Loyola, A. K.Bandara, E. C.Lupu, A.Russo, N. Dulay, and M. Sloman, (2009). Policy conflict analysis for diffserv quality of service management. Network and Service Management, IEEE Transactions on, 6(1), pp. 15–30.

[69] S.Davy, K. Barrett, S. Balasubramaniam, Sven van der Meer, B.Jennings, J. Strassner, “Policy-Based Architecture to Enable Autonomic Communications”, A Position Paper

[70] Agent DEvelopment Platform (JADE), Web site: http://jade.tilab.com, accessed June 2010.

[71] F. Bellifemine, G. Caire, T. Trucco and G. Rimassa, “JADE – A FIPA Compliant Agent Framework”, C SELT Internal Technical Report, 1999.

[72] F. Bellifemine, G. Caire, A. Poggi, and G. Rimassa: JADE: A software framework for developing multiagent applications. Lessons learned. Information & Software Technology 50(1-2): 10-21 (2008)

ACRONYMS

ACE	Autonomic Communication Elements
AMS	Autonomic Management Systems
API	Application Programming Interface
ATM	Asynchronous Transfer Mode
AutoI	Autonomic Internet
BSS	Operational Support System
CAP	Cognitive Access Point
CAPEX	Capital Expenditures
CIM	Common Information Model
DARPA	Defense Advanced Research Projects Agency
DAML	DARPA Agent Markup Language
DF	Directory Facilitator
DiffServ	Differentiated services
DMTF	Distributed Management Task Force
DDoS	Distributed Denial of Service
DOC	Distributed Orchestration Component
DSM	Dynamic Spectrum Management
DSNPM	Dynamic Self-organizing Network Planning & Management
E^3	End-to-End Efficiency
FI	Future Internet
FP6	6th Framework Programme
FP7	7th Framework Programme

GANA	Generic Autonomic Network Architecture
GUI	Graphical User Interface
H2N	Human-To-Network
HNO	Human Network Operator
IMS	IP Multimedia Subsystem
INM	In-Network Management
IP	Internet Protocol
IPv4	IP version 4
IPv6	IP version 6
ISOC	Internet Society
ISP	Internet Service Provider
IT	Information Technology
JADE	Java Agent DEvelopment Framework
JEE	Java Platform, Enterprise Edition or Java EE
JME	jMonkey Engine
JSE	Java Platform, Standard Edition
KP	Knowledge Plane
MP	Management Plane
NAT	Network Address Translation
NEs	Network Elements
ONIX	Overlay Network for Information Exchange
OP	Orchestration Plane
OPEX	Operating Expense
OSS	Operational Support Systems

OWL	Web Ontology Language
PBM	Policy Based management
PBNM	Policy Based Network Management
PCRF	Policy and Charging Rules Function
PDP	Policy Decision Point
PIP	Physical Infrastructure Provider
PVP	Policy Verification Point
PXP	Policy Execution Point
QoE	Quality of experience
QoS	Quality of Service
RAT	Radio Access Technologies
RDF	Resource Description Framework
RTE	Real Time Experience
RTI	Real Time Infrastructure
RuleML	Rule Mark-up Language
SDR	Software Defined Radio
SID	Shared Information and Data model
SLA	Service Level Agreement
SP	Service Enablers Plane
SWRL	Semantic Web Rule Language
TCP	Transport Communication Protocol
TMF	TeleManagement Forum
UMF	Unified Management Framework
VNP	Virtual Network Provider

VoIP	Voice over IP
VP	Virtualization Plane
vSPI	Virtualization System Programming Interface
W3C	WWW Consortium
WS	Web Services
XML	Extensible Markup Language

APPENDIX I: CODES

GOVERNANCE.agent.xml

```
<!--
      <H3>GOVERNANCE_Agent</H3>
-->

<agent xmlns="http://jadex.sourceforge.net/jadex"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:schemaLocation="http://jadex.sourceforge.net/jadex
                          http://jadex.sourceforge.net/jadex-
0.96.xsd"
      name="GOVERNANCE"
      package="E3.governance">

      <imports>
            <import>jadex.runtime.*</import>
            <import>jadex.util.*</import>
            <import>jadex.adapter.fipa.*</import>

            <import>E3.ontolo.*</import>
            <import>E3.governance.*</import>
      </imports>

      <beliefs>
            <belief name="gui" class="GOVERNANCEgui"/>

      </beliefs>

      <goals>
            <performgoal name="startGOVERNANCE"/>
      </goals>

      <plans>
            <plan name="init_plan">
                  <body class="InitPlan"/>
                  <trigger>
                        <goal ref="startGOVERNANCE"/>
                  </trigger>
            </plan>

            <plan name="BusinessLevelEntryNotificationMsgPlan">
                  <body
class="BusinessLevelEntryNotificationPlan"/>
                  <trigger>
                        <messageevent
ref="BusinessLevelEntryNotificationMsg"/>
                  </trigger>
            </plan>

            <plan name="BusinessLevelEntryNotificationAckMsgPlan">
                  <body
class="BusinessLevelEntryNotificationAckPlan"/>
                  <trigger>
```

```
                        <messageevent
ref="BusinessLevelEntryNotificationAckMsg"/>
                  </trigger>
            </plan>

            <plan name="AssociationNotificationMsgPlan">
                  <body class="AssociationNotificationPlan"/>
                  <trigger>
                        <messageevent
ref="AssociationNotificationMsg"/>
                  </trigger>
            </plan>

            <plan name="AssociationNotificationAckMsgPlan">
                  <body class="AssociationNotificationAckPlan"/>
                  <trigger>
                        <messageevent
ref="AssociationNotificationAckMsg"/>
                  </trigger>
            </plan>
      </plans>

      <events>
            <messageevent name="BusinessLevelEntryNotificationMsg"
direction="send" type="fipa">
               <parameter    name="performative"    class="String"
direction="fixed">
                        <value>SFipa.REQUEST</value>
               </parameter>
               <parameter      name="language"      class="String"
direction="fixed">
                        <value>SFipa.JAVA_XML</value>
               </parameter>
               <parameter      name="ontology"      class="String"
direction="fixed">
                        <value>E3Ontology.ONTOLOGY_NAME</value>
               </parameter>
               <parameter name="reply-with" class="String">

      <value>SFipa.createUniqueId($scope.getAgentName())</value>
               </parameter>
               <parameter    name="content-class"    class="Class"
direction="fixed">
                        <value>
BusinessLevelEntryNotification.class</value>
               </parameter>
            </messageevent>

            <messageevent
name="BusinessLevelEntryNotificationAckMsg"    direction="receive"
type="fipa">
               <parameter    name="performative"    class="String"
direction="fixed">
                        <value>SFipa.INFORM</value>
               </parameter>
               <parameter      name="language"      class="String"
direction="fixed">
```

```
                        <value>SFipa.JAVA_XML</value>
               </parameter>
               <parameter      name="ontology"      class="String"
direction="fixed">
                        <value>E3Ontology.ONTOLOGY_NAME</value>
               </parameter>
               <parameter name="reply-with" class="String">

      <value>SFipa.createUniqueId($scope.getAgentName())</value>
               </parameter>
               <parameter    name="content-class"    class="Class"
direction="fixed">
                        <value>
BusinessLevelEntryNotificationAck.class</value>
               </parameter>
            </messageevent>

            <messageevent        name="AssociationNotificationMsg"
direction="send" type="fipa">
               <parameter    name="performative"    class="String"
direction="fixed">
                        <value>SFipa.REQUEST</value>
               </parameter>
               <parameter      name="language"      class="String"
direction="fixed">
                        <value>SFipa.JAVA_XML</value>
               </parameter>
               <parameter      name="ontology"      class="String"
direction="fixed">
                        <value>E3Ontology.ONTOLOGY_NAME</value>
               </parameter>
               <parameter name="reply-with" class="String">

      <value>SFipa.createUniqueId($scope.getAgentName())</value>
               </parameter>
               <parameter    name="content-class"    class="Class"
direction="fixed">
                        <value>
AssociationNotification.class</value>
               </parameter>
            </messageevent>

            <messageevent     name="AssociationNotificationAckMsg"
direction="receive" type="fipa">
               <parameter    name="performative"    class="String"
direction="fixed">
                        <value>SFipa.INFORM</value>
               </parameter>
               <parameter      name="language"      class="String"
direction="fixed">
                        <value>SFipa.JAVA_XML</value>
               </parameter>
               <parameter      name="ontology"      class="String"
direction="fixed">
                        <value>E3Ontology.ONTOLOGY_NAME</value>
               </parameter>
               <parameter name="reply-with" class="String">
```

```
      <value>SFipa.createUniqueId($scope.getAgentName())</value>
               </parameter>
               <parameter    name="content-class"    class="Class"
direction="fixed">
                        <value>
AssociationNotificationAck.class</value>
               </parameter>
            </messageevent>

      </events>


      <configurations default="init_state">
            <configuration name="init_state">
                  <goals>
                        <initialgoal ref="startGOVERNANCE" />
                  </goals>
            </configuration>
      </configurations>

</agent>
```

InitPlan.java

```
public class InitPlan extends Plan {

      public void body(){

            try{
                  System.out.println("GOVERNANCE initialization");
                  getBeliefbase().getBelief("gui").setFact(new
GOVERNANCEgui(getExternalAccess()));
                  System.out.println("INIT OK");
            }
            catch(Exception w){
                  ErLogger.log(w,"GOVERNANCE_InitPlan","Check
body!");
            }
      }

}//end of class
```

AssociationNotificationPlan.java

```
public class AssociationNotificationPlan extends Plan {

      public void body(){
            try{
                  ErLogger.logMessage(getScope().getAgentName(),
"AssociationNotification");
```

```
                  System.out.println("GOVERNANCE:      Association
Notification plan started");

                  String               platformStr              =
getExternalAccess().getAgentIdentifier().getName();  //full  agent
name
                  platformStr                                   =
platformStr.substring(platformStr.indexOf("@"));
                  System.out.println("PLATFORM = "+platformStr);

                  IMessageEvent      me      =      (IMessageEvent)
getInitialEvent();
                  AssociationNotification    notificationreq    =
(AssociationNotification) me.getContent();

                  IMessageEvent               mea               =
getExternalAccess().createMessageEvent("AssociationNotificationMsg
");

      mea.getParameterSet(SFipa.RECEIVERS).addValue(new
AgentIdentifier("ContextAcquisition"+platformStr));
                  mea.setContent(notificationreq);
                  ErLogger.logMessage(getScope().getAgentName(),
"AssociationNotification");
                  sendMessage(mea);
                  System.out.println("GOVERNANCE:      Association
Notification forwarded "+platformStr);
            }
            catch(Exception w){

      ErLogger.log(w,"GOVERNANCE_AssociationNotification","Check
body!");
            }
      }
}
```

AssociationNotificationAckPlan.java

```
public class AssociationNotificationAckPlan extends Plan {

      public void body(){
            try{
                  ErLogger.logMessage(getScope().getAgentName(),
"AssociationNotificationAck");
                  System.out.println("GOVERNANCE:          Context
Notification Ack plan started");

                  String               platformStr              =
getExternalAccess().getAgentIdentifier().getName();  //full  agent
name
                  platformStr                                   =
platformStr.substring(platformStr.indexOf("@"));
                  System.out.println("PLATFORM = "+platformStr);
```

```
                  IMessageEvent      me      =      (IMessageEvent)
getInitialEvent();
                  AssociationNotificationAck   notificationresp   =
(AssociationNotificationAck) me.getContent();

                  IMessageEvent              meresp              =
me.createReply("AssociationNotificationAckMsg");

      meresp.getParameterSet(SFipa.RECEIVERS).addValue(new
AgentIdentifier("GOVERNANCE"+platformStr));
                  meresp.setContent(notificationresp);
                  ErLogger.logMessage(getScope().getAgentName(),
"AssociationNotificationAck");
                  sendMessage(meresp);
                  System.out.println("GOVERNANCE:      Association
Notification Ack received "+platformStr);
            }
            catch(Exception w){

      ErLogger.log(w,"GOVERNANCE_AssociationNotification","Check
body!");
            }
```

BusinessLevelEntryNotificationPlan.java

```
public class BusinessLevelEntryNotificationPlan extends Plan {

      public void body(){
            try{
                  ErLogger.logMessage(getScope().getAgentName(),
"BusinessLevelEntryNotification");
                  System.out.println("GOVERNANCE:          Context
Notification plan started");

                  String               platformStr               =
getExternalAccess().getAgentIdentifier().getName();  //full  agent
name
                  platformStr                                    =
platformStr.substring(platformStr.indexOf("@"));
                  System.out.println("PLATFORM = "+platformStr);

                  IMessageEvent      me      =      (IMessageEvent)
getInitialEvent();
                  BusinessLevelEntryNotification notificationreq =
(BusinessLevelEntryNotification) me.getContent();

                  IMessageEvent                mea                 =
getExternalAccess().createMessageEvent("BusinessLevelEntryNotifica
tionMsg");

      mea.getParameterSet(SFipa.RECEIVERS).addValue(new
AgentIdentifier("ContextAcquisition"+platformStr));
                  mea.setContent(notificationreq);
```

```
                  ErLogger.logMessage(getScope().getAgentName(),
"BusinessLevelEntryNotification");
                  sendMessage(mea);
                  System.out.println("GOVERNANCE:  Business  Level
Entry Notification forwarded "+platformStr);
            }
            catch(Exception w){

      ErLogger.log(w,"GOVERNANCE_BusinessLevelEntryNotification","
Check body!");
            }
```

BusinessLevelEntryNotificationAckPlan.java

```
public class BusinessLevelEntryNotificationAckPlan extends Plan {

      public void body(){
            try{
                  ErLogger.logMessage(getScope().getAgentName(),
"BusinessLevelEntryNotificationAck");
                  System.out.println("GOVERNANCE:          Context
Notification Ack plan started");

                  String               platformStr              =
getExternalAccess().getAgentIdentifier().getName();  //full  agent
name
                  platformStr                                   =
platformStr.substring(platformStr.indexOf("@"));
                  System.out.println("PLATFORM = "+platformStr);

                  IMessageEvent      me      =      (IMessageEvent)
getInitialEvent();
                  BusinessLevelEntryNotificationAck
notificationresp       =       (BusinessLevelEntryNotificationAck)
me.getContent();

                  IMessageEvent              meresp             =
me.createReply("BusinessLevelEntryNotificationAckMsg");

      meresp.getParameterSet(SFipa.RECEIVERS).addValue(new
AgentIdentifier("GOVERNANCE"+platformStr));
                  meresp.setContent(notificationresp);
                  ErLogger.logMessage(getScope().getAgentName(),
"BusinessLevelEntryNotificationAck");
                  sendMessage(meresp);
                  System.out.println("GOVERNANCE:  Business  Level
Entry Notification Ack received "+platformStr);
            }
            catch(Exception w){

      ErLogger.log(w,"GOVERNANCE_BusinessLevelEntriesNotification"
,"Check body!");
            }
```

TrfRulesGUI.java

The initialization of messages' exchange starts with a push of a button in each case. For the case of BussinessLevelEntryNotification, the button is called "trfRefreshBtn" and its code can be found in TrfRulesGUI.java file. Particularly,

```
else if(src == trfRefreshBt){
                        businessLevelEntryNotification    =    new
BusinessLevelEntryNotification();

                        pstmt   =   conGOV.prepareStatement("SELECT
trfrules.*,  servdata.*,  ucdata.*,  qualdata.*,  locationdata.*,
timezonedata.*,     mobilitydata.*,     servrules.*,     initsq.*,
parametersdata.*,   initparameters.*   FROM   trfrules,   servdata,
ucdata,   qualdata,   locationdata,   timezonedata,   mobilitydata,
servrules,    initsq,    parametersdata,    initparameters    WHERE
RT_Serv_DBid = Serv_DBid AND RT_UC_DBid = UC_DBid AND RT_Loc_DBid
= Loc_DBid AND RT_Zon_DBid = Zon_DBid AND RT_Mob_DBid = MOB_DBid
AND  Qual_DBid  =  RS_Qual_DBid  AND  RT_UC_DBid  =  RS_UC_DBid  AND
SQ_Qual_DBid  =  RS_Qual_DBid  AND  SQ_Serv_DBid  =  RS_Serv_DBid  AND
SQ_Param_DBid   =   IN_par_DBid    AND   IN_par_DBid   =   PAR_DBid   AND
SQ_ParValue_DBid = IN_DBid ORDER BY RT_DBid");
                        rs = pstmt.executeQuery();
                        while(rs.next()){
                              businessLevelEntry       =       new
BusinessLevelEntry();
                              govLocation = new GovLocation();
                              govTimezone = new GovTimezone();
                              govApplication        =        new
GovApplication();
                              govUserClass = new GovUserClass();
                              govQualityLevel        =        new
GovQualityLevel();
                              govMobility = new GovMobility();
                              govQualityLevelParam      =      new
GovQualityLevelParam();
                        /**application**/
                              int           setServID           =
rs.getInt("Serv_DBid");

      govApplication.setGovApplicationID(setServID);
                              String          setServ           =
rs.getString("Serv_name");

      govApplication.setGovApplicationName(setServ);

                              /**user class**/
                                    int         setUClID        =
rs.getInt("UC_DBid");

      govUserClass.setGovUserClassID(setUClID);
```

```
                                    String      setUClName      =
rs.getString("UC_name");

      govUserClass.setGovUserClassName(setUClName);

                                    /**quality level**/
                                          int      setqualID      =
rs.getInt("RS_Qual_DBid");

      govQualityLevel.setGovQualityLevelID(setqualID);
                                          String   setqualName    =
rs.getString("Qual_name");

      govQualityLevel.setGovQualityLevelName(setqualName);

                                           /**quality        level
parameter**/
                                                int   setparID    =
rs.getInt("PAR_DBid");

      govQualityLevelParam.setParamID(setparID);
                                                String  setparName
= rs.getString("PAR_name");

      govQualityLevelParam.setParamName(setparName);
                                                String setparValue
= rs.getString("IN_par_value");

      govQualityLevelParam.addParamReferenceValue(setparValue);

      //govQualityLevelParam.

      govQualityLevel.addGovQualityLevelParamInfo(govQualityLevelP
aram);

      govUserClass.addGovQualityLevelInfo(govQualityLevel);

                                          /**mobility**/
                                          int      setMobID       =
rs.getInt("MOB_DBid");

      govMobility.setGovMobilityID(setMobID);
                                          String    setMobName    =
rs.getString("MOB_name");

      govMobility.setGovMobilityName(setMobName);
                                          String    setMobType    =
rs.getString("MOB_type");

      govMobility.setGovMobilityType(setMobType);

      govUserClass.setGovMobilityInfo(govMobility);


      govApplication.addGovUserClassInfo(govUserClass);

      businessLevelEntry.setGovApplicationInfo(govApplication);
```

```
                        /**location**/
                              int           setLocID          =
rs.getInt("Loc_DBid");

      govLocation.setGovLocationID(setLocID);
                              String          setLoc          =
rs.getString("Loc_name");

      govLocation.setGovLocationName(setLoc);
                              int setLocx = rs.getInt("Loc_x");

      govLocation.setGovLocationXcoords(setLocx);
                              int setLocy = rs.getInt("Loc_y");

      govLocation.setGovLocationYcoords(setLocy);
                              int        setLocrange          =
rs.getInt("Loc_range");
                              govLocation.setRange(setLocrange);

      businessLevelEntry.setGovLocationInfo(govLocation);
                        /**timezone**/
                              String          setZon          =
rs.getString("Zon_name");
                              String        setZonmin         =
String.valueOf(rs.getInt("Zon_min"));
                              String        setZonmax         =
String.valueOf(rs.getInt("Zon_max"));
                              govTimezone.setTimezoneName(setZon);
                              govTimezone.setStartTime(setZonmin);
                              govTimezone.setEndTime(setZonmax);

      businessLevelEntry.setGovTimezoneInfo(govTimezone);
                        /**users**/
                              int           setUsers          =
rs.getInt("RT_Users");

      businessLevelEntry.setNumberOfUsers(setUsers);
                        /**percent**/
                              int        setPercentage        =
rs.getInt("RT_Users_Active");

      businessLevelEntry.setTrafficPercentage(setPercentage);


      businessLevelEntryNotification.addBusinessLevelEntryInfo(bus
inessLevelEntry);
                        }
                        platformStr                           =
agent.getAgentIdentifier().getName(); //full agent name
                        platformStr                           =
platformStr.substring(platformStr.indexOf("@"));
                        System.out.println("PLATFORM          =
"+platformStr);

                        IMessageEvent            mea          =
agent.createMessageEvent("BusinessLevelEntryNotificationMsg");
```

```
        mea.getParameterSet(SFipa.RECEIVERS).addValue(new
AgentIdentifier("ContextAcquisition"+platformStr));

        mea.setContent(businessLevelEntryNotification);
                        agent.sendMessage(mea);
                        System.out.println("Business  Level  Entry
Notification sent "+platformStr);
                    }
```

AssociationPanel.java

While the “AssociationNotificationMessage” message is initialized by “updateBut” button which is designed in AssociationPanel.java. Particularly,

```
if(src == updateBut){
                        if(assocChanged = true){
                            associationNotification    =    new
AssociationNotification();

                            PreparedStatement      pstmt      =
conGOV.prepareStatement("SELECT servrules.*, servdata.*, ucdata.*,
qualdata.*  FROM  servrules,  servdata,  ucdata,  qualdata  WHERE
RS_Serv_DBid = Serv_DBid AND RS_UC_DBid = UC_DBid AND RS_Qual_DBid
= Qual_DBid ORDER BY RS_DBid");
                            ResultSet rs = pstmt.executeQuery();
                            while(rs.next()){
                                govApplication      =      new
GovApplication();
                                govUserClass       =       new
GovUserClass();
                                govQualityLevel      =     new
GovQualityLevel();
                                govMobility       =       new
GovMobility();
                                govQualityLevelParam   =   new
GovQualityLevelParam();
                            /**application**/
                                int        setServID        =
rs.getInt("Serv_DBid");

        govApplication.setGovApplicationID(setServID);
                                String       setServ        =
rs.getString("Serv_name");

        govApplication.setGovApplicationName(setServ);

                                /**user class**/
                                    int      setUClID      =
rs.getInt("UC_DBid");

        govUserClass.setGovUserClassID(setUClID);
                                    String    setUClName    =
rs.getString("UC_name");
```

```
      govUserClass.setGovUserClassName(setUClName);

                                    /**quality level**/
                                          int   setqualID   =
rs.getInt("RS_Qual_DBid");

      govQualityLevel.setGovQualityLevelID(setqualID);
                                          String setqualName
= rs.getString("Qual_name");

      govQualityLevel.setGovQualityLevelName(setqualName);

                                          /**quality   level
parameter*
                                                int setparID
= rs.getInt("PAR_DBid");

      govQualityLevelParam.setParamID(setparID);
                                                String
setparName = rs.getString("PAR_name");

      govQualityLevelParam.setParamName(setparName);
                                                String
setparValue = String.valueOf(rs.getInt("PAR_value"));

      govQualityLevelParam.addParamReferenceValue(setparValue);

      //govQualityLevelParam.

      govQualityLevel.addGovQualityLevelParamInfo(govQualityLevelP
aram);*/

      govUserClass.addGovQualityLevelInfo(govQualityLevel);

                                          /**mobility*
                                          int   setMobID   =
rs.getInt("MOB_DBid");

      govMobility.setGovMobilityID(setMobID);
                                          String  setMobName
= rs.getString("MOB_name");

      govMobility.setGovMobilityName(setMobName);
                                          String  setMobType
= rs.getString("MOB_type");

      govMobility.setGovMobilityType(setMobType);

      govUserClass.setGovMobilityInfo(govMobility);*/


      govApplication.addGovUserClassInfo(govUserClass);

      associationNotification.addApplicationAssociationInfo(govApp
lication);
                              }
```

```
                              platformStr                          =
agent.getAgentIdentifier().getName(); //full agent name
                              platformStr                          =
platformStr.substring(platformStr.indexOf("@"));
                              System.out.println("PLATFORM         =
"+platformStr);

                              IMessageEvent         mea            =
agent.createMessageEvent("BusinessLevelEntryNotificationMsg");

      mea.getParameterSet(SFipa.RECEIVERS).addValue(new
AgentIdentifier("ContextAcquisition"+platformStr));

      mea.setContent(associationNotification);
                              agent.sendMessage(mea);
                              System.out.println("Association
Notification sent "+platformStr);

                              updateBut.setBackground(Color.GRAY);
                              assocChanged = true;
                        }
                  }
```

GOVERNANCEgui.java

In this file it is clear that Governance is called as agent. This file is also the executable file which every other java file that was used in the integration.

```
public class GOVERNANCEgui extends JFrame {

      IExternalAccess agent;

      public static JTabbedPane jTabbedPane;
      public AssociationPanel associationPanel;

      public GOVERNANCEgui(IExternalAccess _agent){
            try{
                  System.out.println("GOVERNANCE GUI");
                  agent = _agent;

                  CommonUtilities.CUinitialize(1);  //0  ->  no  DB
support, 1 -> with DB support

                  /****************************************    TABS
START ***********************************/
                  jTabbedPane = new JTabbedPane();

      jTabbedPane.setBackground(CommonUtilities.backgrndColor);

      jTabbedPane.setBorder(BorderFactory.createRaisedBevelBorder(
));

                  UIManager um = new UIManager();
```

```
                  um.put("TabbedPane.contentAreaColor",
CommonUtilities.backgrndColor);

                  associationPanel = new AssociationPanel(this);

                  jTabbedPane.addTab("Settings",            new
ConfigGUI(this));
                  jTabbedPane.addTab("Associations",
associationPanel);
                  jTabbedPane.addTab("Trafic      Rules",      new
TrfRulesGUI(this));
                  //jTabbedPane.addTab("Applications/User
Classes/QoS levels Associations", new ServRulesGUI());
                  /**************************************   TABS
END **************************************/

                  /************************************    JFrame
START ************************************/
                  setTitle("GOVERNANCE");
                addWindowListener(new WindowAdapter(  ) {
                  public   void   windowClosing(WindowEvent   we)
{ System.exit(0); }
                });

                Dimension               screenSize              =
Toolkit.getDefaultToolkit().getScreenSize();
                setSize((int)  (screenSize.width  -  getWidth()),
(int) (screenSize.height - getHeight()));


      getContentPane().setBackground(CommonUtilities.backgrndColor
);
                  getContentPane().setLayout(new
BorderLayout(10,10));
                  getContentPane().add(jTabbedPane,
BorderLayout.CENTER);
                  getContentPane().add(new          FooterPanel(),
BorderLayout.SOUTH);

                setVisible(true);
                  /************************************    JFrame
END ***************************************/

                  /*================ loads ================*/
                  ConfigGUI.loadServicesJsp();
                  ConfigGUI.loadUserClassesJsp();
                  ConfigGUI.loadQosLevelsJsp();
                  ConfigGUI.loadParametersJsp();
                  ConfigGUI.loadTecJsp();
                  ConfigGUI.loadLocJsp();
                  ConfigGUI.loadPoAJsp();
                  ConfigGUI.loadZonJsp();
                  ConfigGUI.loadMobJsp();
                  /*============== END loads ==============*/
            }
            catch(Exception w){
```

```
                  ErLogger.log(w,"GOVERNANCEgui","Check
constructor");
            }
      }//end of "GOVERNANCEgui"
```

APPENDIX II: SUMMARY IN GREEK LANGUAGE

1. ΕΣΑΓΩΓΗ

Η τελευταία δεκαετία χαρακτηρίστηκε από σημαντικές εξελίξεις σε δικτυακές συσκευές, πρωτόκολλα επικοινωνίας και υπηρεσίες. Αυτή η συνεχής εξέλιξη των δικτύων επικοινωνίας έχει οδηγήσει σε περίπλοκα, δυναμικά και ανομοιογενή δίκτυα που έχουν εξελιχθεί σε παγκοσμίως πλούσια υποδομή υπηρεσιών επικοινωνίας. Τα δεδομένα και τα τηλεπικοινωνιακά δίκτυα και οι υπηρεσίες που προσφέρουν έχουν αυξηθεί σημαντικά σε μέγεθος, γεγονός σε συνδυασμό των παραπάνω χαρακτηριστικών καθιστούν δυσχερές, αν όχι αδύνατο? για να αντιμετωπιστούν αποτελεσματικά και να λειτουργήσουν μέσα σε επιθυμητά όρια με τη χρήση των παραδοσιακών προσεγγίσεων, τα οποία βασίζονται σχετικά με στη ανθρώπινη παρακολούθηση και παρέμβαση.

Επιπλέον, η εξέλιξη του μελλοντικού διαδικτύου (FI) ([2], [3], [4]) εισάγει την ελεύθερη ανταλλαγή πληροφοριών μεταξύ των επιχειρηματικών εφαρμογών και των συστημάτων δικτύωσης το οποίο προωθεί η εξατομίκευση των υπηρεσιών και επιτρέπει σε πολλούς διαφορετικούς τύπους τελικών εφαρμογών και διαχειρίσεων να βελτιστοποιούν την απόδοση του δικτύου. Αποτέλεσμα αυτής της ελεύθερης ανταλλαγής πληροφοριών, είναι η ανάγκη για διευκόλυνση της σύστασης των πληροφοριών μεταξύ αυτών των εφαρμογών, η εναρμόνιση των διαφορών μεταξύ των λειτουργιών, η διαχείριση των δεδομένων και των μοντέλων πληροφοριών σε ετερογενή δίκτυα, καθώς και εισαγωγή συστημάτων διαχείρισης της εφαρμογής.

Από την άλλη πλευρά, οι φορείς θα έχουν να αντιμετωπίσουν επιπλέον πολυπλοκότητα συγκριτικά με σήμερα, στη διαχείριση των δικτύων και των υπηρεσιών.

Στο αυτό το πλαίσιο, οι ερευνητές και οι επαγγελματίες που αποσκοπούν στην αυτόνομη διαχείρισης του δικτύου, που με την εισαγωγή ικανοτήτων αυτο-ρύθμισης της συμπεριφοράς των φορέων του δικτύου φαίνεται να είναι η πλέον ενδεδειγμένη στην παρούσα κατάσταση.

Παρά το γεγονός ότι, οι παρόν υπηρεσίες των φορέων των δικτύων αντιμετωπίζουν τεχνολογικά όρια μεταξύ των ασύρματων και ενσύρματων τομέων, χρήζουν επίσης την ανάγκη της (αυτο)-διαχείρισης εξίσου όπως και τα συστήματα υποστήριξής τους.

Τα γνωσιακά/ αυτόνομα διαχειριστικά συστήματα ([5]-[10], παρουσιάστηκαν επίσης στο WG6 [11]), φαίνεται πως είναι πολύτιμη διέξοδο για τα ο Διαδίκτυο του Μέλλοντος.

Ο βασικός παράγοντας της επιτυχίας είναι η υλοποίηση ενός Unified Management Framework (UMF) για τις διάφορες υπάρχουσες επαναστατικές αρχιτεκτονικές, ή για τις αυτόνομες αρχιτεκτονικές που σχεδιάστηκαν σε παλαιότερες έρευνες. Αυτό το ενιαίο πλαίσιο πρέπει να αποτελέσει μια τεχνολογία που να αφορά (ασύρματα και ενσύρματα) συστήματα και υπηρεσίες καλύπτοντας τα όποια κενά στην τεχνολογία ή στα πρότυπα. Αυτό θα μπορούσε να αξιοποιηθεί τόσο για την έρευνα που γίνεται στην αυτόνομη δικτύωσης όσο και για να αποδείξει την εφαρμοσιμότητά του σε βιομηχανικά πρότυπα.

Αυτό το πλαίσιο αρχικά θα πρέπει πρώτα να διασφαλίσει ότι διαφορετικά πολύπλοκα διαχειριστικά συστήματα που εφαρμόζονται σε διαφορετικές αυτόνομες αρχιτεκτονικές θα μπορούν να συνεργαστούν. Δεύτερον, θα πρέπει να εξασφαλίσει αυτόνομες λειτουργίες να μπορούν να ξεχωριστά ανάλογα με την εκάστοτε επιλεγμένη αρχιτεκτονική του διαχειριστικού συστήματος.

Τέλος, αυτό το πλαίσιο θα πρέπει α εξασφαλίζει κάποιες κοινές διαδικασίες που να μπορούν να χρησιμοποιούνται σε όλα τα τεχνολογικά επίπεδα κι από όλους τους φορείς των επικοινωνιών, λειτουργίες όπως η διαχείριση σφαλμάτων, η βελτιστοποίηση των πόρων κα, ανάλογα των επιχειρηματικών στόχων και των ισχυουσών πολιτικών διακυβέρνησης των επιμέρους τομέων.

Ο στόχος αυτός θα πρέπει να έχει ένα σκέλος που είναι μεταξύ του φορέα και των συστημάτων, παρέχοντας στον πρώτο αρκετή γνώση τέτοια ώστε να ελέγχει πάντα το τελευταίο, και να είναι σε θέση να αναλάβει τις αυτόνομες λειτουργίες.

Μια πολύτιμη διαδικασία αυτού του πλαισίου είναι η υιοθέτηση μεθόδων διακυβέρνησης για να καθοδηγούν τις συμπεριφορές των υποδομών και να προσφέρουν μια εικόνα των υπηρεσιών. Όλο αυτό προσφέρει στους φορείς των επικοινωνιών δυνατότητες διακυβέρνησης το οποίο αποτελεί ένα από τις απαιτήσεις της εγκαθίδρυσης των αυτόνομων διαχειριστικών συστημάτων. Αυτό σημαίνει ότι οι διαχειριστές επικοινωνιών μπορούν να δηλώσουν τις επιθυμίες ή τις ενστάσεις τους ώστε στη συνέχεια να βγουν κάποιοι μηχανισμοί ή κανόνες. Γενικά αυτό επιτυγχάνεται μέσων των πολιτικών ([14]).

Αυτή η διπλωματική εργασία επικεντρώνεται ακριβώς σε αυτή την κατεύθυνση. Η διακυβέρνηση θα πρέπει να δρα με το ρόλο μιας υπηρεσίας που να διαμορφώνει αυτές τις πολιτικές. Συγκεκριμένα, η κεντρική ιδέα είναι ότι ο φορέας θα ενεργοποιείται ώστε να θέσει επιχειρηματικούς στόχους διαμορφωμένους σε υψηλό επίπεδο. Οι πολιτικές θα πρέπει να βγαίνουν με βάση αυτούς τους επιχειρησιακούς στόχους και κατά συνέπεια να υποβάλει περιορισμούς και προτεραιότητες ώστε να επιλύει τις όποιες συγκρούσεις εμφανίζονται.

Στα ήδη υπάρχοντα συστήματα δεν υπάρχει μια τέτοια διεπαφή η οποία να εισάγει επιχειρηματικούς στόχους διαμορφωμένους σε υψηλό επίπεδο και να επιτρέπει στο σύστημα να λειτουργεί αυτόνομα ώστε να αντιμετωπίσει κάθε απαίτηση. Η υλοποίηση αυτής της διεπαφής της οποίας η λειτουργικότητα να χαρακτηρίζεται ως διακυβέρνηση αυτόνομης διαχείρισης είναι ένα ακόμα μέρος αυτής της εργασίας.

Πιο αναλυτικά παρακάτω ακολουθεί η δομή αυτή της εργασίας. Μια σύντομη ανάλυση για το Διαδίκτυο του μέλλοντος και τα χαρακτηριστικά του παρουσιάζεται στην ενότητα 2 ενώ στην 3η ενότητα μπορούμε να βρούμε παρουσίαση αυτόνομων συστημάτων διαχείρισης αυτού. Στη συνέχεια και στην ενότητα 4 υπάρχει η αναλυτική παρουσίαση της διακυβέρνησης της αυτόματης διαχείρισης ώστε να μας οδηγήσει στην ενότητα 5 και στην ανάλυση της H2N διεπαφής ως μέρους του διακυβερνητικού εργαλείου. Αυτή η διπλωματική εργασία κλείνει στην ενότητα 6 όπου συνοψίζει τα βασικά σημεία αλλά και προτείνει ιδέες για περαιτέρω μελλοντική ανάπτυξη. Τέλος, στο τέλος αυτής εργασίας υπάρχουν δύο παραρτήματα τα APPENDIX I: και APPENDIX II: , στα οποία κανείς μπορεί να βρει επιλεγμένα κομμάτια του κώδικα της εφαρμογής καθώς και μια συνοπτική περίληψη της εργασίας στα ελληνικά, αντιστοίχως.

2. ΔΙΑΔΊΚΤΥΟ ΤΟΥ ΜΈΛΛΟΝΤΟΣ

2.1. Εισαγωγή

Το Διαδίκτυο του μέλλοντος (FI) αποτελεί κέντρο ενδιαφέροντος και ερευνών όσο αναφορά τα δίκτυα και τις υπηρεσίες. Ο βασικός λόγος εναπόκειται στο ότι το διαδίκτυο που σχεδιάστηκε στα 1970 δεν μπορεί να υποστηρίξει τα σημερινά επικοινωνιακά και υπολογιστικά συστήματα. Συγκεκριμένα, δεν είχε σχεδιαστεί ώστε να εξυπηρετεί ποικίλους και ραγδαία αναπτυσσόμενους αριθμούς χρηστών ασύρματων και μη, επιχειρηματικών μοντέλων , εφαρμογών, μηχανισμών και δικτύων. Αυτή οι κατασκευαστικοί περιορισμοί είναι ευδιάκριτοι παγκοσμίως.

Στο μελλοντικό Διαδίκτυο η πρόσβαση θα είναι διαθέσιμη από παντού και η συνδεσιμότητα θα καταστεί θεμελιώδης υπηρεσία που όλες οι κοινότητες που την χρησιμοποιούνται θα πρέπει να στηρίζονται σε αυτή. Το FI θα προσφέρει δυο κατευθύνσεις: μία που θα επικεντρώνεται στις ανάγκες συγκεκριμένων χρηστών και μία που θα βασίζει το ενδιαφέρον της στη σύσταση του δικτύου.

Στη συνέχεια, θα παρουσιαστούν τα χαρακτηριστικά τού παρόν διαδικτύου σε αντιδιαστολή με αυτά του μελλοντικού, αλλά και οι προδιαγραφές και οι αλλαγές που θα προκύψουν.

2.2. Τεχνικοί περιορισμοί του σημερινού Διαδικτύου

Το παρόν διαδίκτυο χαρακτηρίζεται από σχεδιαστικές αρχές όπως: η μετάδοση πακέτων, ένα δίκτυο απαρτιζόμενο από άλλα συνεργαζόμενα δίκτυα, ευφυή συστήματα κα. Σήμερα υπάρχει μόνο μία απλή σύνδεση μεταξύ του δικτύου και του διαδικτύου το οποίο είναι και η αιτία για την αδυναμία διασύνδεσης συνομιλούντων με μεγάλο φάσμα.

Παράλληλα η απώλεια ελαστικότητας των κόμβων γίνεται αισθητή με την ανάπτυξη της τάξης μεγέθους του Διαδικτύου. Το μέγεθος και το πεδίο εφαρμογής του σημερινού Διαδικτύου, καθιστούν την ανάπτυξη νέων τεχνολογιών δικτύου αρκετά δύσκολη, από άποψη αυξανόμενης ζήτησης ικανότητα συνδεσιμότητας. Καινοτομία έχει παρουσιαστεί μόνο στις εφαρμογές και στις τηλεπικοινωνίας μετάδοσης παρά στον πυρήνα της τεχνολογίας, και στα στρώματα του δικτύου και των μεταφορών TCP / IP. Οι ακόλουθοι τεχνολογικοί περιορισμοί πρέπει να ξεπεραστούν για να αντιμετωπίσουν τις μελλοντικές προκλήσεις για την παγκόσμια δίκτυο επικοινωνίας [17]:

- Περιορισμένη Υποστήριξη των ασύρματων τερματικών.
- Έλλειψη ενσωματωμένης ασφάλειας
- Ζητήματα κλιμάκωση
- Προκλήσεις επίδοσης και ποιότητας υπηρεσιών.

2.3. Προδιαγραφές κι αλλαγές εξαιτίας του Μελλοντικού Διαδικτύου

Βασικά στόχος του μελλοντικού διαδικτύου είναι να αναπτύξει τεχνολογίες και αρχιτεκτονικές που να ανταποκρίνονται τόσο στο ασύρματο όσο και στο ενσύρματο δίκτυο. Οι παράγοντες που θα συντελέσουν σε αυτό περιγράφονται ακολούθως.

2.3.1. Απαιτήσεις των εφαρμογών για το FI

Εκτός από την αύξηση των χρηστών παρατηρείται και αύξηση των υπηρεσιών. Το μελλοντικό Διαδίκτυο ανταποκρίνεται σε αυτό με:

- Προσφορά υψηλής ποιότητας και διάδοσης του περιεχομένου τους
- Διασύνδεση όλων των αντικειμένων και πραγμάτων
- Service- Oriented διαδίκτυο

2.3.2. Το Μελλοντικό Διαδίκτυο αντιμετωπίζει τις τρέχουσες μεγάλες δυσχέρειες

Επειδή το παρόν διαδίκτυο αρχικά σχεδιάστηκε για «αμετακίνητα» τερματικά, σήμερα αντιμετωπίζει πρόβλημα στο να αντιμετωπίσει της μετακίνησης. Έτσι θα πρέπει να αναπτυχθούν νέα συστήματα διαχείρισης της κινητικότητας τα οποία θα έχουν χαρακτηριστικά όπως αυτά που φαίνονται παρακάτω.

- Ευελιξία
- Ασφάλεια
- Κινητικότητα
- Διαχειριστικότητα

2.3.3. Απαιτήσεις για την ασύρματη δικτύωση

Το δίκτυο του μέλλοντος πιστεύεται ότι θα εξυπηρετεί εκατομμύρια ασύρματους χρήστες μέσω ασύρματων συσκευών. Κατά συνέπεια, προκύπτουν ζητήματα του μεγέθους της διασύνδεσης. Για να αντιμετωπιστεί ορθά τα μελλοντικά ασύρματα δίκτυα θα πρέπει να πληρούν χαρακτηριστικά όπως:

5. Επεκτασιμότητα στο να δέχονται ανυπολόγιστο αριθμό ασύρματων χρηστών και σε ακραίες πυκνότητες.
6. Μετρητές πολυπλοκότητας και ετερογένειας.
7. Δυνατότητα γρήγορης υιοθέτησης της αλλαγής.
8. Διατήρηση χρήστο-κεντρικών χαρακτηριστικών.

3. ΑΥΤΟΝΟΜΗ ΔΙΑΧΕΙΡΙΣΗ ΤΟΥ ΜΕΛΛΟΝΤΙΚΟΥ ΔΙΑΔΙΚΤΥΟΥ

3.1. Εισαγωγή

Η διαχειριστική λειτουργικότητα που θα παρουσιάζει το μελλοντικό διαδίκτυο θα βασίζει την επιχειρηματικότητα του στην προσφερόμενη υπηρεσία. Αυτές οι προσφερόμενες υπηρεσίες οδηγούν στην παροχή καλύτερη ποιότητα υπηρεσίας, όπου αυτό απαιτείται.

Εξαιτίας αυτής της πρόκλησης, το διαδίκτυο του μέλλοντος φαίνεται να υιοθετεί συστήματα αυτοδιαχείρισης [23]. Το πρόβλημα όμως είναι ότι, όπως οι επιχειρηματικές εφαρμογές υπηρεσίες που καλύπτουν τους οργανισμούς έχουν διαδοθεί, έτσι έχουν αυξηθεί και τα προβλήματα με την διαχείριση αυτών των αιτήσεων και των υπηρεσιών σε πολλούς τομείς της διαχείρισης χρήση ετερογενών τεχνολογιών διαχείρισης.

Μέσω της αυτόνομης διαχείρισης το διαδίκτυο του μέλλοντος επιδιώκει ότι θα αντιμετωπίσει αυτά τα προβλήματα. Επιδιώκει ότι θα να χρησιμοποιούν τεχνικές για να καταστεί δυνατή η αποτελεσματική λειτουργία και τη χρησιμοποίηση των συστημάτων και υπηρεσιών με μικρή ή και μηδαμινή ανθρώπινη παρέμβαση. Τα εν λόγω αυτόνομα συστήματα διαχείρισης αναμένεται να λειτουργήσει μέσα σε ένα σύνολο περιορισμών ή προϋποθέσεων, δηλαδή, επιχειρησιακών ή συμπεριφορικών απαιτήσεων. Όπως θα δούμε παρακάτω αυτές οι απαιτήσεις είναι δυνατόν να καθοριστούν είτε από τους διαχειριστές του συστήματος ή οριστούν ως μέρη συμφωνιών.

3.2. Αυτόνομες και γνωσιακές λειτουργίες που βασίζονται στο Μελλοντικό Διαδίκτυο

Οι διαδικασίες που χαρακτηρίζουν την αυτόνομη διαχείριση του μελλοντικού διαδικτύου είναι οι εξείς:

- Αυτοδιαχείριση (Self-management)
- Αυτό- ευαισθητοποίησης (Self-awareness)
- Αυτό-βελτιστοποίηση (Self-optimization)
- Αυτο-ρύθμιση (Self-configuration)
- Αυτό ίαση (Self-healing)
- Αυτοπροστασία (Self-protection)
- Αυτό-οργάνωση (Self-organisation)

3.3. Οι πολιτικές στην αυτόνομη διαχείριση του Μελλοντικού Διαδικτύου

Η αυτόνομη διαχείριση στο διαδίκτυο του μέλλοντος προσπαθεί να διασφαλίσει ότι τα συστήματα και οι υπηρεσίες θα λειτουργούν όπως αναμένεται. Χρήση πολιτικών μπορούν να χρησιμοποιηθούν ώστε να διασφαλίσουν την επιθυμητή συμπεριφορά των στην συστημάτων και των υπηρεσιών κι έτσι να αποσκοπήσουν με αυτή τους τη λειτουργία να χαρακτηρίσουν τα αυτόνομα διαχειριστικά συστήματα με τα απαιτούμε λειτουργικά χαρακτηριστικά. Τέτοιες πολιτικές μπορούν να ανακτηθούν από το ”capturing” και το “mapping” αυτόνομων διαχειριστικών δράσεων.

Η διαχείριση που βασίζεται στις πολιτικές είναι ένα ευρέως διαδιδόμενο παράδειγμα διαχείρισης το οποίο απαιτεί πολλές προόδους στην τεχνολογία, κυρίως όσον αφορά:

- Διαχείριση των ετερογενών λειτουργικοτήτων

- Προσαρμοστικότητα
- Μάθηση των εφαρμογών και συλλογιστικών τεχνικών για την στήριξη της ευφυής αλληλεπίδρασης

3.4. Η ανθρώπινη παρέμβαση στη διαχείριση του Μελλοντικού Διαδικτύου

Ένα κεντρικό πρόβλημα στη διαχείριση του μελλοντικού διαδικτύου είναι η ανθρώπινη παρέμβαση ([26], [27]). Το ζήτημα είναι να μειωθεί ο ανθρώπινη μεσολάβηση όσο το δυνατόν περισσότερο. Οι τεχνικές αυτοδιαχείρισης που αναφέρθηκαν προηγουμένως αναπτύσσουν μια προσέγγιση κατά την αναπτύσσονται διαχειριστικές δυνατότητες στα αυτόνομα συστήματα που προβλέπουν τις απαιτήσεις κι έτσι λύνεται το πρόβλημα της ανθρώπινης παρέμβασης [28].

3.5. Σχεδιαστικά ζητήματα για την υλοποίηση της αυτόνομης διαχείρισης του Μελλοντικού Διαδικτύου

Η υλοποίησης της αυτοδιαχείρισης για το μελλοντικό διαδίκτυο αποτελεί μια πολύτιμη αλυσίδα που απεικονίζει τον τρόπο τους στόχους της επιχείρησης συσχετίζονται με την υποδομή. Ως αποτέλεσμα, κι εδώ παρουσιάζονται τυπικά πια προβλήματα όπως τα ακόλουθα.

- Ευελιξία
- Συντονισμός
- Παροχή πόρων:
- Πεδίο εφαρμογής
- Χωρητικότητα φορτίου σε ώρα αιχμής:
- Τοπική Βελτιστοποίηση

- Διασκορπισμένοι πόροι
- Συνεργασία
- Φυσική μετακίνηση
- Διαχείριση των ακραίων σημείων
- Ροή εργασίας

4. Η ΔΙΑΚΥΒΕΡΝΗΣΗ ΤΗΣ ΑΥΤΟΝΟΜΗΣ ΔΙΑΧΕΙΡΙΣΗΣ

4.1. Εισαγωγή

Για να γίνουν τα δίκτυα αυτόνομα θα πρέπει να προσδιοριστούν διαχειριστικές απαιτήσεις σε υψηλό επίπεδο από τους διαχειριστές. Η αυτόνομη υποδομή υπονοεί ένα μεγάλο βήμα στις αυτοματοποιημένες διαδικασίες των δικτύων.

Η διακυβέρνηση έχει εμφανιστεί σαν όρος για να περιγράψει αυτόν τον νέο τρόπο (τεχνικές και λειτουργίες) για να πραγματοποιηθεί η διαχείριση σε αυτόνομους ρυθμούς αποφεύγοντας να κολλήσει στις κλασικές ταχτικές διαχείρισης των δικτύων.

Η δικτυακή διακυβέρνηση είναι σχεδόν πάντα συνδεδεμένη με τις πολιτικές. Συνήθως, οι επιχειρηματικές πολιτικές σε υψηλότερα επίπεδα ορίζονται και σε εκείνο το επίπεδο. Στη συνέχεια οι πολιτικές διοχετεύονται στο δίκτυο σε όλα τα επίπεδά του όπου μορφοποιούνται σε χαμηλότερου επιπέδου πολιτικές και χαρακτηρίζονται πια από τεχνολογικές εντολές.

Αυτό το κεφάλαιο αποτελεί το βασικό κορμό αυτής της διπλωματικής εργασίας γι' αυτό αρχικά ξεκινάει με ένα ορισμό για το τι σημαίνει ο όρος διακυβέρνηση. Στην συνέχεια και στην ενότητα 4.3 παρουσιάζονται σημαντικές προκλήσεις που αντιμετωπίζει αυτή ώστε να επιτύχει μια σωστή λειτουργία. Παράλληλα, στην ενότητα 4.4 γίνεται μια προσπάθεια παρουσίας των μέχρι τώρα προσεγγίσεων στην περιοχή της διακυβέρνησης ώστε να φτάσουμε στην ενότητα 4.5 όπου γίνεται η παρουσίαση της προσέγγισης που ακολουθήθηκε σε αυτήν την εργασία.

4.2. Ορισμός για τη Δικτυακή Διακυβέρνηση

Για την διακυβέρνηση του δικτύου μπορούν να βρεθούν αρκετοί ορισμοί στην βιβλιογραφία. Ο πιο προσιτός που υιοθετούμε σε αυτή την εργασία λέει ότι η διακυβέρνηση είναι ένας μηχανισμός υψηλού επιπέδου που περιέχει όλες εκείνες τις λειτουργίες που απαιτούνται ώστε να καλυφθεί το κενό μεταξύ των διαμορφώσεων υψηλού επιπέδου που ορίζουν οι φορείς και την υπάρχουσα διαχείριση των πηγών της υποδομής. Ο όρος διακυβέρνηση επίσης περιέχει τη επικοινωνία ανθρώπου – δικτύου (H2N) και ο ορισμός πολιτικών και επιχειρηματικών στόχων στο δίκτυο.

4.3. Προκλήσεις της Δικτυακής Διακυβέρνησης

Ο αυξανόμενος αριθμός των ετερογενών συσκευών που χρησιμοποιούνται σε διαφορετικές περιοχές ώστε να χρησιμοποιήσουν διαφορετικές υπηρεσίες ή εφαρμογές με εύχρηστο, ασφαλή και δυναμικό τρόπο, απαιτεί τηλεπικοινωνιακή υποδομή με πλήρως κατανεμημένες δυνατότητες [31]. Αυτή η πολυπλοκότητα της υποδομής είναι από τις βασικότερες προκλήσεις που αντιμετωπίζει σήμερα η τηλεπικοινωνιακή βιομηχανία. Οι τηλεπικοινωνιακοί φορείς πρέπει να αποσκοπήσουν σε αυτό το παράδειγμα και να ρίξουν το βάρος τους στη διαχείριση της πολυπλοκότητας [32].

Η υποδομή που προσπαθούν να υλοποιήσουν πρέπει να είναι αυτόνομη και να λειτουργεί με δυναμικές διασυνδέσεις. Η υλοποίηση του πλαισίου της διακυβέρνησης είναι μέρος αυτής της υποδομής. Παρόλα αυτά, η ανάπτυξη αυτή συναντά σε προκλήσεις σε πέντε βασικές τεχνολογικές κατηγορίες.

- Επιχειρηματική γλώσσα:

Η διακυβέρνηση του δικτύου προβλέπεται να προσφέρει μηχανισμούς που να καθορίζουν τα χαρακτηριστικά και τις αλλαγές στους σε γλώσσα υψηλού επιπέδου. Για να γίνει όμως αυτό θα πρέπει οι φορείς θα πρέπει να βοηθηθούν με τη χρήση αυτής της γλώσσας ώστε να εκφράσουν τις ανάγκες του δικτύου.

- Μετάφραση:

Αφού οι φορείς ορίσουν τις απαιτήσεις τους με την επιχειρησιακή γλώσσα, τώρα αυτές οι απαιτήσεις θα πρέπει να μεταφραστούν σε πολιτικές που να καθορίζουν την κατάλληλη λειτουργική συμπεριφορά ώστε να επιτύχουν αυτόνομη λειτουργικότητα.

- Συλλογισμός:

Ο συλλογισμός είναι μια σημαντική πρόκληση στο πεδίο της δικτυακής διακυβέρνησης δεδομένου ότι μπορεί να αξιοποιηθεί για τη διαμεσολάβηση και διαπραγμάτευση μεταξύ των μεμονωμένων οντοτήτων.

- Πολιτικές:

Οι πολιτικές είναι στενά συνδεμένες με την διακυβέρνηση του δικτύου. Οι πολιτικές ορίζουν κανόνες που θα καθορίσουν τη συμπεριφορά των διαχειριζόμενων οντοτήτων.

- Επιβολή διαμόρφωσης:

Οι μηχανισμοί επιβολής διαμόρφωσης ώστε να επιτύχουν την απόφαση διαμόρφωσης.

4.4. State of the art

Ο όρος διακυβέρνηση του δικτύου δεν είναι πρωτοεμφανιζόμενος. Παλαιότερες έρευνες έχουν προσεγγίσει αρκετά αυτόν όρο με διάφορες τεχνικές και λειτουργικότητες. Παρακάτω γίνεται μια αναφορά σ' αυτά.

4.4.1. Δικτυακή Διακυβέρνηση

Αυτόνομες αρχιτεκτονικέ, πλαίσια και προγράμματα που χρησιμοποίησαν τον όρο της διακυβέρνησης του δικτύου:

- “4WARD” [33]
- “AUTOI” ([35] - [39]),
- “CASCADAS” ([40], [41]),
- E^3 [43]
- “FOCALE” ([46], [47])
- “GANA”, [49]).
- “Self-NET”[50].

➢ Επιχειρηματική γλώσσα

Επιχειρηματικές γλώσσες που χρησιμοποιήθηκαν για τον ορισμός υψηλού επιπέδου στόχων:

➢ Common Information Model [53]

➢ Shared Information and Data model[55]

➢ Άλλες μορφές μορφοποίησης (όπως τα SLA [57])

4.4.2. Μηχανισμοί μετάφρασης

Μηχανισμοί μετάφρασης των πολιτικών σε χαμηλότερου επιπέδου εντολές:

➢ Semantics & reasoning [61]

➢ DARPA Agent Mark-up Language

➢ Web Ontology Language[62]

➢ Semantic Web Rule Language [63]

4.5. Περιγραφή της μεθοδολογίας/ Προσέγγισης που υιοθετήθηκε

4.5.1. Η H2N διεπαφή

Σε αυτή την ενότητα περιγράφεται πως αναπτύχτηκε η διεπαφή ανθρώπου- δικτύου (H2N) ως μέρος του λειτουργικού κομματιού της διακυβέρνησης (Figure 1).

Η βασική λειτουργία της H2N διεπαφής συνοψίζεται στο προσφέρει στους φορείς ένα εργαλείο ώστε να μπορούν να του εισάγουν υψηλού επιπέδου επιχειρηματικούς στόχους, αυτοί να μεταφράζονται αυτόνομα σε τεχνικές εντολές έτσι ώστε να μην μεσολαβεί καμία ανθρώπινη επαφή με τεχνικά ζητήματα. Μια όψη υψηλού επιπέδου της H2N διεπαφής φαίνεται στην εικόνα Figure 2.

Οι επιχειρηματικοί στόχοι έχουν σχέση με τον προσδιορισμό νέας εφαρμογής, μια ομάδα από user classes αυτής της εφαρμογής, ομάδες ποιότητας υπηρεσίας (QoS) για κάθε user class της εφαρμογής, κλπ. αυτός ο προσδιορισμός μπορεί να συσχετιστεί με μια συγκεκριμένη περιοχή, χρονική περίοδο, αριθμό χρηστών κλπ. Επιπλέον η H2N διεπαφή επιτρέπει να πραγματοποιούνται συσχετισμοί μεταξύ των εφαρμογών/ υπηρεσιών και των User Classes, των QoS levels, των τεχνολογιών του δικτύου, άλλες εφαρμογές, QoS levels με άλλες QoS παραμέτρους.

Αυτές οι υψηλού επιπέδου πολιτικές που ορίζονται από την H2N διεπαφή στη συνέχεια ταξιδεύουν στο δίκτυο περνώντας τα διάφορα επίπεδα του ώστε να μεταφραστούν σε τεχνολογικές εντολές. Η επόμενη λειτουργική μονάδα μετα την H2N διεπαφή είναι ο Context Acquisition (Figure 1) η οποία μεταφράζει την στον επιχειρηματικό στόχο από απαιτήσεις υπηρεσίας σε εντολές δικτυακής διαμόρφωσης κι επιτρέπει στο το σύστημα να λειτουργήσει αυτόνομα.

4.5.1.1. Information flow

Όπως ήδη περιγράφηκε προηγούμενος η διακυβέρνηση επιτρέπει τον προσδιορισμό επιχειρηματικών στόχων/ πολιτικών υψηλού επιπέδου μέσω της H2N διεπαφής. Τυπικά, οι υψηλού επιπέδου επιχειρηματικές πολιτικές καθορίζουν επιχειρηματικές προκλήσεις/ στόχους. Για αυτό τον λόγο η H2N διεπαφή ορίζει δύο τύπους υψηλού επιπέδου πολιτικές. Αυτό γίνεται με τον προσδιορισμό αντίστοιχων μηνυμάτων τα οποία ονομάζονται Business level entries και Associations και αναλύονται στις υπο-ενότητες, 4.5.1.2 και 4.5.1.3 αντίστοιχα. Επίσης στην υπο-ενότητα 4.5.1.4 παρουσιάζεται οι χαμηλότερου επιπέδου πολιτικές που ενεργοποιούν την λειτουργία της αναδιαμόρφωσης των συσκευών για τους χρήστες και των στοιχείων του δικτύου.

4.5.1.2. Business level entries

Αυτή η υποενότητα παρουσιάζει τα business level entries σαν παράδειγμα υψηλού επιπέδου στόχων (πολιτικών). Τα Business level entries είναι πληροφορίες που εναπόκεινται σε επιχειρηματικό επίπεδο και σχετίζονται με τον αριθμό των χρηστών που αναμένουν μια εφαρμογή, ένα user class, σε μια συγκεκριμένη περιοχή και χρονική στιγμή. Συγκεκριμένα, όπως μπορούμε να δούμε στην Figure 3, τα business level entries συνοψίζουν πληροφορίες όπως ο αριθμός των χρηστών, το ποσοστό της κινητικότητας, π.χ., τον αριθμό των χρηστών που αναμένουν εκείνη τη στιγμή, στην περιοχή (π.χ. του Πειραιά, κέντρο της Αθήνας, ...), την χρονική στιγμή (π.χ. 08:00-11:00, 21:00-22:00, ...), την εφαρμογή (π.χ. IPTV, ...), το User Class (π.χ. Gold, Silver, Bronze, ...), το Quality Level (π.χ. High, Medium, Low), τις παραμέτρους του Quality level (π.χ. Bit rate, delay, jitter, ...) και το Mobility pattern (π.χ. High, low, train, car, ...).

4.5.1.3. Associations

Σε αυτή την υπο-ενότητα παρουσιάζονται οι οντότητες που συσχετίζονται κι αποτελούν ένα επιπλέον παράδειγμα υψηλού επιπέδου στόχων (πολιτικών). Οι συσχετίσεις είναι υψηλού επιπέδου πολιτικές που περιέχουν κανόνες συσχέτισης της εφαρμογής με τις user classes και τα quality levels, τον συνδυασμό μιας συγκεκριμένης εφαρμογής με άλλες εφαρμογές και τον συνδυασμό μεταξύ των user classes. Όπως φαίνεται στην Figure 4, μια συσχέτιση συνοψίζει πληροφορίες για ένα σετ εφαρμογών. Κάθε εφαρμογή μπορεί να συσχετιστεί με ένα ή περισσότερα User Classes. Κάθε User Class μπορεί να συσχετιστεί με ένα ή περισσότερα QoS levels. Κάθεh QoS level μπορεί να συσχετιστεί με έναν ή περισσότερους QoS παραμέτρους.

4.5.1.4. Πολιτικές Διαμόρφωσης

Οι πολιτικές διαμόρφωσης προσφέρουν κανόνες οι περιορισμούς τα οποία θα πρέπει να λαμβάνονται υπόψη στην επιλογή της βέλτιστης διαμόρφωσης μιας Περιοχής υπηρεσιών, ενός Access Point ή μιας Συσκευής. Οι πολιτικές διαμόρφωσης θα πρέπει να προκύπτουν από τα business level entries και από τις συσχετίσεις (Figure 5). Μια πολιτική διαμόρφωσης.

5. ΑΝΑΠΤΥΞΗ ΤΗΣ ΕΦΑΡΜΟΓΗΣ ΚΑΙ ΛΕΙΤΟΥΡΓΙΑ ΤΗΣ H2N ΔΙΑΣΥΝΔΕΣΗΣ

5.1. Εισαγωγή

Σε αυτή την ενότητα γίνεται μια συνολική περιγραφή της λειτουργίας της διεπαφής αλλά και τα τεχνικά χαρακτηριστικά που χρησιμοποιήθηκαν για την ανάπτυξή της.

5.2. Απαιτήσεις σχεδιασμού

Η H2N διεπαφή προσφέρει ένα γραφικό περιβάλλον που επιτρέπει στον φορέα να αλληλεπιδρά με στο σύστημα. Η παρούσα διεπαφή έχει αναπτυχθεί με Java κώδικα, μέρος του οποίου βρίσκετε στο APPENDIX I:

Το λειτουργικό κομμάτι της διακυβέρνησης είναι ένας πράκτορας και γι' αυτό αναπτύσσεται σε περιβάλλον Java Agent DEvelopment Framework (JADE) [70]. Στην Figure 6 φαίνεται αυτό το περιβάλλον όπως και τα μηνύματα που ανταλλάσσονται μεταξύ των πρακτόρων.

Για την υλοποίηση ενός μηνύματος χρησιμοποιούνται XML αρχεία για την εφαρμογή των Java κλάσεων.

Παράλληλα, για το σχεδιασμό της διεπαφής μας σχεδιάστικε μια βάση δεδομένων η επονομαζόμενη "governancedb" η οποία περιέχει πίνακες όπως οι:

- initparameters
- initsq
- locationdata
- mobilitydata
- parametersdata
- poadata
- qualdata
- servdata
- servrules
- techdata

- timezonedata
- trfdata
- trfrules
- ucdata

Η εφαρμογή χτίστηκε με τη βοήθεια του “EMS SQL Manager 2005” προγράμματος.

5.3. Η λειτουργικότητα της διεπαφής

Μια σειρά από εικόνων παραθέτουν την λειτουργία και της διεπαφής. Στην Figure 21 παρουσιάζεται μια σύνοψη των παραμέτρων που μπορούν να μεταβληθούν μέσω του γραφικού περιβάλλοντος και οι οποίοι είναι οι:

- Εφαρμογή (Figure 8)
- User Class (Figure 9)
- QoS Levels (Figure 10 και Figure 11)
- QoS παράμετροι (Figure 12 και Figure 13)
- Η τεχνολογία (Figure 14).
- CAP (Figure 15)
- Τοποθεσία (Figure 16)
- Χρονική στιγμή (Figure 17) και
- Mobility Pattern (Figure 18).

6. ΕΠΊΛΟΓΟΣ

Η εμφάνιση του το μελλοντικού διαδικτύου κατέστησε αδύναμους τους φορείς να είναι σε θέση να το αντιμετωπίσουν με τη χρήση τεχνικών διαχείρισης που υπάρχουν σήμερα. Εφαρμόζοντας αυτόνομη αρχές για τη διαχείριση του δικτύου είναι ιδιαίτερα ενδιαφέρον και την ίδια στιγμή φαίνεται να είναι ο μόνος τρόπος αντίδρασης.

Ωστόσο, η πολυπλοκότητα των συστημάτων και των εφαρμογών του σήμερα απαιτεί την ανάγκη να αυτοματοποιήσουν τη διαχείριση τους, ώστε να πληρούν τις λειτουργικές ή συμπεριφορικές απαιτήσεις. Στην πραγματικότητα, η ουσία του αυτόνομου δικτύου, και, συνεπώς, αυτόνομη επικοινωνία, είναι να επιτρέψει στο αυτόνομο δίκτυο (τόσο τα συστήματα και τα συστατικά που συνθέτουν το δίκτυο) την προσαρμογή των υπηρεσιών και των πόρων που προσφέρουν, σύμφωνα με τις μεταβαλλόμενες ανάγκες των χρηστών, τις περιβαλλοντικές συνθήκες , και τους επιχειρησιακούς στόχους, αρκεί να εξασφαλίζεται η προστασία των εκάστοτε επιχειρηματικών στόχων.

Παρ 'όλα αυτά, αυτό που θα εξαχθεί από αυτή την ανάλυση είναι ότι πολλές από τις πρωτοβουλίες που εξετάστηκαν κίνησαν την προσοχή τους προς την κατεύθυνση μιας πραγματικής διακυβέρνησης των αυτοδιαχειριζόμενων συμπεριφορών με στόχο που θα οδηγήσει στα αυτόνομα δίκτυα και συστήματα που θα χουν τη δυνατότητα δυναμικής προσαρμογής στις αλλαγές, σύμφωνα με του υψηλού επιπέδου επιχειρήσεων πολιτικών.

Στο πλαίσιο αυτής της εξέλιξης οι ανθρώπινοι φορείς έθεσαν νέες απαιτήσεις, με την ονομασία επιχειρησιακοί στόχοι, μέσω των πολιτικών διαχείρισης. Αυτοί οι επιχειρησιακοί στόχοι μπορούν να χωριστούν σε υψηλού επιπέδου επιχειρηματικούς στόχους που εκφράζουν την τους στόχους επιδόσεων των φορέων και των χρηστών και σε χαμηλότερο επιπέδου στόχους που εκφράζουν τις τεχνικές εντολές των συστημάτων. Από αυτό συμπεραίνουμε ότι δημιουργείται η ανάγκη να γεφυρωθεί το χάσμα μεταξύ αυτών των δυο ώστε να εγγυηθεί μια μετακίνηση προς τη διακυβέρνηση των αυτοδιαχειριζόμενων συμπεριφορές.

Έτσι, το ενδιαφέρον μας στράφηκε στην ανάπτυξη αυτού του αυτόνομου συστήματος διαχείρισης, προκειμένου να καταστεί δυνατή η αποτελεσματική λειτουργία και τη χρησιμοποίηση των συστημάτων και υπηρεσιών με μικρή ή μηδαμινή ανθρώπινη παρέμβαση.

Η H2N διασύνδεση αποτελεί μέρος του εργαλείου της διακυβέρνησης, ένα λειτουργικό τμήμα που εξάγει υψηλού επιπέδου πολιτικές, και τους ονομάζει επιχειρηματικούς στόχους. Αυτοί οι επιχειρηματικοί στόχοι των επιχειρήσεων αναφέρονται σε εφαρμογές σε μια συγκεκριμένη θέση και ζητούν ορισμένες πόρους σύμφωνα με τις εκτιμήσεις τους. Προκειμένου να διασφαλιστεί ότι η λειτουργικότητα του συστήματος προσαρμόζεται στις μεταβολές των επιχειρησιακών στόχων, αυτοί οι στόχοι θα πρέπει να μεταφραστούν σε τεχνικά στόχους χαμηλότερου επιπέδου ώστε να ενημερωθούν όλα τα επίπεδα των δικτύων. Αυτό θα μπορούσε να αποτελείται από ένα μελλοντικό στάδιο της ολοκλήρωσης.

Από την άλλη πλευρά, αυτό το αυτόνομο παράδειγμα επιδιώκει να μειώσει την απαίτηση για ανθρώπινη παρέμβαση στη διαδικασία της διαχείρισης με τη χρήση ενός ή περισσότερων βρόχων ελέγχου που συνεχώς να ρυθμίζουν το σύστημα ώστε να διατηρήσει τη συμπεριφορά του εντός επιθυμητών ορίων.

Ωστόσο, με μια δήλωση όπως η «αυτοματισμός δεν σημαίνει ανεξαρτησία", λίγη προσοχή δίνεται στο είδος του διαλόγου ανάμεσα σε ένα ανθρώπινο χειριστή δικτύου (HNO) και την αυτο-διαχείριση του δικτύου και ιδίως για τις καινοτομίες και τις ιδιαιτερότητες που πιθανότατα το HNO θα πρέπει να χειριστεί κατά τη μετάβαση στο νέο αυτό είδος διαχείρισης.

Οι παράγοντες που μόλις αναφέρθηκαν μπορούν να συντελέσουν ένα νέο πεδίο έρευνας, είτε με την εισαγωγή ενός σαφώς ανεξάρτητου και αυτόνομου συστήματος ή στην ανακάλυψη του της ακριβή φύση του διαλόγου μεταξύ HNO και της αυτο-διαχείρισης του δικτύου.